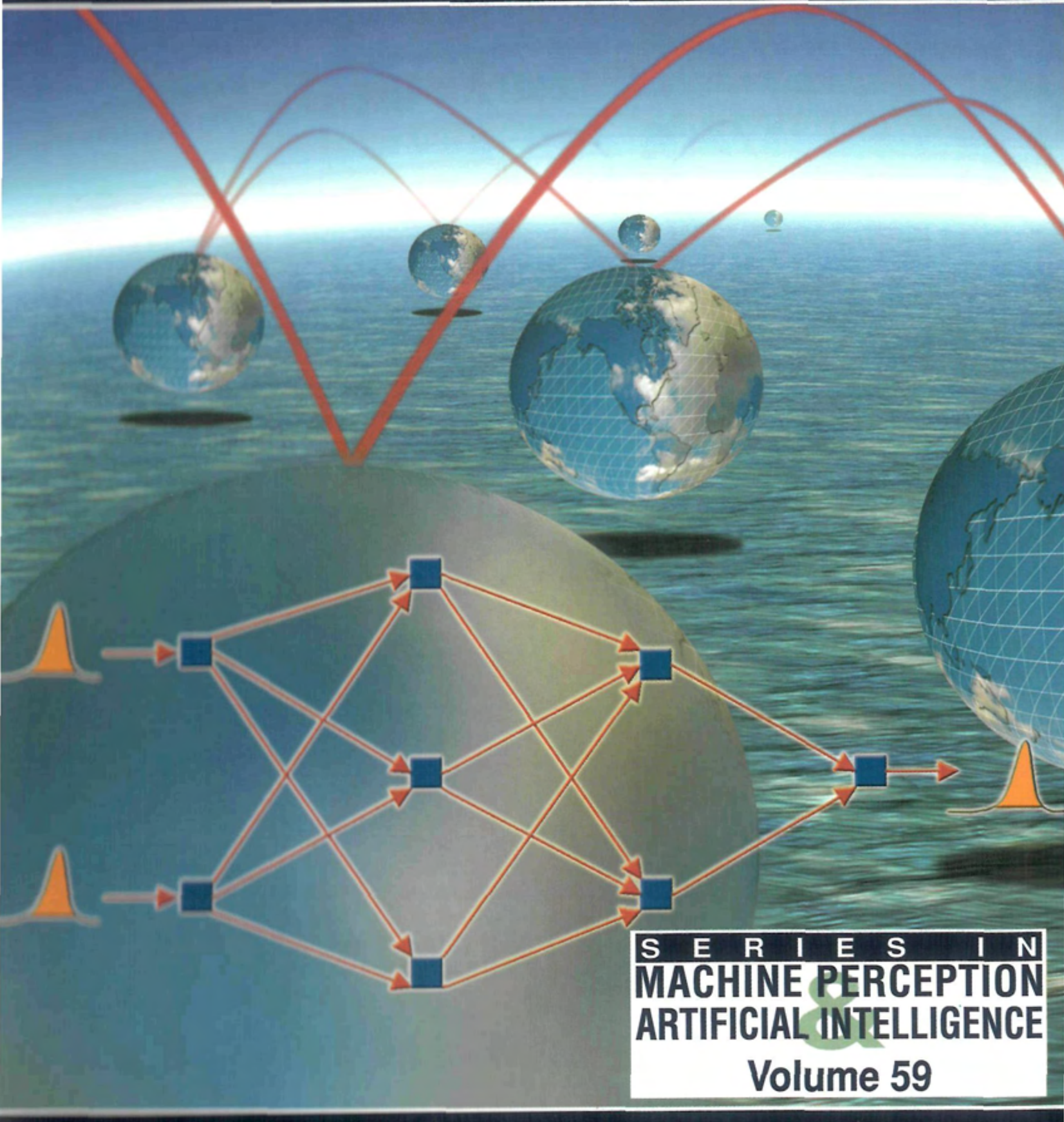
FUZZY NEURAL NETWORK THEORY AND APPLICATION

Puyin Liu • Hongxing Li



SERIES IN
MACHINE PERCEPTION & ARTIFICIAL INTELLIGENCE
Volume 59

World Scientific

FUZZY NEURAL NETWORK THEORY AND APPLICATION

SERIES IN MACHINE PERCEPTION AND ARTIFICIAL INTELLIGENCE*

Editors: **H. Bunke** (Univ. Bern, Switzerland)
P. S. P. Wang (Northeastern Univ., USA)

Vol. 43: Agent Engineering
(Eds. *Jiming Liu, Ning Zhong, Yuan Y. Tang and Patrick S. P. Wang*)

Vol. 44: Multispectral Image Processing and Pattern Recognition
(Eds. *J. Shen, P. S. P. Wang and T. Zhang*)

Vol. 45: Hidden Markov Models: Applications in Computer Vision
(Eds. *H. Bunke and T. Caelli*)

Vol. 46: Syntactic Pattern Recognition for Seismic Oil Exploration
(*K. Y. Huang*)

Vol. 47: Hybrid Methods in Pattern Recognition
(Eds. *H. Bunke and A. Kandel*)

Vol. 48: Multimodal Interface for Human-Machine Communications
(Eds. *P. C. Yuen, Y. Y. Tang and P. S. P. Wang*)

Vol. 49: Neural Networks and Systolic Array Design
(Eds. *D. Zhang and S. K. Pal*)

Vol. 50: Empirical Evaluation Methods in Computer Vision
(Eds. *H. I. Christensen and P. J. Phillips*)

Vol. 51: Automatic Diatom Identification
(Eds. *H. du Buf and M. M. Bayer*)

Vol. 52: Advances in Image Processing and Understanding
A Festschrift for Thomas S. Huwang
(Eds. *A. C. Bovik, C. W. Chen and D. Goldgof*)

Vol. 53: Soft Computing Approach to Pattern Recognition and Image Processing
(Eds. *A. Ghosh and S. K. Pal*)

Vol. 54: Fundamentals of Robotics — Linking Perception to Action
(*M. Xie*)

Vol. 55: Web Document Analysis: Challenges and Opportunities
(Eds. *A. Antonacopoulos and J. Hu*)

Vol. 56: Artificial Intelligence Methods in Software Testing
(Eds. *M. Last, A. Kandel and H. Bunke*)

Vol. 57: Data Mining in Time Series Databases
(Eds. *M. Last, A. Kandel and H. Bunke*)

Vol. 58: Computational Web Intelligence: Intelligent Technology for
Web Applications
(Eds. *Y. Zhang, A. Kandel, T. Y. Lin and Y. Yao*)

Vol. 59: Fuzzy Neural Network Theory and Application
(*P. Liu and H. Li*)

*For the complete list of titles in this series, please write to the Publisher.

Series in Machine Perception and Artificial Intelligence – Vol. 59

FUZZY NEURAL NETWORK THEORY AND APPLICATION

Puyin Liu

National University of Defense Technology
Changsha, China

Hongxing Li

Beijing Normal University
Beijing, China

World Scientific

NEW JERSEY • LONDON • SINGAPORE • BEIJING • SHANGHAI • HONG KONG • TAIPEI • CHENNAI

Published by

World Scientific Publishing Co. Pte. Ltd.

5 Toh Tuck Link, Singapore 596224

USA office: Suite 202, 1060 Main Street, River Edge, NJ 07661

UK office: 57 Shelton Street, Covent Garden, London WC2H 9HE

British Library Cataloguing-in-Publication Data
A catalogue record for this book is available from the British Library.

FUZZY NEURAL NETWORK THEORY AND APPLICATION

ISBN 981-238-786-2

Printed in Singapore by World Scientific Printers (S) Pte Ltd

Table of Contents

Foreword

Authored by Professors P. Liu and H. Li, "Fuzzy Neural Network Theory and Application," or FNNTA for short, is a highly important work. Essentially, FNTTA is a treatise that deals authoritatively and in depth with the basic issues and problems that arise in the conception, design and utilization of fuzzy neural networks. Much of the theory developed in FNTTA goes considerably beyond what can be found in the literature.

Fuzzy neural networks, or more or less equivalently, neurofuzzy systems, as they are frequently called, have a long history. The embryo was a paper on fuzzy neurons by my former student, Ed Lee, which was published in 1975. Thereafter, there was little activity until 1988, when H. Takagi and I. Hayashi obtained a basic patent in Japan, assigned to Matsushita, which described systems in which techniques drawn from fuzzy logic and neural networks were employed in combination to achieve superior performance.

The pioneering work of Takagi and Hayashi opened the door to development of a wide variety of neurofuzzy systems. Today, there is an extensive literature and a broad spectrum of applications, especially in the realm of consumer products.

A question which arises is: Why is there so much interest and activity in the realm of neurofuzzy systems? What is it that neurofuzzy systems can do that cannot be done equally well by other types of systems? To understand the underlying issues, it is helpful to view neurofuzzy systems in a broader perspective, namely, in the context of soft computing.

What is soft computing? In science, as in many other realms of human activity, there is a tendency to be nationalistic-to commit oneself to a particular methodology and employ it exclusively. A case in point is the well-known Hammer Principle: When the only tool you have is a hammer, everything looks like a nail. Another version is what I call the Vodka Principle: No matter what your problem is, vodka will solve it.

What is quite obvious is that if A, B, ..., N are complementary methodologies, then much can be gained by forming a coalition of A, B, ..., N. In this perspective, soft computing is a coalition of methodologies which are tolerant of imprecision, uncertainty and partial truth, and which collectively provide a foundation for conception, design and utilization of intelligent systems. The principal numbers of the coalition are: fuzzy logic, neurocomputing, evolutionary computing, probabilistic computing, rough set theory, chaotic computing and machine learning. A basic credo which underlies soft computing is that, in general, better results can be obtained by employing the constituent methodologies of soft computing in combination rather in a stand-alone mode.

In this broader perspective, neurofuzzy systems may be viewed as the domain of a synergistic combination of neurocomputing and fuzzy logic; inheriting from neurocomputing the concepts and techniques related to learning and approximation, and inheriting from fuzzy logic the concepts and techniques related to granulation, linguistic variable, fuzzy if-then rules and rules of inference and constraint propagation.

An important type of neurofuzzy system which was pioneered by Arabshahi et al starts with a neuro-based algorithm such as the backpropagation algorithm, and improves its performance by employing fuzzy if-then rules for adaptive adjustment of parameters. What should be noted is that the basic idea underlying this approach is applicable to any type of algorithm in which human expertise plays an essential role in choosing parameter-values and controlling their variation as a function of performance. In such applications, fuzzy if-then rules are employed as a language for describing human expertise.

Another important direction which emerged in the early nineties involves viewing a Takaga-Sugeno fuzzy inference system as a multilayer network which is similar to a multilayer neural network. Parameter adjustment in such systems is achieved through the use of gradient techniques which are very similar to those associated with backpropagation. A prominent example is the ANFIS system developed by Roger Jang, a student of mine who conceived ANFIS as a part of his doctoral dissertation at UC Berkeley. The widely used method of radical basis functions falls into the same category.

Still another important direction—a direction initiated by G. Bortolan—involves a fuzzification of a multilayer, feedforward neural network, resulting in a fuzzy neural network, FNN. It is this direction that is the principal concern of the work of Professors Liu and Li.

Much of the material in FNNTA is original with the authors and reflects their extensive experience. The coverage is both broad and deep, extending from the basics of FNN and FAM (fuzzy associate memories) to approximation theory of fuzzy systems, stochastic fuzzy systems and application to image restoration. What is particularly worthy of note is the author's treatment of universal approximation of fuzzy-valued functions.

A basic issue that has a position of centrality in fuzzy neural network theory—and is treated as such by the authors— is that of approximation and, in particular, universal approximation. Clearly, universal approximation is an issue that is of great theoretical interest. A question which arises is: Does the theory of universal approximation come to grips with problems which arise in the design of fuzzy neural networks in realistic settings? I believe that this issue is in need of further exploration. In particular, my feeling is that the usual assumption about continuity of the function that is approximated, is too weak, and that the problem of approximation of functions which are smooth, rather than continuous, with smoothness defined as a fuzzy characteristic, that is, a matter of degree, must be addressed.

FNNTA is not intended for a casual reader. It is a deep work which addresses complex issues and aims at definitive answers. It ventures into territories which have not been explored, and lays the groundwork for new and important applications. Professors Liu and Li, and the publisher, deserve our thanks and congratulations for producing a work that is an important contribution not just to the theory of fuzzy neural networks, but, more broadly, to the conception and design of intelligent systems.

Lotfi A. Zadeh
Professor in the Graduate School, Computer Science Division
Department of Electrical Engineering and Computer Sciences
University of California
Berkeley, CA 94720–1776
Director, Berkeley Initiative in Soft Computing (BISC)

March, 2004

This page is intentionally left blank

Preface

As a hybrid intelligent system of soft computing technique, the fuzzy neural network (FNN) is an efficient tool to deal with nonlinearly complicated systems, in which there are linguistic information and data information, simultaneously. In view of two basic problems—learning algorithm and universal approximation, FNN's are thoroughly and systematically studied in the book. The achievements obtained here are applied successfully to pattern recognition, system modeling and identification, system forecasting, and digital image restoration and so on. Many efficient methods and techniques to treat these practical problems are developed.

As two main research objects, learning algorithms and universal approximations of FNN's constitute the central part of the book. The basic tools to study learning algorithms are the max–min ($\vee - \wedge$) functions, the cuts of fuzzy sets and interval arithmetic, etc. And the bridges to research universal approximations of fuzzified neural networks and fuzzy inference type networks, such as regular FNN's, polygonal FNN's, generalized fuzzy systems and generalized fuzzy inference networks and so on are the fuzzy valued Bernstein polynomial, the improved type extension principle and the piecewise linear functions. The achievements of the book will provide us with the necessary theoretic basis for soft computing technique and the applications of FNN's.

There have been a few of books and monographs on the subject of FNN's or neuro-fuzzy systems. There are several distinctive aspects which together make this book unique.

First of all, the book is a thorough summation and deepening of authors' works in recent years in the fields related. So the readers can get latest information, including latest research surveys and references related to the subjects through this book. This book treats FNN models both from mathematical perspective with the details of most proofs of the results included: only simple and obvious proofs are omitted, and from applied or computational aspects with the realization steps of main results shown, also many application examples included. So it is helpful for readers who are interested in mathematical aspects of FNN's, also useful for those who do not concern themselves with the details of the proofs but the applied aspects of FNN's.

Second, the perspective of the book is centered on two typical problems on FNN's, they are universal approximation and learning algorithm. The achievements about universal approximation of FNN's may provide us with the theoretic basis for FNN applications in many real fields, such as system modeling and system identification, information processing and system optimization and so on. And learning algorithms for FNN's may lead to rational treatments of FNN's for their architectures, implementation procedures and all kinds practical applications, etc. So readers may easily enter through this book the fields related by taking the two subjects as leads. Also the book includes many well-designed simulation examples for readers' convenience to understand the results related.

Third, the arrangement of contents of the book is novel and there are few overlaps with other books related to the field. Many concepts are first introduced for approximation and learning of FNN's. The constructive proofs of universal approximations provide us with much convenience in modeling or identifying a real system by FNN's. Also they are useful to build some learning algorithms to optimize FNN architectures.

Finally almost all common FNN models are included in the book, and as many as possible references related are listed in the end of each chapter. So readers may easily find their respective contents that they are interested. Moreover, those FNN models and references make this book valuable to people interested in various FNN models and applications.

The specific prerequisites include fuzzy set theory, neural networks, interval analysis and image processing. For the fuzzy theory one of the following books should be sufficient: Zimmermann H. -J. (1991), Dubois D. and Prade H. (1980). For the neural networks one of the following can provide with sufficient background: Khanna T. (1990), Haykin S. (1994). For the interval analysis it suffices to reference one of following books: Alefeld G. and Herzberger J. (1983), Diamond P. and Kloeden P. (1994). And for image processing one of the following books is sufficient: Jain A. K. (1989), Astola J. and Kuosmanen P. (1997). The details of these books please see references in Chapter I.

Now let us sketch out the main points of the book, and the details will be presented in Chapter I. This book consists of four primary parts: the first part focuses mainly on FNN's based on fuzzy operators '$\vee$' and '$\wedge$', including FNN's for storing and classifying fuzzy patterns, dynamical FNN's taking fuzzy Hopfield networks and fuzzy bidirectional associative memory (FBAM) as typical models. They are dealt with in Chapter II and Chapter III, respectively. The second part is mainly contributed to the research of universal approximations of fuzzified neural networks and their learning algorithms. The fuzzified neural networks mean mainly two classes of FNN's, i.e. regular FNN's and polygonal FNN's. A series of equivalent conditions that guarantee universal approximations are built, and several learning algorithms for fuzzy weights are developed. Also implementations and applications of fuzzified neural networks are included. The third part focuses on the research of universal approximations, including ones of generalized fuzzy systems to integrable functions and ones of stochastic fuzzy systems to some stochastic processes. Also the learning algorithms for the stochastic fuzzy systems are studied. The fourth part is contributed to the applications of the achievements and methodology on FNN's to digital image restoration. A FNN representation of digital images is built for reconstructing images and filtering noises. Based on fuzzy inference networks some efficient FNN filters are developed for removing impulse noise and restoring images.

When referring to a theorem, a lemma, a corollary, a definition, etc in the same chapter, we utilize the respective numbers as they appear in the statements, respectively. For example, Theorem 4.2 means the second theorem in Chapter IV, while Definition 2.4 indicates the fourth definition in Chapter II, and so on.

Although we have tried very hard to give references to original papers, there are many researchers working on FNN's and we are not always aware of contributions by various authors, to which we should give credit. We have to say sorry for our omissions. However we think the references that we have listed are helpful for readers to find the related works in the literatures.

We are indebted to Professor Lotfi A. Zadeh of University of California, Berkeley who writes the preface of the book in the midst of pressing affairs at authors' invitation. We are specially grateful to Professors Guo Guirong and He Xingui who read the book carefully and make their many of insightful comments. Thanks are also due to Professor Bunke Horst who accepts this book in the new book series edited by him. Finally we express our indebtedness to Dr. Seldrup Ian the editor of the book and the staff at the World Scientific Publishing for displaying a lot of patience in our final cooperation.

We were supported by several National Natural Science Foundation Grants of China (e.g., No.69974041, No.60375023 and No.60174013) during the years this book was written.

Puyin Liu and Hongxing Li

March 2004

CHAPTER I

Introduction

As information techniques including their theory and applications develop further, the studying objects related have become highly nonlinear and complicated systems, in which natural linguistic information and data information coexist [40]. In practice, a biological control mechanism can carry out complex tasks without having to develop some mathematical models, and without solving any complex integral, differential or any other types of mathematical equations. However, it is extremely difficult to make an artificial mobile robot to perform the same tasks with vague and imprecise information for the robot involves a fusion of most existing control techniques, such as adaptive control, knowledge-based engineering, fuzzy logic and neural computation and so on. To simulate biological control mechanisms, efficiently and to understand biological computational power, thoroughly a few of powerful fields in modern technology have recently emerged [30, 61]. Those techniques take their source at Zadeh's soft data analysis, fuzzy logic and neural networks together with genetic algorithm and probabilistic reasoning [68–71]. The soft computing techniques can provide us with an efficient computation tool to deal with the highly nonlinear and complicated systems [67]. As a collection of methodologies, such as fuzzy logic, neural computing, probabilistic reasoning and genetic algorithm (GA) and so on, soft computing is to exploit the tolerance for imprecision, uncertainty and partly truth to achieve tractability, robustness and low solution cost. In the partnership of fuzzy logic, neural computing and probabilistic reasoning, fuzzy logic is mainly concerned with imprecision and approximate reasoning, neural computing with learning and curve fitting, and probabilistic reasoning with uncertainty and belief propagation.

§1.1 Classification of fuzzy neural networks

As a main ingredient of soft computing, fuzzy neural network (FNN) is a hybrid intelligent system that possess the capabilities of adjusting adaptively and intelligent information processing. In [36, 37] Lee S. C. and Lee E. T. firstly proposed the fuzzy neurons and some systematic results on FNN's were developed by softening the McCulloch–Pitts neurons in the middle 1970s when the interest in neural networks faltered. So such novel neural systems had not attracted any attention until 1987 when Kosko B. developed a fuzzy associative memory (FAM) to deal with intelligent information by introducing some fuzzy operators in associative memory networks [32]. Since the early 1980s the research on neural networks has increased dramatically because of the works done by Hopfield J. J. (see [26]). The FNN models have also attracted many scholars' attention. A lot of new new concepts, such as innovative architecture and training rules and models about FNN's have been developed [30, 32, 61]. In practice FNN's have found useful in many application fields, for instance, system modelling [16, 24], system reliability analysis [7, 42], pattern recognition [33, 56], and knowledge engineering and so on. Based on fuzziness involved in FNN's developed since the late of 1980s, one may broadly classify all FNN models as three main types as shown in Figure 1.1:

- FNN
  - Based on fuzzy operators
    - Feedforward neural networks (1980's),
    - Feedback neural networks (1990's),
  - Fuzzified neural networks
    - Regular FNN's (1990's),
    - Improved FNN's (1990's),
  - Fuzzy inference networks
    - Mamdani type (1990's),
    - Takagi–Sugeno type (1990's),
    - Generalized type (1990's).

Figure 1.1 Classification of FNN's

§1.2 Fuzzy neural networks with fuzzy operators

FNN's based on fuzzy operators are firstly studied by Lee and Lee in 1970s. Such FNN's have become one of the foci in neural network research since Kosko introduced the fuzzy operators '$\vee$' and '$\wedge$' in associative memory to define fuzzy associative memory (FAM) in 1987. A FAM is a feedforward FNN whose information flows from input layer to output layer. It possesses the capability of storing and recalling fuzzy information or fuzzy patterns [32, 41]. So storage capability and fault-tolerance are two main problems we focus on in the research on FAM's. In practice a applicable FAM should possess strong storage capability, i.e. as many fuzzy patterns as possible can be stored in a FAM. So far many learning algorithms, including the fuzzy Hebbian rule, the fuzzy delta rule and the fuzzy back propagation (BP) algorithm and so on have been developed to train FAM's and to improve storage capability of a FAM [32]. Equally, fault-tolerance, i.e. the capability of a FAM to recall the right fuzzy pattern from a distorted input is of real importance, for in practice many fuzzy patterns to handle are inherently imprecise and distorted. The research on feedforward FAM's, including their topological architecture designs, the selection of fuzzy operators for defining the internal operations, learning algorithms and so on attracts much attention [41]. The achievements related to FAM's have found useful in many applied areas, such as pattern recognition [33, 59], pattern classification [41, 58], system analysis [38], signal processing [27] and so on.

A main object of the research on FAM's is to improve the storage capability, which relates closely to fuzzy relational equation theory. Blanco et al in [4, 5] express a fuzzy system as a fuzzy relational equation, thus, the feedforward FAM's can identify a fuzzy relation by designing suitable learning algorithm. On the other hand, many methods for solving fuzzy relational equations are employed to improve the storage capability of FAM's. Li and Ruan in [38] use FAM's based on several fuzzy operator pairs including '$\vee - \wedge$', '$\vee - \times$', '$+ - \times$' and '$+ - \wedge$' etc to identify many fuzzy system classes by building a few of novel learning algorithms. Moreover, they show the convergence of fuzzy delta type iteration algorithms. Liu et al utilize the approaches for solving fuzzy relational equations to build a series of equivalent conditions that a given fuzzy pattern family can be stored by a FAM [41]. Furthermore, some learning algorithms for improving storage capability of FAM's are developed. These constitute one of the main parts in Chapter II. Pedrycz in [51] put forward two logic type fuzzy neurons based on general fuzzy operators, and such FAM's can be applied to realize fuzzy logic relations efficiently.

Adaptive resonance theory (ART) is an efficient neural model of human cognitive information processing. It has since led to an evolving series of real-time neural network models for unsupervised category learning and pattern recognition. The model families include ART1, which can process patterns expressed as vectors whose components are either 0 or 1 [8]; ART2 which can categorize either analog or binary input patterns [9], and ART3 which can carry out parallel search, or hypothesis testing, of distributed recognition codes in a multi-level network hierarchy [10]. The fuzzy ART model developed by Carpenter et al in [11] generalizes ART1 to be capable of learning stable recognition categories in response to both analog and binary input patterns. The fuzzy operators '$\vee$' and '$\wedge$' are employed to define the operations between fuzzy patterns. The research related to fuzzy ART is mainly focused on the classification characteristics and applications to pattern recognition in all kinds of applied fields, which is presented in Chapter II.

Another important case of FNN's based on fuzzy operators is one of feedback FNN's. It imitates human brain in understanding objective world, which is a procedure of self-improving again and again. A feedback neural network as a dynamical system finishes information processing by iterating repeatedly from an initial input to the equilibrium state. In practice, an equilibrium state of a dynamical FNN turns out to be the right fuzzy pattern to recall. So the recalling procedure of a dynamical FNN is in nature a process that the FNN evolutes to its equilibrium state from an initial fuzzy pattern. In the book we focus mainly on two classes of dynamical FNN's, they are fuzzy Hopfield networks and fuzzy bidirectional associative memory (FBAM). By the comparison between the two dynamical FNN's and the corresponding crisp neural networks we can see

1. The FNN's do not need the transfer function used in the crisp networks, for a main function of the transfer function in artificial neural networks lies in controlling output range, which may achieve by the fuzzy operators '$\vee$', '$\wedge$', and '$\wedge$' is a threshold function [4, 5].

2. In practice, it is much insufficient to represent fuzzy information by the strings only consisting of 0, 1. We should utilize fuzzy patterns whose components belong to [0, 1] to describe fuzzy information. So the dynamical FNN's may be applied much more widely than the crisp networks may.

Similarly with crisp dynamical networks, in the research related to the dynamical FNN's, the stability analysis, including the global stability of dynamical systems and Lyapounov stability of the equilibrium state (attractor), the attractive basins of attractors and the discrimination of attractors and pseudo-attractors and so on are main subjects to study. Those problems will be studied thoroughly in Chapter III.

§1.3 Fuzzified neural networks

A fuzzified neural network means such a FNN whose inputs, outputs and connection weights are all fuzzy set, which is also viewed as a pure fuzzy system [61]. Through the internal relationships among fuzzy sets of a fuzzified neural network, a fuzzy input can determines a fuzzy output. One most important class of fuzzified neural networks is regular FNN class, each of which is the fuzzifications of a crisp feedforward neural network. So for a regular FNN, the topological architecture is identical to one of the corresponding crisp neural network, and the internal operations are based on Zadeh's extension principle [44] and fuzzy arithmetic [20]. Since regular FNN's were put forward by Buckley et al [6] and Ishibuchi et al [28] about in 1990s, the systematic achievements related have been built by focusing mainly on two basic problem—learning algorithm and universal approximation.

1.3.1 Learning algorithm for regular FNN's

There are two main approaches to design learning algorithms for regular FNN's, they are $\alpha-$cut learning [28, 41], and the genetic algorithm (GA) for fuzzy weights [2]. The main ideas for the $\alpha-$cut learning algorithm rest with the fact that for any $\alpha \in [0, 1]$, we utilize the BP algorithm for crisp neural networks to determine the two endpoints of $\alpha-$cut and consequently establish the $\alpha-$cut of a fuzzy weight, and then define the fuzzy weight. Thus, the fuzzy connection weights of the regular FNN is trained suitably. However, the $\alpha-$cut learning algorithm loses its effectiveness frequently since for $\alpha_1, \alpha_2 \in [0, 1]: \alpha_1 < \alpha_2$, by the BP algorithm we obtain the $\alpha-$cuts $\widetilde{W}_{\alpha_1}$, $\widetilde{W}_{\alpha_2}$, and if no constraint is added, the fact $\widetilde{W}_{\alpha_2} \subset \widetilde{W}_{\alpha_1}$ can not be guaranteed. And therefore the fuzzy set $\widetilde{W}$ can not be defined. So in order to ensure the effectiveness of the algorithm it is necessary to solve the following optimization problem:

$$\begin{cases} \min\{E(\widetilde{W}_1, ..., \widetilde{W}_n)|\widetilde{W}_1, ..., \widetilde{W}_n \text{ are fuzzy sets}\}, \\ \text{s.t. } \forall \alpha_1, \alpha_2 \in [0,1] : \alpha_1 < \alpha_2, \forall i \in \{1, ..., n\}, (\widetilde{W}_i)_{\alpha_2} \subset (\widetilde{W}_i)_{\alpha_1}, \end{cases} \tag{1.1}$$

where $E(\cdot)$ is an error function. If no constraint is added, (1.1) is generally insolvable. Even if we may find a solution of (1.1) in some special cases the corresponding solving procedure will be extremely complicated. Another difficulty to hinder the realization of the $\alpha-$cut learning algorithm is to define a suitable error function $E(\cdot)$ [41], so that not only its minimization can ensure to realize the given input—output (I/O) relationship, approximately, but also its derivatives related are easy to calculate. To avoid solve (1.1), a common method to define $E(\cdot)$ is to introduce some constraints on the fuzzy connection weights, for instance, we may choose the fuzzy weights as some common fuzzy numbers such as triangular fuzzy numbers, trapezoidal fuzzy numbers and Gaussian type fuzzy numbers and so on, which can be determined by a few of adjustable parameters. Ishibuchi et al utilize the triangular or, trapezoidal fuzzy numbers to develop some $\alpha-$cut learning algorithms for training the fuzzy weights of regular FNN's. And some successful applications of regular FNN's in the approximate realization of fuzzy inference rules are demonstrated in [28]. Park et al in [50] study the inverse procedure of the learning for a regular FNN, systematically. That is, using the desired fuzzy outputs of the FNN to establish conversely the conditions for the corresponding fuzzy inputs. This is a fuzzy version of the corresponding problem for crisp neural networks [39]. Solution of such a problem rest in nature with treating the $\alpha-$cut learning.

However, no matter how different fuzzy weights and error functions these learning algorithms have, two important operations '$\vee$' and '$\wedge$' are often involved. An indispensable step to construct the fuzzy BP algorithm is to differentiate $\vee - \wedge$ operations by using the unit step function, that is, for the given real constant a, let

$$\frac{\partial(x \vee a)}{\partial x} = \begin{cases} 1, & x \geq a, \\ 0, & x < a; \end{cases} \qquad \frac{\partial(x \wedge a)}{\partial x} = \begin{cases} 1, & x \leq a, \\ 0, & x > a. \end{cases} \tag{1.2}$$

Above representations are only valid for special case $x \neq a$. And if $x = a$, they are no longer valid. Based on these two derivative formulas, the chain rules for differentiation of composition functions are only in form, and lack rigorous mathematical sense. Apply the results in [73] to analyze the $\vee - \wedge$ operations fully and to develop a rigorous theory for the calculus of $\vee$ and $\wedge$ operations are two subsidiary results in Chapter IV.

The GA's for fuzzy weights are also developed for such fuzzy sets that they can be determined uniquely by a few of adjustable parameters when it is possible to code fuzzy weights and to ensure one to one correspondence between a code sequence in GA and fuzzy connection weights. For instance, Aliev et al in [2] employ simple GA to train the triangular fuzzy number weights and biases of regular FNN's. They encode all fuzzy weights as a binary string (chromosome) to complete the search process. The transfer function σ related is assumed to be an increasing real function. The research in this field is at its infancy and many fundamental problems, such as, how to define a suitable error function? what more efficient code techniques can be employed and what are the more efficient genetic strategies? and so on remain to be solved.

Regardless of $\alpha-$cut learning algorithm and GA for the fuzzy weights of regular FNN's they are efficient only for a few of special fuzzy numbers, such as triangular or trapezoidal fuzzy numbers, Gaussian type fuzzy numbers and so on. The applications of the learning algorithms are much restricted. And therefore it is meaningful and important to develop the BP type learning algorithms or, GA for fuzzy weights of regular FNN's within a general framework, that is, we have to build learning algorithms for general fuzzy weights. The subject constitutes one of central parts of Chapter IV. To speed the convergence of the fuzzy BP algorithm we develop a fuzzy conjugate gradient (CG) algorithm [18] to train a regular FNN with general fuzzy weights.

1.3.2 Universal approximation of regular FNN's

Another basic problem for regular FNN's is the universal approximation, which can provide us with the theoretic basis for the FNN applications. The universal approximation of crisp feedforward neural networks means such a fact that for any compact set U of the input space and any continuous function f defined on the input space, f can be represented with arbitrarily given degree of accuracy $\varepsilon > 0$ by a feedforward crisp neural network. The research related has attracted many scholars since the late 1980s. It is shown that a three–layer feedforward neural network with a given nonlinear activation function in the hidden layer is capable of approximating generic class of functions, including continuous and integrable ones [13, 14, 57]. Recently Scarselli and Tsoi [57] present a detail survey of recent works on the approximation by feedforward neural networks, and obtain some new results by studying the computational aspects and training algorithms for the approximation problem. The approximate representation of a continuous function by a three layer feedforward network can with the approximate sense solve the 13−th Hilbert problem with a simple approach [57], and Kolmogorov had to employ a complicated approach to solve the problem analytically in 1950s [57]. The achievements related to the field have not only solved the approximation representation of some multivariate functions by the combination of finite compositions of one-variable functions, but also found useful in many real fields, such as the approximation of structural synthesis [57], system identification [14], pattern classification [25], and adaptive filtering [52], etc.

Since the middle 1990s many authors have begun to paid their attentions to the similar approximation problems in fuzzy environment [6, 22, 28, 41]. Firstly Buckley et al in [6] study systematically the universal approximation of FNN's and obtain such a fact: Hybrid FNN's can be approximator to fuzzy functions while regular FNN's are not capable of approximating continuous fuzzy functions to any degree of accuracy on the compact sets of a general fuzzy number space. Considering the arbitrariness of a hybrid FNN in its architectures and internal operations, we find such a FNN is inconvenient for realizations and applications. Corresponding to different practical problems the respective hybrid FNN's with different topological architectures and internal operations have to be constructed [6]. However, regular FNN's, whose topological architectures are identical to the corresponding crisp ones, internal operations are based on extension principle and fuzzy arithmetic, have found convenient and useful in many applications. Thus some important questions arise. What are the conditions for continuous fuzzy functions that can be arbitrarily closely approximated by regular FNN's? that is, which function class can guarantee universal approximation of regular FNN's to hold? Whether the corresponding equivalent conditions can be established? Since the inputs and outputs of regular FNN's are fuzzy sets, the common operation laws do not hold any more. It is difficult to employ similar approaches for dealing with crisp feedforward neural networks to solve above problems for regular FNN's.

Above problems attract many scholars' attention. At first Buckley and Hayashi [6] show a necessary condition for the fuzzy functions that can be approximated arbitrarily closely by regular FNN's, that is, the fuzzy functions are increasing. Then Feuring et al in [22] restrict the inputs of regular FNN's as trapezoidal fuzzy numbers, and build the approximate representations of a class of trapezoidal fuzzy functions by regular FNN's. Also they employ the approximation of the regular FNN's with trapezoidal fuzzy number inputs and connection weights to solve the overfitting problem. We establish some sufficient conditions for fuzzy valued functions defined on an interval $[0, T_0]$ that ensure universal approximation of three layer regular FNN's to hold [41]. However these results solve only the first problem partly, and do not answer the second problem. To solve the universal approximation of regular FNN's completely, Chapter IV and Chapter V develop comprehensive and thorough discussion to above problems. And some realization algorithms for approximating procedure are built.

In practice many I/O relationships whose internal operations are characterized by fuzzy sets are inherently fuzzy and imprecise. For instance, the natural inference of human brain, industrial process control, chemical reaction and natural evolution process and so on [17, 30]. Regular FNN's have become the efficient tools to model these real processes, for example fuzzy regression models [28], data fitting models [22], telecommunication networks [42] and so on are the successful examples of regular FNN applications.

§1.4 Fuzzy systems and fuzzy inference networks

Fuzzified neural networks as a class of pure fuzzy systems can deal with natural linguistic information efficiently. In practice, in addition to linguistic information, much more cases relate to data information. From a real fuzzy system we can get a collection of data information that characterizes its I/O relationship by digital sensor or data surveying instrument. so it is of very real importance to develop some systematic tools that are able to utilize linguistic and data information, synthetically. Fuzzy systems take an important role in the research related. In a fuzzy system we can deal with linguistic information by developing a family of fuzzy inference rules such as 'IF...THEN...'. And data information constitutes the external conditions that may adjust system parameters, including the membership functions of fuzzy sets and defuzzification etc, rationally. Using fuzzy inference networks we may represent a fuzzy system as the I/O relationship of a neural system, and therefore fuzzy systems also possess the function of self-learning and self-improving.

Since recent twenty years, fuzzy systems and fuzzy inference networks have attracted much attention for they have found useful in many applied fields such as pattern recognition [30, 33, 56], system modelling and identification [16, 24], automatic control [30, 53, 61], signal processing [12, 35, 60], data compression [47] and telecommunication [42] and so on. As in the research of neural networks we study the applications of fuzzy systems and fuzzy inference networks by taking their universal approximation as a start point. Therefore, in the following let us take the research on approximating capability of fuzzy systems and fuzzy inference networks as a thread to present a survey to theory and application of this two classes of systems.

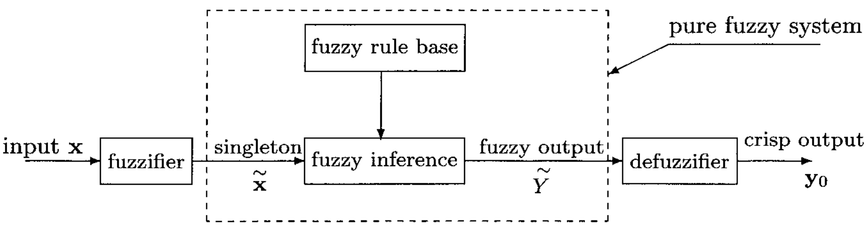


Figure 1.2 Fuzzy system architecture

1.4.1 Fuzzy systems

In practice there are common three classes of fuzzy systems [30, 61], they are pure fuzzy systems, Mamdani fuzzy systems and Takagi–Sugeno (T–S) fuzzy systems. Pure fuzzy systems deal mainly with linguistic information while the latter two fuzzy systems can handle both linguistic information and data information [61]. We can distinguish a Mamdani fuzzy system and a T–S fuzzy system by their inference rule consequents. The rule consequent forms of a Mamdani fuzzy system are fuzzy sets while ones corresponding to a T–S fuzzy system are functions of the system input variables. As shown in

Figure 1.2 is the typical architecture of a fuzzy system, which consists of three parts: fuzzifizer, pure fuzzy system and defuzzifizer. The internal structures of the pure fuzzy system are determined by a sequence of fuzzy inference rules. Suppose the fuzzy rule base is composed of N fuzzy rules $R_1, ..., R_N$. For a given input vector $\mathbf{x}$, by fuzzifizer we can get a singleton fuzzy set $\widetilde{\mathbf{x}}$. Using the fuzzy rule R_j and the implication relation we can establish a fuzzy set $\widetilde{Y}_j$ defined on the output space [30]. By a t−conorm S (generally is chosen as $S = \vee$) we synthesize $\widetilde{Y}_1, ..., \widetilde{Y}_N$ to determine the fuzzy set $\widetilde{Y}$ defined on the output space:

$$\widetilde{Y}(\mathbf{y}) = S\big(\widetilde{Y}_1(\mathbf{y}),\ S\big(\widetilde{Y}_2(\mathbf{y}), ..., S\big(\widetilde{Y}_N(\mathbf{y})\big)\cdots\big)\big).$$

We call $\widetilde{Y}$ a synthesizing fuzzy set [30]. Finally we utilize the defuzzifizer D_e to establish the crisp output $\mathbf{y}_0 = D_e\big(\widetilde{Y}\big)$.

As one of main subjects related to fuzzy systems, universal approximation has attracted much attention since the early 1990s [61, 62, 65, 72]. we can classify the achievements in the field into two classes. One belongs to existential results, that is, the existence of the fuzzy systems is shown by the Stone-Weierstrass Theorem [30, 61]. Such a approach may answer the existence problem of fuzzy systems under certain conditions. However its drawbacks are obvious, since it can not deal with many importantly practical problems such as, how can the approximating procedure of fuzzy systems express the given I/O relationship? How is the accuracy related estimated? With the given accuracy how can the size of the fuzzy rule base of the corresponding fuzzy system be calculated? and so on. Moreover, such a way gives the strong restrictions to the antecedent fuzzy sets, the inference composition rules and defuzzification. That is, the fuzzy sets are Gaussian type fuzzy numbers, the compositions are based on '$\sum - \times$' or '$\vee - \times$', and the defuzzifier usually means the method of center of gravity. Another is the constructive proving method, that is, we may directly build the approximating fuzzy systems related by the constructive procedures. Recent years the research related has attracted many scholars' attention. Ying et al in [65] employ a general defuzzification [23] to generalize Mamdani fuzzy systems and T–S fuzzy systems, respectively. Moreover, the antecedent fuzzy sets and the composition fuzzy operators can be general, that is, the fuzzy sets may be chosen as general fuzzy numbers with certain ranking order and the composition may be '$\vee - T$', where T is a t−norm. And some necessary conditions for fuzzy system approximation and their comparison are built. Zeng et al in [72] propose some accuracy analysis methods for fuzzy system approximation, and an approximating fuzzy system with the given accuracy may be established accordingly. So the constructive methods can be more efficient and applicable.

Up to the present the research on the related problems focuses on the approximations of the fuzzy systems to the continuous functions and the realization of such approximations. Although the related achievements are of much real significance, their application areas are definitely restricted. There are many important and fundamental problems in the field remain to be solved.

First, in addition to continuous functions, how about the universal approximation of fuzzy systems to other general functions? For instance, in the control processes to many nonlinear optimal control models and the pulse circuits, the related systems are non-continuous, but integrable. Therefore the research in which the fuzzy systems are generalized within a general framework and more general functions, including integrable functions are approximately represented by the general fuzzy systems with arbitrary degree of accuracy, are very important both in theory and in practice.

Another problem is 'Rule explosion' phenomenon that is caused by so called 'curse of dimensionality', meaning that in a fuzzy system the number of fuzzy rules may exponentially increases as the number of the input variables of the system increases. Although the fuzzy system research has attracted many scholars' attention, and the achievements related have been successfully applied to many practical areas, particularly to the fuzzy control, the applications are usually limited to systems with very few variables, for example, two or at most four input variables [62]. When we increase the input variables, the scale of the rule base of the fuzzy system is immediately becoming overmuch, consequently the system not implementable. So 'rule explosion' do seriously hinder the applications of the fuzzy systems.

To overcome above drawbacks, Raju et al defined in [53] a new type of fuzzy system, that is the hierarchical fuzzy system. Such a system is constructed by a series of lower dimensional fuzzy systems, which are linked in a hierarchical fashion. To realize the given fuzzy inferences, the number of fuzzy rules needed in the hierarchical system is the linear function of the number of the input variables. Thus, we may avoid the 'rule explosion'. Naturally we may put forward an important problem, that is, how may the representation capability of the hierarchical fuzzy systems be analyzed? Kikuchi et al in [31] show that it is impossible to give the precise expression of arbitrarily given continuous function by a hierarchical fuzzy system. So we have to analyze the approximation capability of hierarchical fuzzy systems, i.e. whether are hierarchical fuzzy systems universal approximator or not? If a function is continuously differentiable on the whole space, Wang in [62] shows the arbitrarily close approximation of the function by hierarchical fuzzy systems; and he also in [63] gave the sensitivity properties of hierarchical fuzzy systems and designed a suitable system structure. For each compact set U and the arbitrarily continuous, or integrable function f on U, how may we find a hierarchical fuzzy system to approximate f uniformly with arbitrary error bounds ε?

The third important problem is the fuzzy system approximations in stochastic environment. Recently the research on the properties of the artificial neural networks in the stochastic environment attracts many scholars' attention. The approximation capabilities of a class of neural networks to stochastic processes and the problem whether the neural networks are able to learn stochastic processes are systematically studied. It is shown that the approximation identity neural networks can with mean square sense approximate a class of stochastic processes to arbitrary degree of accuracy. The fuzzy systems can simultaneously deal with data information and linguistic information. So it is undoubtedly very important to study the approximation in the stochastic environment, that is, *the approximation capabilities of fuzzy systems to stochastic processes*.

The final problem is to estimate the size of the rule base of the approximating fuzzy system for the given accuracy. The research related is the basis for constructing the related fuzzy systems.

The systematic study on above problems constitutes the central parts of Chapter VI and Chapter VII. Also many well-designed simulation examples illustrate our results.

1.4.2 Fuzzy inference networks

Fuzzy inference system can simulate and realize natural language and logic inference mechanic. A fuzzy inference network is a multilayer feedforward network, by which a fuzzy system can be expressed as the I/O relationship of a neural system. So a fuzzy system and its corresponding fuzzy inference network are functionally equivalent [30]. As a organic fusion of inference system and neural network, a fuzzy inference network can realize automobile generation and automobile matching of fuzzy rules. Further, it can adaptively adjust to adapt itself to the changes of conditions and to self-improve. Since the early 1990s, many achievements have been achieved and they have found useful in many applied areas, such as system modeling [15, 16], system identification [46, 56], pattern recognition [58] and system forecasting [45] and so on. Moreover, fuzzy inference networks can deal with all kinds of information including linguistic information and data information, efficiently since they possess adaptiveness and fault-tolerance. Thus, they can successfully be applied to noise image processing, boundary detection for noise images, classification and detection of noise, system modeling in noise environment and so on.

Theoretically the research on fuzzy inference networks focuses mainly on three parts: First, design a feedforward neural network to realize a known fuzzy system, so that the network architecture is as simple as possible [30]; Second, build some suitable learning algorithms, so that the connection weights are adjusted rationally to establish suitable antecedent and consequent fuzzy sets; Third, within a general framework study the fuzzy inference networks, which is based on some general defuzzifier [54]. Defuzzification constitutes one important object to study fuzzy inference networks and it attracts many scholars' attention. There are mainly four defuzzification methods, they are center of gravity (COG) method [33, 54], maximum of mean(MOM) method [61], $\alpha-$cut integral method, and $p-$mean method [55]. In addition, many novel defuzzifications to the special subjects are put forward in recent years. They have respective starting point and applying fields, also they have themselves advantages and disadvantages. For example, the COG method synthesizes all actions of points in the support of the synthesizing fuzzy set to establish a crisp output while some special functions of some particular points e.g. the points with maximum membership, are neglected. The MOM method takes only the points with maximum membership under consideration while the other points are left out of consideration. The $\alpha-$cut integral and $p-$mean methods are two other mean summing forms for all points in the support of the synthesizing set. The research on how we can define defuzzifications and fuzzy inference networks within a general framework attracts much attention. Up to now many general defuzzifiers have been put forward [23, 54, 55, 61]. However, they possess respective drawbacks: Either the general principles are too many to be applied conveniently or, the definitions are too concrete to be generalized to general cases.

To introduce some general principle for defuzzidication and to build a class of generalized fuzzy inference network within a general framework constitute a preliminary to study FNN application in image processing in Chapter VIII.

§1.5 Fuzzy techniques in image restoration

The objective of image restoration is to reconstruct the image from degraded one resulted from system errors and noises and so on. There are two ways to achieve such an objective [3, 52]. One is to model the corrupted image degraded by motion, system distortion, and additive noises, whose statistic models are known. And the inverse process may be applied to restore the degraded images. Another is called image enhancement, that is, constructing digital filters to remove noises to restore the corrupted images resulted from noises. Originally, image restoration included the subjects related to the first way only. Recently many scholars put the second way into the field of image restoration [12, 35, 60]. Linear filter theory is an efficient tool to process additive Gaussian noise, but it can not deal with non-additive Gaussian noise. So the research on nonlinear filters has been attracting many scholars' attention [3, 60, 66].

In practice, it is imperative to bring ambiguity and uncertainty in the acquisition or transmission of digital images. One may use human knowledge expressed heuristically in natural language to describe such images. But this approach is highly nonlinear in nature and can not be characterized by traditional mathematical modeling. The fuzzy set and fuzzy logic can be efficiently incorporated to do that. So it is convenient to employ fuzzy techniques in image processing. The related discussions appeared about in 1981 [48, 49], but not until 1994 did the systematic results related incurred. Recently fuzzy techniques are efficiently applied in the field of image restoration, especially in the filtering theory to remove system distortion and impulse noises, smooth non-impulse noises and enhance edges or other salient features of the image.

1.5.1 Crisp nonlinear filters

Rank selection (RS) filter is a useful nonlinear filtering model whose simplest form is median filter [52]. The guidance for building all kinds of RS type filters is that removing impulsive noise while keeping the fine image structure. By median filter, impulsive type noise can be suppressed, but it removes fine image structures, simultaneously. When the noise probability exceeds 0.5, median filter can result in poor filtering performance. To offer improved performance, many generalizations of median filter have been developed. They include weighted order statistic filter, center weighted median filter, rank conditioned rank selection (RCRS) filter, permutation filter, and stack filter, etc (see [3, 52]). The RCRS filter is built by introducing feature vector and rank selection operator. It synthesizes all advantages of RS type filters, also it can be generalized as the neural network filter. Moreover, as a signal restoration model, the RCRS filter possesses the advantage of utilizing rank condition and selection feature of sample set simultaneously. Thus all RS type filters may be handled within a general framework [3]. However, although RS type filters improve median filter from different aspects, their own shortcomings are not overcome, for the outputs of all these filters are the observation samples in the operating window of the image. For example a RCRS filter may change the fine image structure while removing impulsive noise; when the noise probability $p > 0.5$ it is difficult to get a restoration with good performance; the complexity of the RCRS filter increases exponentially with the order (the length of operating window). Such facts have spurred the development of fuzzy filters, which improve the performance of RS filters by extending output range, soft decision and adaptive structure.

1.5.2 Fuzzy filters

The RCRS filter can not overcome the drawbacks of median filter thoroughly since its ultimate output is still chosen from the gray levels in the operating window. So fuzzy techniques may be used to improve the RS type filters from the following parts: extending output range, soft decision and fuzzy inference structure. Recent years fuzzy theory as a soft technique has been successfully applied in modeling degraded image and building noise filters.

Extending output range means generalizing crisp filters within a fuzzy framework. For example, by fuzzifying the selection function as a fuzzy rank, the RCRS filter can be generalized a new version—rank conditioned fuzzy selection (RCFS) filter [12]. It utilize natural language, such as 'Dark' 'Darker' 'Medium' 'Brighter' 'Bright' and so on to describe gray levels of the image related. And so the image information may be used more efficiently. Although the RCFS filter improves the performance of RCRS filter, as well as the filtering capability, the problems similar to RCRS filter arise still. So in more cases, soft decision or fuzzy inference structure are used to improve noise filters. Soft decision means that we may use fuzzy set theory to soften the constraint conditions for the digital image and to build the image restoration techniques. Civanlar et al in [17] firstly establish an efficient image restoration model by soft decision, in which the key part is to define suitable membership functions of fuzzy sets related. Yu and Chen in [66] generalize the stack filter as a fuzzy stack filter, by which the filtering performance is much improved. One of key steps to do that is fuzzifying a positive Boolean function (PBF) as a fuzzy PBF, by which we may estimate a PBF from the upper and the lower, respectively. And so the fuzzy stack filter concludes the stack filter as a special case. Fuzzy inference structure for image processing means that some fuzzy inference rules are built to describe the images to be processed, and then some FNN mechanisms are constructed to design noise filters. An obvious advantage for such an approach is that the fuzzy rules may be adjusted adaptively. The performance of the filters related may be advantageous in processing the high probability ($p > 0.5$) noise images [35, 60]. A central part of Chapter VIII is to build some optimal FNN filters by developing suitable fuzzy inference rules and fuzzy inference networks.

Furthermore in [47], the fault-tolerance of fuzzy relational equations is the tool for image compression and reconstruction; And the classical vector median filter is generalized to the fuzzy one, and so on. Of course, the research of image restoration by fuzzy techniques has been in its infancy period, many elementary problems related are unsolved. Also to construct the systematic theory in the field is a main object for future research related to the subject.

§1.6 Notations and preliminaries

In the following let us present the main notations and terminologies used in the book, and account for the organization of the book.

Suppose $\mathbb{N}$ is the natural number set, and $\mathbb{Z}$ is the integer set. Let $\mathbb{R}^d$ be $d-$dimensional Euclidean space, in which $\|\cdot\|$ means the Euclidean norm. $\mathbb{R} \triangleq \mathbb{R}^1$, and $\mathbb{R}_+$ is the collection of all nonnegative real numbers. If $x \in \mathbb{R}$, $\text{Int}(x)$ means the maximum integer not exceeding x.

By $A, B, C, \ldots$ we denote the subsets of $\mathbb{R}$, and $\overline{A}$ is the closure of A. For $A, B \subset \mathbb{R}^d$, let $d_H(A, B)$ be Hausdorff metric between A and B, i.e.

$$d_H(A, B) = \max\Big\{ \bigvee_{\mathbf{x}\in A} \bigwedge_{\mathbf{y}\in B} \{\|\mathbf{x}-\mathbf{y}\|\}, \bigvee_{\mathbf{y}\in B} \bigwedge_{\mathbf{x}\in A} \{\|\mathbf{x}-\mathbf{y}\|\}\Big\}, \tag{1.3}$$

where '$\vee$' means the supremum operator 'sup', and '$\wedge$' means the infimum operator 'inf'. For the intervals $[a, b]$, $[c, d] \subset \mathbb{R}$, define the metric $d_E([a, b], [c, d])$ as follows:

$$d_E\big([a, b], [c, d]\big) = \big\{(a-c)^2 + (b-d)^2\big\}^{\frac{1}{2}}. \tag{1.4}$$

Give the intervals $[a, b]$, $[c, d] \subset \mathbb{R}$, it is easy to show

$$d_H\big([a, b], [c, d]\big) \le d_E\big([a, b], [c, d]\big) \le \sqrt{2}\cdot d_H\big([a, b], [c, d]\big), \tag{1.5}$$

that is, the metrics d_E and d_H are equivalent. If X is universe, by $\mathcal{F}(X)$ we denote the collection of all fuzzy sets defined on X. Using $\underset{\sim}{A}$, $\underset{\sim}{B}, \ldots$ we denote the fuzzy sets defined on $\mathbb{R}$. And $\mathcal{F}_0(\mathbb{R})$ means a subset of $\mathcal{F}(\mathbb{R})$ with the following conditions holding, i.e. for $\widetilde{A}\in \mathcal{F}_0(\mathbb{R})$, we have

(i) The kernel of $\widetilde{A}$ satisfies, $\text{Ker}(\widetilde{A}) \triangleq \{x \in \mathbb{R} \,|\, \widetilde{A}(x) = 1\} \neq \emptyset$;

(ii) $\forall \alpha \in (0,1]$, then $\widetilde{A}_\alpha \triangleq [a^1_\alpha,\ a^2_\alpha]$ is a bounded and closed interval;

(iii) The support of $\widetilde{A}$ satisfies, $\text{Supp}(\widetilde{A}) \triangleq \overline{\{x \in \mathbb{R} \,|\, \widetilde{A}(x) > 0\}}$ is a bounded and closed set of $\mathbb{R}$.

We denote the support $\text{Supp}(\widetilde{A})$ of a fuzzy set $\widetilde{A}$ by $\widetilde{A}_0$. If $\widetilde{A},\ \widetilde{B}\in \mathcal{F}_0(\mathbb{R})$, define the metric between $\widetilde{A}$ and $\widetilde{B}$ as [19, 20]:

$$D(\widetilde{A},\ \widetilde{B}) = \bigvee_{\alpha\in[0,1]} \{d_H(\widetilde{A}_\alpha,\ \widetilde{B}_\alpha)\} = \bigvee_{\alpha\in[0,1)} \{d_H(\widetilde{A}_{\underset{\bullet}{\alpha}},\ \widetilde{B}_{\underset{\bullet}{\alpha}})\}. \tag{1.6}$$

By [19] it follows that $\big(\mathcal{F}_0(\mathbb{R}), D\big)$ is a complete metric space. If we generalize the condition (ii) as

(ii)′ $\widetilde{A}$ is a convex fuzzy set, that is the following fact holds:

$$\forall x_1,\ x_2 \in \mathbb{R},\ \forall \alpha \in [0,\ 1],\ \widetilde{A}\big(\alpha x_1 + (1-\alpha)x_2\big) \geq \widetilde{A}(x_1) \wedge \widetilde{A}(x_2).$$

Denote the collection of fuzzy sets satisfying (i)(ii)′ and (iii) as $\mathcal{F}_c(\mathbb{R})$. If $\widetilde{A}\in \mathcal{F}_c(\mathbb{R})$, then $\widetilde{A}$ is called a bounded fuzzy number. Obviously, $\mathcal{F}_0(\mathbb{R}) \subset \mathcal{F}_c(\mathbb{R})$. Also it is easy to show, (ii)′ is equivalent to the fact that $\forall \alpha \in [0,\ 1]$, $\widetilde{A}_\alpha \subset \mathbb{R}$ is a interval. Denote

$$\mathcal{F}_0(\mathbb{R})^d = \underbrace{\mathcal{F}_0(\mathbb{R}) \times \cdots \times \mathcal{F}_0(\mathbb{R})}_{d}.$$

And for $(\widetilde{A}_1, ..., \widetilde{A}_d),\ (\widetilde{B}_1, ..., \widetilde{B}_d) \in \mathcal{F}_0(\mathbb{R})^d$, we also denote for simplicity that

$$D\big((\widetilde{A}_1, ...\ \widetilde{A}_d),\ (\widetilde{B}_1, ..., \widetilde{B}_d)\big) = \sum_{i=1}^{n} D(\widetilde{A}_i, \widetilde{B}_i). \tag{1.7}$$

It is easy to show, $(\mathcal{F}_0(\mathbb{R})^d,\ D)$ is also a complete metric space. For $\widetilde{A}\in \mathcal{F}_0(\mathbb{R})$, $|\widetilde{A}|$ means $D(\widetilde{A}, \{0\})$, that is

$$|\widetilde{A}| = \bigvee_{\alpha\in[0,1]} \{|a^1_\alpha| \vee |a^2_\alpha|\}\ \big(\widetilde{A}_\alpha = [a^1_\alpha,\ a^2_\alpha]\big).$$

For a given function $f : \mathbb{R}^d \longrightarrow \mathbb{R}$, we may extend f as $\widetilde{f}$: $\mathcal{F}_0(\mathbb{R})^d \longrightarrow \mathcal{F}(\mathbb{R})$ by the extension principle [44]:

$$\forall (\widetilde{A}_1, ..., \widetilde{A}_d) \in \mathcal{F}_0(\mathbb{R})^d,\ \widetilde{f}(\widetilde{A}_1, ..., \widetilde{A}_d)(y) = \bigvee_{f(x_1,\cdots,x_d)=y} \Big\{\bigwedge_{i=1}^{d} \{\widetilde{A}_i(x_i)\}\Big\}. \tag{1.8}$$

For simplicity, we write also $\tilde{f}$ as f. And f is called an extended function.

$C^1(\mathbb{R})$ is the collection of all continuously differentiable functions on $\mathbb{R}$; and $C^1([a,b])$ is the set of all continuously differentiable functions on the closed interval $[a, b]$.

Definition 1.1 [14] Suppose $g: \mathbb{R} \longrightarrow \mathbb{R}$, and $F_N: \mathbb{R}^d \longrightarrow \mathbb{R}$ is a three layer feedforward neural network whose transfer function is g. That is

$$\forall (x_1, ..., x_d) \in \mathbb{R}^d,\ F_N(x_1, ..., x_d) = \sum_{j=1}^{p} v_j \cdot g\Big(\sum_{i=1}^{d} w_{ij} \cdot x_i + \theta_j\Big).$$

If $F_N(\cdot)$ constitute a universal approximator, then g is called a Tauber-Wiener function.

If g is a generalized sigmoidal function $\sigma: \mathbb{R} \longrightarrow \mathbb{R}$, that is, σ is bounded, and $\lim\limits_{x \to +\infty} \sigma(x) = 1$, $\lim\limits_{x \to -\infty} \sigma(x) = 0$. Then by [14], it follows that g is a Tauber-Wiener function.

We call $g: \mathbb{R} \longrightarrow \mathbb{R}$ a continuous sigmoidal function, if g is continuous and increasing, moreover, $\lim\limits_{x \to +\infty} \sigma(x) = 1$, $\lim\limits_{x \to -\infty} \sigma(x) = 0$. Obviously, a continuous sigmoidal function is a Tauber-Wiener function.

Let μ be a Lebesgue measure on $\mathbb{R}^d$, and $f: \mathbb{R}^d \longrightarrow \mathbb{R}$ be a measurable function. Give $p \in [1, +\infty)$. If f is a $p-$integrable function on $\mathbb{R}^d$, define the $L_p(\mu)-$norm of f as follows:

$$\|f\|_{\mu,p} = \Big\{\int_{\mathbb{R}^d} |f(x)|^p \mathrm{d}\mu\Big\}^{\frac{1}{p}}.$$

If $A \subset \mathbb{R}^d$, and μ is Lebesgue measure on A, we let

$$\|f\|_{A,p} = \Big\{\int_{A} |f(x)|^p \mathrm{d}\mu\Big\}^{\frac{1}{p}},$$

$$L^p(\mathbb{R}, \mathcal{B}, \mu) \stackrel{\triangle}{=} L_p(\mu) = \{f: \mathbb{R}^d \longrightarrow \mathbb{R} |\ \|f\|_{\mu,p} < +\infty\},$$

$$L^p(A) = \{f: \mathbb{R}^d \longrightarrow \mathbb{R} |\ \|f\|_{A,p} < +\infty\},$$

where $\mathcal{B}$ is a $\sigma-$algebra on $\mathbb{R}$. And $\mathcal{C}_F$ is a sub-class of collection of continuous fuzzy functions that $\mathcal{F}_0(\mathbb{R})^d \longrightarrow \mathcal{F}_0(\mathbb{R})$. For $n, m \in \mathbb{N}$, by $\mu_{n\times m}$ we denote the collection of all fuzzy matrices with n rows and m columns. For $\mathbf{x}^1 = (x_1^1, ..., x_n^1)$, $\mathbf{x}^2 = (x_1^2, ..., x_n^2) \in [0,1]^n$, we denote

$$\mathbf{x}^1 \vee \mathbf{x}^2 = \big(x_1^1 \vee x_1^2, ..., x_n^1 \vee x_n^2\big),$$

$$\mathbf{x}^1 \wedge \mathbf{x}^2 = \big(x_1^1 \wedge x_1^2, ..., x_n^1 \wedge x_n^2\big).$$

Other terminologies and notations not being emphasized here, the readers may find them in the respective chapters, or sections, in which they are utilized.

§1.7 Outline of the topics of the chapters

The book tries to develop FNN theory through three main types of FNN models. They are FNN's based on fuzzy operators which are respectively treated in Chapter II and Chapter III; Fuzzified neural networks taking regular FNN's and polygonal FNN's as main components, which are dealt with by Chapter IV and Chapter V, respectively; Fuzzy inference networks being able to realize the common fuzzy systems, such as Mamdani fuzzy systems, T–S fuzzy systems and stochastic fuzzy systems and so on, which are handled in Chapter VI, Chapter VII and Chapter VIII. In each chapter we take some simulation examples to illustrate the effectiveness of our results, especially the FNN models and the learning algorithms related.

Chapter II treats two classes of FNN models—feedforward fuzzy associative memory (FAM) for storing fuzzy patterns, which can also recall right fuzzy patterns stored, and fuzzy adaptive resonance theory (ART) for classifying fuzzy patterns. A fuzzy pattern related can be expressed as a vector whose components belong to [0, 1]. To improve the storage capability of a FAM, we in §2.1 build a novel feedforward FNN—FAM with threshold, that is, introduce a threshold to each neural unit in a FAM. Some equivalent conditions that a given family of fuzzy pattern pairs can be stored in the FAM completely are established. Moreover, an analytic learning algorithm for connection weights and thresholds, which guarantees the FAM to store the given fuzzy pattern pair family is built. To take advantage of the adaptivity of neural systems we build two classes of iteration learning algorithms for the FAM, which are called the fuzzy delta algorithm and the fuzzy BP algorithm in §2.2 and §2.3, respectively. §2.4 focuses on a fuzzy classifying network—fuzzy ART. After recalling some fundamental concepts of ART1 we define a fuzzy version of ART1 through fuzzy operators '$\vee$' and '$\wedge$'. We characterize the classifying procedure of the fuzzy ART and develop some useful properties about how a fuzzy pattern is classified. Finally, corresponding a crisp ARTMAP we propose its fuzzy version—fuzzy ARTMAP by joining two fuzzy ART's together. Many simulation examples are studied in detail to illustrate our conclusions.

Chapter III deals with another type of FNN's based on fuzzy operators '$\vee$' and '$\wedge$'— feedback FAM's, which are dynamic FNN's. We focus on two classes of dynamic FAM's, they are fuzzy Hopfield networks and fuzzy bidirectional associative memories (FBAM's). §3.1 reports many useful dynamic properties of the fuzzy Hopfield networks by studying attractors and attractive basins. And based on fault-tolerance we develop an analytic learning algorithm, by which some correct fuzzy patterns may be recalled through some imprecise inputs. To improve the storage capability and fault-tolerance the fuzzy Hopfield networks with threshold are reported in §3.2. It is also shown that the dynamical systems are uniformly stable and their attractors are Lyapounov stable. In §3.3 and §3.4 the corresponding problems for FBAM's are analyzed, systematically. At first we show the fact that the FBAM's converge their equilibrium stables, i.e. attractors or limit cycles. And then some learning algorithms based on fault-tolerance are built. Many simulation examples are shown to illustrate our conclusions. The transitive laws of attractors, the discrimination of the pseudo-attractors of this two dynamical FNN's are presented in §3.5 and §3.6, respectively. The basic tools to do these include connection networks, fuzzy row-restricted matrices and elementary memories and so on.

Chapter IV develops the systematic theory of regular FNN's by focusing mainly on two classes of important problems, they are learning algorithms for the fuzzy weights of regular FNN's and approximating capability, i.e. universal approximation of regular FNN's to fuzzy functions. To this end, we at first introduce regular fuzzy neurons, and present their some useful properties. Then we define regular FNN's by connecting a group of regular fuzzy neurons. Here a regular FNN means mainly a multi-layer feedforward FNN. And give some results about the I/O relationships of regular FNN's. Buckley's conjecture 'the regular FNN's can be universal approximators of the continuous and increasing fuzzy function class' is proved to be false by a counterexample. However it can be proven that regular FNN's can approximate the extended function of any continuous function with arbitrarily given degree of accuracy on any compact set of $\mathcal{F}_c(\mathbb{R})$. In §4.3 we introduce a novel error function related to three layer feedforward regular FNN's and develop a fuzzy BP algorithm for the fuzzy weights. The basic tools to do that are the $\vee - \wedge$ function and the polygonal fuzzy numbers. Using the fuzzy BP algorithm we can employ a three layer regular FNN to realize a family of fuzzy inference rules approximately. To speed the convergence of the fuzzy BP algorithm, §4.4 develops a fuzzy CG algorithm for the fuzzy weights of the three layer regular FNN's, whose learning constant in each iteration is determined by GA. it is also shown in theory that the fuzzy CG algorithm is convergent to the minimum point of the error function. Simulation examples also demonstrate the fact that the fuzzy CG algorithm improves indeed the fuzzy BP algorithm in convergent speed. In §4.5 we take the fuzzy Bernstein polynomial as a bride to show that the four layer feedforward regular FNN's can be approximators to the continuous fuzzy valued function class. The realization steps of the approximating procedure are presented and illustrated by a simulation example. Taking these facts as the basis we in §4.6 develop some equivalent conditions for the fuzzy function class $\mathcal{C}_F$, which can guarantee universal approximation of four layer regular FNN's to hold. Moreover, an improved fuzzy BP algorithm is developed to realize the approximation with a given accuracy. Thus, the universal approximation problem for four layer regular FNN's is solved completely. Finally in the chapter we in §4.7 employ a regular FNN to represent integrable bounded fuzzy valued functions, approximately with integral norm sense.

In Chapter V we proceed to analyze universal approximation of regular FNN's. The main problem to solve is to simplify the equivalent conditions of the fuzzy function class $\mathcal{C}_F$ in Chapter IV, which can ensure universal approximation of four layer regular FNN's. The main contributions are to introduce a novel class of FNN models—polygonal FNN's and to present useful properties of the FNN's, such as topological architecture, internal operations, I/O relationship analysis, approximation capability and learning algorithm and so on. To this end we in §5.1 at first develop uniformity analysis for three layer, and four layer crisp feedforward neural networks, respectively. For a given function family the crisp neural networks can approximate each function uniformly with a given accuracy. Also we can construct the approximating neural networks directly through the function family. §5.2 reports the topological and analytic properties of the polygonal fuzzy number space $\mathcal{F}_{bc}^{tn}(\mathbb{R})$, for instance, the space is a completely separable metric space; also it is locally compact; a subset in the space is compact if and only if the set is bounded and closed; a bounded fuzzy number can be a limit of a sequence of polygonal fuzzy numbers, and so on. Moreover, Zadeh's extension principle is improved in $\mathcal{F}_{bc}^{tn}(\mathbb{R})$, by developing a novel extension principle and fuzzy arithmetic. Thus, many extended operations such as extended multiplication and extended division and so on can be simplified strikingly. Based on the novel extension principle §5.3 defines the polygonal FNN, which is a three layer feedforward network with polygonal fuzzy number input, output and connection weights. Similarly with §4.3 a fuzzy BP algorithm for fuzzy weights of the polygonal FNN's is developed and it is successfully applied to the approximate realization of fuzzy inference rules. §5.4 treats universal approximation of the polygonal FNN's, and shows the fact that a fuzzy function class can guarantee universal approximation of the polygonal FNN's if and only if each fuzzy function in this class is increasing, which simplifies the corresponding conditions in §4.6, strikingly. So the polygonal FNN's are more applicable.

Chapter VI deals mainly with the approximation capability of generalized fuzzy systems with integral norm. The basic tool to do that is the piecewise linear function that is one central part in §6.1. Also a few of approximation theorems for the piecewise linear functions expressing each $L_p(\mu)-$integrable function are established. In §6.2 we define the generalized fuzzy systems which include generalized Mamdani fuzzy systems and generalized T–S fuzzy systems as special cases. And show the universal approximation of the generalized fuzzy systems to $L_p(\mu)-$integrable functions with integral norm sense; For a given accuracy $\varepsilon > 0$, a upper bound of the size of fuzzy rule base of a corresponding approximating fuzzy system is estimated. One main impediment to hinder the application of fuzzy systems is 'rule explosion' problem, that is, the size of the fuzzy rule base of a fuzzy system increases exponentially as the input space dimensionality increasing. To overcome such an obstacle we in §6.3 employ the hierarchy introduced by Raju et al to define hierarchical fuzzy systems, by which the 'rule explosion' problem can be solved successfully. Moreover, a hierarchical fuzzy system and the corresponding higher dimension fuzzy system are equivalent. So the hierarchical fuzzy systems can be universal approximators with maximum norm and with integral norm respectively, on which we main focus in §6.4. Thus, the fuzzy systems can also applied to the cases of higher dimension complicated system. Many simulation examples are presented to illustrate the approximating results in the chapter.

Some further subjects about approximation of fuzzy systems are studied in Chapter VII, that is, we discuss the approximation capability of fuzzy systems in stochastic environment. To this end we in §7.1 recall some basic concepts about stochastic analysis, for instance, stochastic integral, stochastic measure and canonical representation of a stochastic process and so on. §7.2 introduces two class of stochastic fuzzy systems, they are stochastic Mamdani fuzzy systems and stochastic T–S fuzzy systems, which possess many useful properties. For example, their stochastic integrals with respect to an orthogonal incremental process exist, and the stochastic integrals can expressed approximately as an algebra summation of a sequence of random variables. Using the fundamental results in §6.2 the systematic analysis of approximating capability of stochastic fuzzy systems including stochastic Mamdani fuzzy systems and stochastic T–S fuzzy systems with mean square sense is presented, which is central part in §7.3 and §7.4, respectively. Learning algorithms for stochastic Mamdani fuzzy systems and stochastic T–S fuzzy systems are also developed, and approximating realization procedure of some stochastic processes including a class of non-stationary processes by stochastic fuzzy systems are demonstrated by some simulation examples.

Chapter VIII focuses mainly on the application of FNN's in image restoration. At first we treat fuzzy inference networks within a general framework, and so §8.1 introduces a general fuzzy inference network model by define generalized defuzzifier, in which includes the common fuzzy inference networks as special cases. In theory the generalized fuzzy inference networks can be universal approximators, which provides us with the theoretic basis for the applications of generalized fuzzy inference networks. In dynamical system identification we demonstrate by some real examples that the performance resulting from generalized fuzzy inference networks is much better than that from crisp neural networks or fuzzy systems with the Gaussian type antecedent fuzzy sets. In §8.2 we propose the FNN representations of a 2-D digital image by define the deviation fuzzy sets and coding the image as the connection weights of a fuzzy inference network. Such a representation is accurate when the image is uncorrupted, and the image can be completely reconstructed; when the image is corrupted, the representation may smooth noises and serve as a filter. Based on the minimum absolute error (MAE) criterion we design an optimal filter FR, whose filtering performance is much better than that of median filter. FR can preserve the uncorrupted structure of the image and remove impulse noise, simultaneously. However, when the noise probability exceeds 0.5, i.e. $p > 0.5$ FR may result in bad filtering performance. In order to improve FR in high probability noise, §8.3 develops a novel FNN—selection type FNN, which can be a universal approximator and is suitable for the design of noise filters. Also based on MAE criterion, the antecedent fuzzy sets of the selection type FNN are adjusted rationally, and an optimal FNN filter is built. It preserves the uncorrupted structures of the image as many as possible, and also to a greatest extend it removes impulse noise. So by the FNN filter the restoration image with high quality may be built from the corrupted image degraded by high or low probabilities impulse noises. Further, the FNN filter also can suppress some hybrid type noises. By many real examples we demonstrate that the restoration images with good performances can be obtained through the filter FR, or the FNN filter. Especially the filtering performances of FNN filters to restore high probability ($p > 0.5$) noise images may be much better than that of RS type filters, including RCRS filter.

References

[1] Alefeld G. & Herzberger J., *Introduction to interval computations,* New York: Academic Press, 1983.

[2] Aliev R. A., Genetic algorithm-based learning of fuzzy neural network, Part 1: feed-forward fuzzy neural networks, *Fuzzy Sets and Systems,* 2001, **118**: 351–358.

[3] Astola J. & Kuosmanen P., *Fundamentals of Nonlinear Digital Filtering,* Boca Raton FL: CRC, 1997.

[4] Blanco A., Delgado M. & Requena I., Identification of relational equations by fuzzy neural networks, *Fuzzy Sets and Systems,* 1995, **71**: 215–226.

[5] Blanco A., Delgado M. & Requena I., Improved fuzzy neural networks for solving relational equations, *Fuzzy Sets and Systems,* 1995, **72**: 311–322.

[6] Buckley J. J. & Hayashi Y., Can fuzzy neural nets approximate continuous fuzzy functions, *Fuzzy Sets and Systems,* 1994, **61**(1): 43–51.

[7] Cai K. Y., *Introduction to Fuzzy Reliability,* Boston•Dordrecht•London: Kluwer Academic Publishers, 1996.

[8] Carpenter G. A. & Grossberg S., A massively parallel architecture for a self-organizing neural pattern recognition, *Computer Vision,Graphics, and Image Processing,* 1987, **37**: 54–115.

[9] Carpenter G. A. & Grossberg S., ART2: Stable self-organization of pattern recognition codes for analog input patterns, *Applied Optics,* 1987, **26**: 4919–4930.

[10] Carpenter G. A. & Grossberg S., ART3: Hierarchical search using chemical transmitters in self-organizing pattern recognition architectures, *Neural Networks,* 1991, **3**: 129–152.

[11] Carpenter G. A., Grossberg S. & Rosen D. B., Fuzzy ART: an adaptive resonance algorithm for rapid, stable classification of analog patterns (Tech. Rep. CAS/CNS-91-006). *Proc. of Internat. Joint Conf. on Neural Networks,* 1991, **2**: 411–416.

[12] Chen R. C. & Yu P. T., Fuzzy selection filters for image restoration with neural learning, *IEEE Trans. on Signal Processing,* 1999, **47**: 1446–1450.

[13] Chen T. P., Chen H & Liu R. W., Approximation capability in $C(\overline{\mathbb{R}}^n)$ by multilayer feedforward networks and related problems, *IEEE Trans. on Neural Networks,* 1995, **6**(1): 25–30.

[14] Chen T. P., Approximating problems in system identification by neural networks, *Science in China, Series A,* 1994, **37**(4): 414–422.

[15] Chung F. -L. & Lee T., Fuzzy competitive learning, *Neural Networks,* 1994, **7**(3): 539–551.

[16] Chung F. -L. & Duan J. -C, On multistage fuzzy neural network modeling, *IEEE Trans. on Fuzzy Systems,* 2000, **8**: 125–142.

[17] Civanlar M. R. & Trussell H. J., Digital image restoration using fuzzy sets, *IEEE Trans.on Acoust.,Speech,Signal Processing,* 1986, **34**: 919– 936.

[18] Dai Y., Han J. & Liu G., et al, Convergence properties of nonlinear conjugate gradient algorithms, *SIAM J. on Optimization,* 1999, **10**: 345–358.

[19] Diamond P. & Kloeden P., Characterization of compact subsets of fuzzy sets, *Fuzzy Sets and Systems,* 1989, **29**: 341–348.

[20] Diamond P. & Kloeden P., *Metric Spaces of Fuzzy Sets,* Singapore: World Scientific Publishing, 1994.

[21] Dubois D. & Prade H., *Fuzzy sets and systems: theory and applications,* New York: Academic Press, 1981.

[22] Feuring T. & Lippe W. M., The fuzzy neural network approximation lemma, *Fuzzy Sets and Systems,* 1999, **102**(2): 227–237.

[23] Filev D. P. & Yager R. R., A generalized defuzzification method via bad distributions, *Internat. J. of Intelligent Systems,* 1991, **6**: 687–697.

[24] Horikawa S., Furuhashi T. & Uchikawa Y., On fuzzy modeling using fuzzy neural networks with the back-propagation algorithm, *IEEE Trans. on Neural Networks,* 1992, **3**: 801–806.

[25] Haykin S., *Neural networks, a comprehensive foundation,* Now York: IEEE Press, 1994.

[26] Hopfield J. J., Neural networks and physical systems with emergent collective computational abilities, *Proc. Nat. Acad. Sci.,* 1982, **13**: 2254–2258.

[27] Ingman D. & Merlis Y., Maximum entropy signal reconstruction with neural networks, *IEEE Trans. on Neural Networks,* 1992, **3**: 195–201.

[28] Ishibuchi H., Fujioka R. & Tanaka H., Neural networks that learn from fuzzy if–then rules, *IEEE Trans.on Fuzzy Systems,* 1993, **1**: 85–97.

[29] Jain A. K., *Fundamentals of digital image processing,* Englewood Cliffs, NJ: PTR Prentice-Hall, 1989.

[30] Jang J. S. R., Sun C. T. & Mizutani E., *Neoro-fuzzy and soft computing,* Prentice Hall, 1997.

[31] Kikuchi H., Otake A. & Nakanishi S., Functional completeness of hierarchical fuzzy modeling, *Information Sciences,* 1998, **110**: 51–61.

[32] Kosko B., Fuzzy associative memories, in: Kandel A. (Ed.) *Fuzzy expert systems,* Addison–Weley, Reading, MA, 1987.

[33] Kwan H. K. & Cai Y., A fuzzy neural network and its application to pattern recognition, *IEEE Trans. on Fuzzy Systems,* 1994, **2**: 185–193.

[34] Khanna T., *Foundation of neural networks,* Reading, MA: Addison–Wesley, 1990.

[35] Kuo Y. H., Lee C. S. & Chen C. L., High-stability AWFM filter for signal restoration and its hardware design, *Fuzzy Sets and Systems,* 2000, **114**: 185–202.

[36] Lee S. C. & Lee E. T., Fuzzy neural networks, *Math. Biosci.,* 1975, **23**: 151–177.
[37] Lee S. C. & Lee E. T., Fuzzy sets and neural networks, *J. Cybernetics,* 1974, **4**: 83–103.
[38] Li X. Z. & Ruan Da, Novel neural algorithms based on fuzzy δ rules for solving fuzzy relation equations: Part I, II, III, *Fuzzy Sets and Systems,* 1997, **90**: 11–23; 1999, **103**: 473–486; 2000, **109**: 355–362.
[39] Linden A. & Kindermann J., Inversion of multilayer nets, *Proc. Internat. Joint Conf. on Neural Networks,* 1989: 425–430.
[40] Liu Puyin & Li Hongxing, Soft computing and its philosophical intension, *Studies in Dialectics of Nature* (In Chinese), 2000, **16**(5): 29–34.
[41] Liu Puyin & Zhang Hanjiang, Research on fuzzy neural network theory: a survey, *Fuzzy Systems and Math.* (In Chinese), 1998, **10**(1): 77–87.
[42] Liu Puyin & Zhang Weiming, Mathematical problems in research on reliability of telecommunication networks, *Acta Sinca Telecomm.,* 2000, **21**(10): 50–57.
[43] Lorentz G. G., *Bernstein polynomials,* New York: Chelsea, 1986.
[44] Nguyen H. T., A note on the extension principle for fuzzy sets, *J. Math. Anal. Appl.,* 1978, **64**: 369–380.
[45] Nie J., Nonlinear time-series forecasting: A fuzzy-neural approach, *Neurocomputing,* 1997, **16**: 63–76.
[46] Nishina T. & Hagiwara M., Fuzzy inference neural network, *Neurocomputing,* 1997, **14**: 223–239.
[47] Nobuhara H., Pedrycz W. & Hirota K., Fast solving method of fuzzy relational equation and their application to lossy image compression/reconstruction. *IEEE Trans. on Fuzzy Systems,* 2000, **8**: 325–334.
[48] Pal S. K. & King R. A., Image enhancement using smoothing with fuzzy sets, *IEEE Trans. on Systems Man and Cybernat.,* 1981, **11**: 494–501.
[49] Pal S. K. & King R. A., Image enhancement using fuzzy sets, *Electron. Lett.,* 1980, **16**: 376–378.
[50] Park S. & Han T., Iterative inversion of fuzzified neural networks, *IEEE Trans. on Fuzzy Systems.,* 2000, **8**: 266–280.
[51] Pedrycz W., Fuzzy neural networks and neurocomputations, *Fuzzy Sets and Systems,* 1993, **56**(1): 1–28.
[52] Pitas I. & Venetsanopoulos A. N., *Nonlinear digital filters— principles and applications,* Boston: Kluwer Academic Publishers, 1990.
[53] Raju G. V. S., Zhou J. & Kisner R. A., Hierarchical fuzzy control, *Internat. J. of Control,* 1991, **54**: 1201–1216.
[54] Runkler T. A. & Glesner M., A set of axioms for defuzzification strategies towards a theory of rational defuzzification operators, *Proc. IEEE Intern. Conf. on Fuzzy Systems,* 1993, **2**: 1161–1166.
[55] Saade J. J. & Diab H. B., Defuzzification techniques for fuzzy controllers, *IEEE Trans. on Systems, Man, and Cybernet.—Part B,* 2000, **30**: 223–229.
[56] Sankar K. Pal & Sushmita M., Multilayer perceptron, fuzzy sets and classification, *IEEE Trans. on Neural Networks,* 1992, **3**(5): 683–697.

[57] Scarselli F. & Tsoi A. C., Universal approximation using feedforward neural networks:a survey of some existing methods,and some new results, *Neural Networks,* 1998, **11**(1): 15–37.

[58] Simpson P. K., Fuzzy Min-Max neural networks–part 1: classification, *IEEE Trans. on Neural Networks,* 1992, **3**: 777–786.

[59] Simpson P. K., Fuzzy Min-Max neural networks–part 2: clustering, *IEEE Trans. on Fuzzy Systems,* 1993, **1**: 32–45.

[60] Tsai H. H. & Yu P. T., Adaptive fuzzy hybrid multichannel filters for removal of impulsive noise from color image, *Signal Processing,* 1999, **74**(2): 127–151.

[61] Wang L. X., *Adaptive fuzzy systems and control: design stability analysis,* Englewood Cliffs, NJ: PTR Prentice-Hall, 1994.

[62] Wang L. X., Universal approximation by hierarchical fuzzy systems, *Fuzzy Sets and Systems,* 1998, **93**(2): 223–230.

[63] Wang L. X., Analysis and design of hierarchical fuzzy systems. *IEEE Trans. on Fuzzy Systems,* 1999, **7**(6): 617–624.

[64] Yager R. & Zadeh L. A., Eds., *Fuzzy sets, neural networks, and soft computing,* New York: Van Nostrand Rehinhold, 1994.

[65] Ying H., Ding Y. & Li S., et al, Comparison of necessary conditions for typical Takagi–Sugeno and Mandani fuzzy systems as universal approximators, *IEEE Trans. on Systems, Man and Cybernet.-A,* 1999, **29**(5): 508–514.

[66] Yu P. T. & Chen R. C., Fuzzy stack filters—their definition, fundamental properties and application in image processing, *IEEE Trans. on Image Processing,* 1996, **5**: 838–854.

[67] Zadeh L. A., Fuzzy logic, neural networks and soft computing, *One-page Course Announcement of CS— the University of California at Berkeley,* 1993, **294**(4).

[68] Zadeh L. A., Fuzzy logic = computing with words, *IEEE Trans. on Fuzzy Systems,* 1996, **4**(2): 103–111.

[69] Zadeh L. A., From computing with numbers to computing with words—from manipulation of measurements to manipulation of perceptions, *IEEE Trans. on Circuits Systems-I,* 1999, **45**(1): 105–119.

[70] Zadeh L. A., Fuzzy logic, neural networks and soft computing, *Commun. ACM,* 1994, **37**(3): 77–84.

[71] Zadeh L. A., Soft computing and fuzzy logic, *IEEE Software,* 1994, **4** (11): 48–56.

[72] Zeng X. J. & Singh M. G., A relationship between membership functions and approximation accuracy in fuzzy systems, *IEEE Trans. on Systems, Man and Cybernet.* 1996, **26**(1): 176–180.

[73] Zhang X. H., Huang C. C. & Tan S. H., et al, The min-max function differentiation and training of fuzzy neural networks, *IEEE Trans. on Neural Networks,* 1996, **7**: 1139–1150.

[74] Zimmermann H. -J., *Fuzzy set theory and its application,* Dordrecht: Kluwer Academic Press, 1991.

CHAPTER II

Fuzzy Neural Networks for Storing and Classifying

If the fuzzy information handled by a FNN flows in one direction, from input to output, such a FNN is called a feedforward FNN. In this chapter, we focus mainly on such feedforward FNN's whose internal operations are based on the fuzzy operator pair '$\vee - \wedge$', which is called fuzzy associative memories (FAM's). A FAM constitutes a fuzzy perceptron, which was firstly proposed by Kosko B. about in 1987. It is developed based on a crisp feedforward neural network by introducing the fuzzy operators '$\vee$' and '$\wedge$'. The fuzzy information can be described by vectors in $[0, 1]^n$. An important subject related to FAM's is the storage capacity of the network [9, 10, 20, 23, 48], since the hardware and computation requirements for implementing a FAM with good storage capacity can be reduced, significantly. So there exist a lot of researches about the storage capacity of FAM in recent years. At first Kosko [25] develops a fuzzy Hebbian rule for FAM's, but it suffers from very poor storage capacity. To make up the defects of the fuzzy Hebbian rule, Fan et al improve Kosko's methods with maximum solution matrix in [12] to develop some equivalent conditions, under which a family of fuzzy pattern pairs can be stored in a FAM, completely. Recent years FAM's have been applied widely in many real fields, such as fuzzy relational structure modeling [18, 21, 39], signal processing [42], pattern classification [43, 45–47] and so on.

The classifying capability is another important subject related to the storage capacity of a FNN. The stronger the classifying ability of a FNN is, the more the FNN can store fuzzy patterns. By introducing the fuzzy operators '$\vee$' and '$\wedge$' the crisp adaptive resonance theory (ART) can be generalized as a FNN model—fuzzy ART, which can provide us with much easier classification for a given fuzzy pattern family [7].

In the chapter we present further researches about FAM's in storage capacity, learning algorithm for the connection weight matrices, associative space and so on. Some optimal connecting fashions of the neurons in a FAM, and some learning algorithms are developed based on storage capacity of the FAM. Finally we propose some systematic approaches to deal with the fuzzy ART, and its many classifying characteristics are developed. Some real examples show stronger classifying capability of the fuzzy ART.

§2.1 Two layer max–min fuzzy associative memory

Since the fuzzy operators '$\vee$' and '$\wedge$' can adapt the outputs to prefix range, such as [0, 1], also the operation '$\wedge$' is a threshold function [2, 3], no transfer function in FAM's is considered in the following. Suppose the input signal $\mathbf{x} \in [0, 1]^n$, and the output signal $\mathbf{y} \in [0, 1]^m$. Thus the input—output (I/O) relationship of a two layer FAM can be expressed as: $\mathbf{y} = \mathbf{x} \circ W$, where '$\circ$' means '$\vee - \wedge$' composition operation, $W = (w_{ij})_{n\times m} \in \mu_{n\times m}$ is the connection weight matrix, that is, if let $\mathbf{x} = (x_1, ..., x_n)$, $\mathbf{y} = (y_1, ..., y_m)$, then

$$y_j = \bigvee_{i=1}^{n} \{x_i \wedge w_{ij}\} \ (j = 1, ..., m). \tag{2.1}$$

Give a fuzzy pattern pair family as $(\mathcal{X}, \mathcal{Y}) \triangleq \{(\mathbf{x}_k, \mathbf{y}_k) | k \in \mathrm{P}\}$, where $\mathbf{x}_k = (x_1^k, ..., x_n^k)$, $\mathbf{y}_k = (y_1^k, ..., y_m^k)$, and $\mathrm{P} = \{1, ..., p\}$ $(p \in \mathbb{N})$. One of the main objects for studying (2.1) is to develop some learning algorithms for W, so that each pattern pair in $(\mathcal{X}, \mathcal{Y})$ can be stored in (2.1). Next let us present the topological architecture corresponding to (2.1), as shown in Figure 2.1.

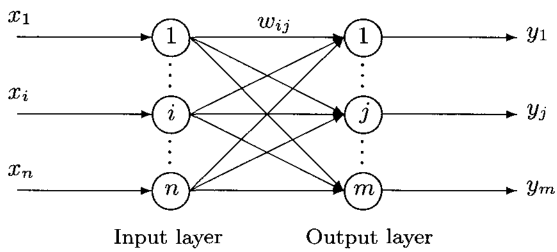


Figure 2.1 Topological architecture of two layer max–min FAM

For the fuzzy pattern pair family $(\mathcal{X}, \mathcal{Y}) = \{(\mathbf{x}_k, \mathbf{y}_k) | k \in \mathrm{P}\}$, Kosko in [25] develops a fuzzy Hebbian learning algorithm for W, that is by the following formula W can be established:

$$W = \bigvee_{k \in \mathrm{P}} \{\mathbf{x}_k^{\mathrm{T}} \circ \mathbf{y}_k\}, \tag{2.2}$$

where $\mathbf{x}_k^{\mathrm{T}}$ means the transpose of $\mathbf{x}_k$. The analytic learning algorithm (2.2) can not ensure each pattern pair $(\mathbf{x}_k, \mathbf{y}_k)$ $(k \in \mathrm{P})$ to be stored in (2.1). To guarantee more pattern pairs in $(\mathcal{X}, \mathcal{Y})$ to be stored in (2.1), we improve the algorithm (2.2) in the following. Denote $\mathrm{M} = \{1, ..., m\}$, $\mathrm{N} = \{1, ..., n\}$, and

$$G_{ij}(\mathcal{X}, \mathcal{Y}) = \{k \in \mathrm{P} | x_i^k > y_j^k\}, \quad E_{ij}(\mathcal{X}, \mathcal{Y}) = \{k \in \mathrm{P} | x_i^k = y_j^k\},$$
$$GE_{ij}(\mathcal{X}, \mathcal{Y}) = G_{ij}(\mathcal{X}, \mathcal{Y}) \cup E_{ij}(\mathcal{X}, \mathcal{Y}),$$
$$L_{ij}(\mathcal{X}, \mathcal{Y}) = \{k \in \mathrm{P} | x_i^k < y_j^k\}, \quad LE_{ij}(\mathcal{X}, \mathcal{Y}) = L_{ij}(\mathcal{X}, \mathcal{Y}) \cup E_{ij}(\mathcal{X}, \mathcal{Y}).$$

By the following analytic learning algorithm we can re-establish the connection weight matrix $W = W_0 = (w_{ij}^0)_{n\times m}$ in (2.1):

$$w_{ij}^0 = \begin{cases} \bigwedge_{k\in G_{ij}(\mathcal{X},\mathcal{Y})} \{y_j^k\}, & G_{ij}(\mathcal{X},\,\mathcal{Y})\neq \emptyset, \\ 1, & G_{ij}(\mathcal{X},\,\mathcal{Y})= \emptyset \end{cases} \tag{2.3}$$

For $i \in \mathrm{N}$, $j \in \mathrm{M}$, define the sets $S_{ij}^G(W_0, \mathcal{Y})$ and M^w respectively as follows:

$$S_{ij}^G(W_0,\mathcal{Y})= \{k \in GE_{ij}(\mathcal{X},\,\mathcal{Y})\,|\,y_j^k \le w_{ij}^0\};$$
$$M^w = \{W \in \mu_{n\times m}|\ \forall k \in \mathrm{P},\ \mathbf{x}_k \circ W = \mathbf{y}_k\}.$$

Theorem 2.1 *For the given fuzzy pattern pair family* $\{(\mathbf{x}_k,\,\mathbf{y}_k)|k \in \mathrm{P}\}$, *and* $W_0 = (w_{ij}^0)_{n\times m}$, *we have*

(i) $\forall k \in \mathrm{P}$, $\mathbf{x}_k \circ W_0 \subset \mathbf{y}_k$, *and if the fuzzy matrix* W *satisfies:* $\forall k \in \mathrm{P}$, $\mathbf{x}_k \circ W \subset \mathbf{y}_k$, *then* $W \subset W_0$;

(ii) If $M^w \neq \emptyset$, *it follows that,* $W_0 \in M^w$, *and* $\forall W = (w_{ij})_{n\times m} \in M^w$, $W \subset W_0$, *i.e.* $\forall i \in \mathrm{N}$, $j \in \mathrm{M}$, $w_{ij} \le w_{ij}^0$;

(iii) The set $M^w \neq \emptyset$ *if and only if* $\forall j \in \mathrm{M}$, $\bigcup_{i\in\mathrm{N}} S_{ij}^G(W_0,\mathcal{Y})= \mathrm{P}$.

Proof. (i) By the definition of W_0, $\forall k \in \mathrm{P}$, $\forall j \in \mathrm{M}$, easily we can show, $k \in G_{ij}(\mathcal{X},\,\mathcal{Y})$, $\Longrightarrow w_{ij}^0 \le y_j^k$. Moreover

$$\bigvee_{i\in\mathrm{N}}\{w_{ij}^0 \wedge x_i^k\}= \Big(\bigvee_{i|k\in G_{ij}(\mathcal{X},\mathcal{Y})}\{w_{ij}^0 \wedge x_i^k\}\Big)\vee\Big(\bigvee_{i|k\notin G_{ij}(\mathcal{X},\mathcal{Y})}\{w_{ij}^0 \wedge x_i^k\}\Big)\le y_j^k.$$

Therefore, $\forall k \in \mathrm{P}$, $\mathbf{x}_k \circ W_0 \subset \mathbf{y}_k$. Also if $W \in \mu_{n\times m}$ satisfies the given conditions, then we can conclude that

$$\bigvee_{i\in\mathrm{N}}\{x_i^k \wedge w_{ij}\}\le y_j^k,\Longrightarrow \forall i \in \mathrm{N},\ w_{ij} \wedge x_i^k \le y_j^k,\Longrightarrow \forall k \in G_{ij}(\mathcal{X},\,\mathcal{Y}),\ w_{ij} \le y_j^k.$$

So if $G_{ij}(\mathcal{X},\,\mathcal{Y})\neq \emptyset$, $w_{ij} \le \bigwedge_{k\in\mathrm{P}}\{y_j^k\}$, by (2.3) we have, $\forall i \in \mathrm{N}$, $j \in \mathrm{M}$, $w_{ij} \le w_{ij}^0$, that is, $W \subset W^0$. So (i) is true.

(ii) By the assumption we suppose $W = (w_{ij})_{n\times m} \in M^w$. For any $k \in \mathrm{P}$, $j \in \mathrm{M}$, we can conclude that

$$\bigvee_{i\in\mathrm{N}}\{x_i^k \wedge w_{ij}\}= y_j^k,\Longrightarrow \forall i \in \mathrm{N},\ w_{ij} \wedge x_i^k \le y_j^k,\Longrightarrow \forall k \in G_{ij}(\mathcal{X},\,\mathcal{Y}),\ w_{ij} \le y_j^k.$$

Similarly with (i) we can show, $W \subset W^0$. Also for any $k \in \mathrm{P}$, $j \in \mathrm{M}$, it is easy to show the following fact:

$$\forall i \in \mathrm{N},\ w_{ij}^0 \wedge x_i^k \le y_j^k,\Longrightarrow y_j^k \ge \bigvee_{i\in\mathrm{N}}\{x_i^k \wedge w_{ij}^0\}\ge \bigvee_{i\in\mathrm{N}}\{x_i^k \wedge w_{ij}\}= y_j^k.$$

Thus, $\bigvee_{i\in\mathrm{N}} \{x_i^k \wedge w_{ij}^0\}= y_j^k, \Longrightarrow W_0 \in M^w$. (ii) is proved.

(iii) Let $M^w \neq \emptyset$, and $W = (w_{ij})_{n\times m} \in M^w$. Then by (i), $W_0 = (w_{ij}^0)_{n\times m} \in M^w$, $W \subset W_0$. If there is $j_0 \in$ M, satisfying $\bigcup_{i\in\mathrm{N}} S_{ij_0}^G(W_0, \mathcal{Y}) \neq$ P, then there exists $k_0 \in$ P, but $\forall i \in$ N, $k_0 \notin S_{ij_0}^G(W_0, \mathcal{Y})$, and hence, either $x_i^{k_0} < y_{j_0}^{k_0}$ or, $y_{j_0}^{k_0} > w_{ij_0}^0 \geq w_{ij}$. Therefore, $w_{ij_0} \wedge x_i^{k_0} < y_{j_0}^{k_0}$. So $\bigvee_{i\in\mathrm{N}} \{x_i^{k_0} \wedge w_{ij_0}\} < y_{j_0}^{k_0}$, which is a contradiction since $W \in M^w$. So $\forall j \in$ M, $\bigcup_{i\in\mathrm{N}} S_{ij}^G(W_0, \mathcal{Y}) =$ P. On the other hand, let $\bigcup_{i\in\mathrm{N}} S_{ij}^G(W_0, \mathcal{Y}) =$ P $(j \in$ M). For any $j \in$ M, $k \in$ P, there is $i_0 \in$ N, so that $k \in S_{i_0 j}$. Thus

$$x_{i_0}^k \geq y_j^k,\ w_{ij_0}^0 \geq y_j^k, \Longrightarrow x_{i_0}^k \wedge w_{i_0 j}^0 \geq y_j^k, \Longrightarrow \bigvee_{i\in\mathrm{N}} \{x_i^k \wedge w_{ij}^0\} \geq y_j^k. \tag{2.4}$$

By the definition (2.3) for w_{ij}^0, it is easy to show

$$\forall i \in \mathrm{N},\ j \in \mathrm{M},\ k \in \mathrm{P},\ x_i^k \wedge w_{ij}^0 \leq y_j^k, \Longrightarrow \bigvee_{i\in\mathrm{N}} \{x_i^k \wedge w_{ij}^0\} \leq y_j^k.$$

Synthesizing (2.4) we get, $W_0 \in M^w, \Longrightarrow M^w \neq \emptyset$. (iii) is true. □

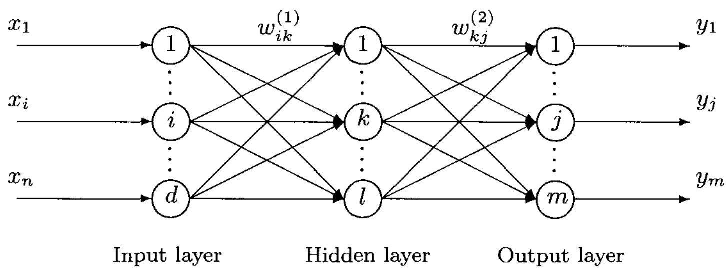


Figure 2.2 Topological architecture of three layer FAM

In the network (2.1), for the given connection weight matrix $W \in \mu_{n\times m}$, define

$$P_2^a(W) = \{(\mathbf{x}, \mathbf{y}) \in [0,\ 1]^n \times [0,\ 1]^m | \mathbf{x} \circ W = \mathbf{y}\},$$

The set $P_2^a(W)$ is called the associative space of (2.1). In practice, a main problem to study FAM is how to design the network (2.1) so that it can store as many fuzzy pattern pairs as possible, which can be viewed as such a problem that enlarging the associative space of (2.1). For the FAM's based on the fuzzy operator pair '$\vee - \wedge$', it is impossible to treat the problem by increasing unit or, node layers of FAM's. To account for the fact, we propose a three layer FAM, as shown in Figure 2.2.

Suppose the output of the hidden unit k in Figure 2.2 is o_k, then the corresponding I/O relationship can be expressed as follows:

$$\begin{cases} o_k = \bigvee_{i=1}^{n} \{x_i \wedge w_{ik}^{(1)}\} \ (k = 1, ..., l), \\ y_j = \bigvee_{k=1}^{l} \{o_k \wedge w_{kj}^{(2)}\} \ (j = 1, ..., m). \end{cases} \tag{2.5}$$

(2.5) stands for a three layer FAM. Next let us prove that the storage capacities of (2.1) and (2.5) are identical.

Theorem 2.2 *Let $W_1 = (w_{ik}^{(1)})_{n\times l}$, $W_2 = (w_{ik}^{(2)})_{l\times m}$, and $P_3^a(W_1, W_2)$ is the associative space of the three layer FAM (2.5), i.e.*

$$P_3^a(W_1, W_2) = \{(\mathbf{x}, \mathbf{y})\big|\mathbf{x} = (x_1, ..., x_n),\ \mathbf{y} = (y_1, ..., y_m) \text{ satisfy } (2.5)\}.$$

Then we can conclude that

(i) For given W_1, W_2, there is $W \in \mu_{n\times m}$, so that $P_3^a(W_1, W_2) \subset P_2^a(W)$;

(ii) If $l \geq m \wedge n$, then for given $W \in \mu_{n\times m}$, there are $W_1 \in \mu_{n\times l}$, $W_2 \in \mu_{l\times m}$, so that $P_2^a(W) \subset P_3^a(W_1, W_2)$.

Proof. (i) For any $(\mathbf{x}, \mathbf{y}) \in P_3^a(W_1, W_2) : \mathbf{x} = (x_1, ..., x_n),\ \mathbf{y} = (y_1, ..., y_m)$, by the assumption we get

$$\begin{aligned} y_j &= \bigvee_{k=1}^{l}\Big\{\Big(\bigvee_{i=1}^{n}\{x_i \wedge w_{ik}^{(1)}\}\Big)\wedge w_{kj}^{(2)}\Big\} = \bigvee_{k=1}^{l}\Big\{\bigvee_{i=1}^{n}\{x_i \wedge w_{ik}^{(1)} \wedge w_{kj}^{(2)}\}\Big\} \\ &= \bigvee_{i=1}^{n}\Big\{\bigvee_{k=1}^{l}\{x_i \wedge w_{ik}^{(1)} \wedge w_{kj}^{(2)}\}\Big\} = \bigvee_{i=1}^{n}\Big\{x_i \wedge \Big(\bigvee_{k=1}^{l}\{w_{ik}^{(1)} \wedge w_{kj}^{(2)}\}\Big)\Big\}, \end{aligned} \tag{2.6}$$

where, $j \in \mathrm{M}$. Define the connection weight matrix of (2.1), $W = (w_{ij})_{n\times m}$ as follows:

$$w_{ij} = \bigvee_{k=1}^{l}\{w_{ik}^{(1)} \wedge w_{kj}^{2}\} \ (i \in \mathrm{N},\ j \in \mathrm{M}).$$

Then by (2.6) it follows that, $\forall(\mathbf{x}, \mathbf{y}) \in P_3^a(W_1, W_2) : \mathbf{x} = (x_1, ..., x_n), \mathbf{y} = (y_1, ..., y_m)$, we have

$$\forall j \in \mathrm{M},\ y_j = \bigvee_{i=1}^{n}\{x_i \wedge w_{ij}\},$$

i.e. $\mathbf{x} \circ W = \mathbf{y}, \Longrightarrow (\mathbf{x}, \mathbf{y}) \in P_2^a(W)$. Therefore, $P_3^a(W_1, W_2) \subset P_2^a(W)$. (i) is proved.

(ii) Let $l \geq m$. For any $(\mathbf{x}, \mathbf{y}) \in P_2^a(W) : \mathbf{x} = (x_1, ..., x_n),\ \mathbf{y} = (y_1, ..., y_m)$, then for $j \in \mathrm{M}$, $y_j = \bigvee_{i=1}^{n}\{x_i \wedge w_{ij}\}$. For $i \in \mathrm{N}$, $j \in \mathrm{M}$ and $k = 1, ..., l$, define

$$w_{ik}^{(1)} = \begin{cases} w_{ik}, & k \leq m, \\ 0, & m < k \leq l; \end{cases} \qquad w_{kj}^{(2)} = \begin{cases} 1, & k \leq m,\ k = j \\ 0, & \text{otherwise.} \end{cases}$$

Then for $k = 1, ..., l$, by (2.5) easily we can get

$$o_k = \bigvee_{i=1}^{n} \{x_i \wedge w_{ik}^{(1)}\} = \begin{cases} \bigvee_{i=1}^{n} \{x_i \wedge w_{ik}\}, & k \leq m, \\ 0, & m < k \leq l. \end{cases}$$

Thus, we may conclude that

$$\forall j \in \mathrm{M}, \ \bigvee_{k=1}^{l} \{o_k \wedge w_{kj}^{(2)}\} = o_j = \bigvee_{i=1}^{n} \{x_i \wedge w_{ij}\} = y_j.$$

So if let $W_1 = (w_{ik}^{(1)})_{n\times l}$, $W_2 = \left(w_{kj}^{(2)}\right)_{k\times m}$, then $(\mathbf{x}, \mathbf{y}) \in P_3^a(W_1, W_2)$. Therefore, $P_2^a(W) \subset P_3^a(W_1, W_2)$. (ii) is proved. □

By Theorem 2.2, increasing unit layer of a FAM based on '$\vee - \wedge$' can not improve the storage capacity. To improve FAM's in their storage capacity or associative space, let us now aim at the optimization of the connecting fashions among the units.

2.1.1 FAM with threshold

In the FAM's as shown in Figure 2.1, we introduce thresholds c_i, d_j to the input unit i and output unit j, respectively, where $i \in \mathrm{N}$, $j \in \mathrm{M}$. Then the corresponding I/O relationship can be expressed as

$$y_j = \Big(\bigvee_{i=1}^{n} \{(x_i \vee c_i) \wedge w_{ij}\}\Big) \vee d_j = \bigvee_{i=1}^{n} \{(x_i \vee c_i \vee d_j) \wedge (w_{ij} \vee d_j)\}, \tag{2.7}$$

(2.7) is called a FAM with threshold. Using the fuzzy matrix $W = (w_{ij})_{n\times m}$ and the fuzzy vector $\mathbf{c} = (c_1, ..., c_n)$, $\mathbf{d} = (d_1, ..., d_m)$ we re-write (2.7) as

$$\mathbf{y} = \big((\mathbf{x} \vee \mathbf{c}) \circ W\big) \vee \mathbf{d}. \tag{2.8}$$

For $j \in \mathrm{M}$, we introduce the following set:

$$J_i(\mathcal{X}, \mathcal{Y}) = \{j \in \mathrm{M} | LE_{ij}(\mathcal{X}, \mathcal{Y}) \neq \emptyset\}.$$

By the following (2.9) (2.10) we establish the connection weight matrix $W_0 = (w_{ij}^0)_{n\times m}$ and the threshold vectors $\mathbf{c}^0 = (c_1^0, ..., c_n^0)$, $\mathbf{d}^0 = (d_1^0, ..., d_m^0)$:

$$w_{ij}^0 = \begin{cases} \bigwedge_{k\in G_{ij}(\mathcal{X},\mathcal{Y})} \{y_j^k\}, & G_{ij}(\mathcal{X}, \mathcal{Y}) \neq \emptyset, \\ 1, & G_{ij}(\mathcal{X}, \mathcal{Y}) = \emptyset; \end{cases} \tag{2.9}$$

$$c_i^0 = \begin{cases} \bigvee_{k\in LE_{ij}(\mathcal{X},\mathcal{Y}),\, j\in J_i(\mathcal{X},\mathcal{Y})} \{y_j^k\}, & J_i(\mathcal{X}, \mathcal{Y}) \neq \emptyset, \\ 0, & J_i(\mathcal{X}, \mathcal{Y}) = \emptyset. \end{cases} \tag{2.10}$$

And $d_j^0 = \bigwedge_{k\in P}\{y_j^k\}$. For $i \in N$, $j \in M$, define the sets

$$TG_{ij}\big((\mathcal{X},\mathbf{c}_0);\mathbf{d}_0,\mathcal{Y}\big)=\big\{k\in P\big|x_i^k\vee c_i^0\vee d_j^0>y_j^k\big\},$$
$$TE_{ij}\big((\mathcal{X},\mathbf{c}_0);\mathbf{d}_0,\mathcal{Y}\big)=\big\{k\in P\big|x_i^k\vee c_i^0\vee d_j^0=y_j^k\big\},$$
$$TL_{ij}\big((\mathcal{X},\mathbf{c}_0);\mathbf{d}_0,\mathcal{Y}\big)=\big\{k\in P\big|x_i^k\vee c_i^0\vee d_j^0<y_j^k\big\},$$
$$TGE_{ij}\big((\mathcal{X},\mathbf{c}_0);\mathbf{d}_0,\mathcal{Y}\big)=TG_{ij}\big((\mathcal{X},\mathbf{c}_0);\mathbf{d}_0,\mathcal{Y}\big)\cup TE_{ij}\big((\mathcal{X},\mathbf{c}_0);\mathbf{d}_0,\mathcal{Y}\big),$$
$$TS_{ij}^G\big((W_0,\mathbf{d}_0);\mathcal{Y}\big)=\big\{k\in TGE_{ij}\big((\mathcal{X},\mathbf{c}_0);\mathbf{d}_0,\mathcal{Y}\big)\big|y_j^k\le w_{ij}^0\vee d_j^0\big\}.$$

Since $\forall i \in N$, $j \in M$, $TGE_{ij}\big((\mathcal{X},\mathbf{c}_0);\mathbf{d}_0,\mathcal{Y}\big)\supset GE_{ij}(\mathcal{X},\mathcal{Y})$, and $d_j^0 \le w_{ij}^0$ are obviously true, we have

$$TS_{ij}^G\big((W_0,\mathbf{d}_0);\mathcal{Y}\big)=\big\{k\in TGE_{ij}\big((\mathcal{X},\mathbf{c}_0);\mathbf{d}_0,\mathcal{Y}\big)\big|y_j^k\le w_{ij}^0\big\},$$
$$TS_{ij}^G\big((W_0,\mathbf{d}_0);\mathcal{Y}\big)\supset S_{ij}^G(W_0,\mathcal{Y}).$$

In (2.8) we give the connection weight matrix $W \in \mu_{n\times m}$, and the threshold vectors $\mathbf{c}$, $\mathbf{d}$. Define the set

$$TP^a(W,\mathbf{c},\mathbf{d})=\big\{(\mathbf{x},\mathbf{y})\in[0,1]^n\times[0,1]^m\big|\big((\mathbf{x}\vee\mathbf{c})\circ W\big)\vee\mathbf{d}=\mathbf{y}\big\}.$$

We call $TP^a(W, \mathbf{c}, \mathbf{d})$ the associative space of the FAM with threshold. Let $\{(\mathbf{x}_k, \mathbf{y}_k)|k \in P\}$ be a given fuzzy pattern pair family. define

$$M^{wcd}=\big\{(W,\mathbf{c},\mathbf{d})\big|\forall k\in P,\ \big((\mathbf{x}_k\vee\mathbf{c})\circ W\big)\vee\mathbf{d}=\mathbf{y}_k\big\}.$$

Theorem 2.3 *Let* $W = (w_{ij})_{n\times m} \in \mu_{n\times m}$, $\mathbf{c} = (c_1, ..., c_n) \in [0, 1]^n$, *and* $\mathbf{d} = (d_1, ..., d_m) \in [0,1]^m$, $(W, \mathbf{c}, \mathbf{d}) \in M^{wcd}$. *Then* $\forall i \in N$, $j \in M$, $w_{ij} \le w_{ij}^0$, $d_j \le d_j^0$.

Proof. Let $a, b \in [0, 1]$, define '$\circledS$' as follows:

$$a\circledS b=\begin{cases}1, & a\le b,\\ b, & a>b.\end{cases}$$

Then we can show, $\forall a, b, c \in [0, 1]$, $a \circledS (a\wedge b) \ge b$. And $b > c, \Longrightarrow a \circledS b \ge a \circledS c$. Since $(W, \mathbf{c}, \mathbf{d}) \in M^{wcd}$, it follows that $\forall j \in M$, $d_j \le \bigwedge_{k\in P}\{y_j^k\} = d_j^0$. So next it suffices to prove $w_{ij} \le w_{ij}^0$. If $G_{ij}(\mathcal{X},\mathcal{Y}) = \emptyset$, then $w_{ij}^0 = 1 \ge w_{ij}$. If $G_{ij}(\mathcal{X},\mathcal{Y}) \ne \emptyset$, there is $k_0 \in G_{ij}(\mathcal{X},\mathcal{Y})$, so that

$$y_j^{k_0}=\bigwedge_{k\in G_{ij}(\mathcal{X},\mathcal{Y})}\{y_j^k\},\Longrightarrow w_{ij}^0=\bigwedge_{k\in G_{ij}(\mathcal{X},\mathcal{Y})}\{y_j^k\}=y_j^{k_0}<x_i^{k_0}.$$

Using the definition of 'Ⓢ' and the assumptions easily we can show

$$\begin{aligned} w_{ij}^0 &= y_j^{k_0} = x_i^{k_0} \circledS\, y_j^{k_0} = x_i^{k_0} \circledS \left(\bigvee_{i'=1}^{n} \left\{ (x_{i'}^{k_0} \vee c_{i'} \vee d_j) \wedge (w_{i'j} \vee d_j) \right\} \right) \\ &\geq x_i^{k_0} \circledS\, (x_i^{k_0} \wedge w_{ij}) \geq w_{ij}. \end{aligned}$$

The theorem is therefore proved. □

By Theorem 2.3, $\mathbf{c}^0$, W_0 defined by (2.9) possess a maximality with the sense of storing fuzzy patterns.

Theorem 2.4 *Let* $(W, \mathbf{c}, \mathbf{d}) \in M^{wcd}$. *Then there is a threshold vector* $\mathbf{c}_1 = (c_1^1, ..., c_n^1)$, *so that* $\forall i \in \mathrm{N}$, $c_i^1 \leq c_i^0$, *and* $(W, \mathbf{c}_1, \mathbf{d}_0) \in M^{wcd}$.

Proof. For any $i \in \mathrm{N}$, define c_i^1 as follows:

$$c_i^1 = \begin{cases} \bigwedge_{k\in\mathrm{P}, j\in\mathrm{M}} \{y_j^k\}, & J_i(\mathcal{X}, \mathcal{Y}) \neq \emptyset, \\ 0, & J_i(\mathcal{X}, \mathcal{Y}) = \emptyset. \end{cases}$$

If $J_i(\mathcal{X}, \mathcal{Y}) = \emptyset$, then $c_i^1 = 0 \leq c_i^0$, and $\forall k \in \mathrm{P}$, $j \in \mathrm{M}$, $x_i^k > y_j^k$. Then by (2.9) and Theorem 2.3 it follows that $x_i^k > w_{ij}^0 \vee d_j^0 \geq w_{ij} \vee d_j$. Therefore

$$\begin{aligned} J_i(\mathcal{X}, \mathcal{Y}) = \emptyset, \Longrightarrow\ & (x_i^k \vee c_i \vee d_j) \wedge (w_{ij} \vee d_j) = w_{ij} \vee d_j \\ \leq\ & (x_i^k \vee c_i^1 \vee d_j^0) \wedge (w_{ij} \vee d_j^0) \leq y_j^k. \end{aligned} \tag{2.11}$$

If $J_i(\mathcal{X}, \mathcal{Y}) \neq \emptyset$, define the set $KJ_i(\mathbf{c}, \mathcal{Y}) = \{(k, j) \in \mathrm{P} \times \mathrm{M} | c_i > y_j^k\}$. If $KJ_i(\mathbf{c}, \mathcal{Y}) = \emptyset$, then we get

$$\forall j \in \mathrm{M},\, k \in \mathrm{P},\, c_i \leq y_j^k,\ \ c_i \leq \bigwedge_{k\in\mathrm{P}, j\in\mathrm{M}} \{y_j^k\} = c_i^1.$$

Therefore $\forall j \in \mathrm{M}$, $k \in \mathrm{P}$, by $KJ_i(\mathbf{c}, \mathcal{Y}) = \emptyset$, we imply

$$(x_i^k \vee c_i \vee d_j) \wedge (w_{ij} \vee d_j) \leq (x_i^k \vee c_i^1 \vee d_j^0) \wedge (w_{ij} \vee d_j^0) \leq y_j^k. \tag{2.12}$$

If $KJ_i(\mathbf{c}, \mathcal{Y}) \neq \emptyset$, then we can conclude that

$$(k, j) \in KJ_i(\mathbf{c}, \mathcal{Y}), \Longrightarrow w_{ij} \leq y_j^k, \Longrightarrow w_{ij} \leq \bigwedge_{(k,j)\in KJ_i(\mathbf{c},\mathcal{Y})} \{y_j^k\}.$$

Also it is easy to show the following facts:

$$\bigwedge_{(k,j)\in KJ_i(\mathbf{c},\mathcal{Y})} \{y_j^k\} \leq \bigwedge_{(k,j)\notin KJ_i(\mathbf{c},\mathcal{Y})} \{y_j^k\};\ \ w_{ij} \leq \bigwedge_{k\in\mathrm{P}} \{y_j^k\}.$$

That is, when $KJ_i(\mathbf{c}, \mathcal{Y}) \neq \emptyset$ (2.12) holds also. So

$$\begin{aligned} J_i(\mathcal{X}, \mathcal{Y}) \neq \emptyset, \Longrightarrow\ & (x_i^k \vee c_i \vee d_j) \wedge (w_{ij} \vee d_j) = w_{ij} \vee d_j \\ \leq\ & (x_i^k \vee c_i^1 \vee d_j^0) \wedge (w_{ij} \vee d_j^0) \leq y_j^k. \end{aligned} \tag{2.13}$$

Thus, by (2.11) (2.13) and $(W,\, \mathbf{c},\, \mathbf{d}) \in M^{wcd}$, we get

$$\begin{aligned} y_j^k &= \bigvee_{i\in \mathrm{N}} \{(x_i^k \vee c_i \vee d_j) \wedge (w_{ij} \vee d_j)\} \\ &= \Big(\bigvee_{i|J_i(\mathcal{X},\mathcal{Y})=\emptyset} \{(x_i^k \vee c_i \vee d_j) \wedge (w_{ij} \vee d_j)\}\Big) \vee \\ &\quad \vee \Big(\bigvee_{i|J_i(\mathcal{X},\mathcal{Y})\neq\emptyset} \{(x_i^k \vee c_i \vee d_j) \wedge (w_{ij} \vee d_j)\}\Big) \\ &\leq \Big(\bigvee_{i|J_i(\mathcal{X},\mathcal{Y})=\emptyset} \{(x_i^k \vee c_i^1 \vee d_j^0) \wedge (w_{ij} \vee d_j^0)\}\Big) \vee \\ &\quad \vee \Big(\bigvee_{i|J_i(\mathcal{X},\mathcal{Y})\neq\emptyset} \{(x_i^k \vee c_i^1 \vee d_j^0) \wedge (w_{ij} \vee d_j^0)\}\Big) \\ &= \bigvee_{i\in \mathrm{N}} \{(x_i^k \vee c_i^1 \vee d_j^0) \wedge (w_{ij} \vee d_j^0)\} \leq y_j^k. \end{aligned}$$

Hence $\forall j \in \mathrm{M},\, k \in \mathrm{P},\ \bigvee_{i\in\mathrm{N}} \{(x_i^k \vee c_i^1 \vee d_j^0) \wedge (w_{ij} \vee d_j^0)\} = y_j^k$. That is, $\mathbf{y} = \big((\mathbf{x} \vee \mathbf{c}_1) \circ W\big) \vee \mathbf{d}_0$. Consequently, $(W,\, \mathbf{c}_1,\, \mathbf{d}_0) \in M^{wcd}$. □

Lemma 2.1 *Suppose $j \in \mathrm{M}$, $k \in \mathrm{P}$. Then the following facts hold:*

$$\begin{aligned} y_j^k &= \bigvee_{i|k\in TS_{ij}^G((W_0,\mathbf{d}_0);\mathcal{Y})} \{(x_i^k \vee c_i^0 \vee d_j^0) \wedge (w_{ij}^0 \vee d_j^0)\} \\ &> \bigvee_{i|k\notin TS_{ij}^G((W_0,\mathbf{d}_0);\mathcal{Y})} \{(x_i^k \vee c_i^0 \vee d_j^0) \wedge (w_{ij}^0 \vee d_j^0)\}. \end{aligned}$$

Proof. If $k \notin TS_{ij}^G\big((W_0, \mathbf{d}_0); \mathcal{Y}\big)$, then

$$\text{either } x_i^k \vee c_i^0 \vee d_j^0 < y_j^k \text{ or, } x_i^k \vee c_i^0 \vee d_j^0 \geq y_j^k > w_{ij}^0 \vee d_j^0,$$

which can implies, $(x_i^k \vee c_i^0 \vee d_j^0) \wedge (w_{ij}^0 \vee d_j^0) < y_j^k$. Therefore

$$\bigvee_{i|k\notin TS_{ij}^G((W_0,\mathbf{d}_0);\mathcal{Y})} \{(x_i^k \vee c_i^0 \vee d_j^0) \wedge (w_{ij}^0 \vee d_j^0)\} < y_j^k.$$

If $k \in TS_{ij}^G\big((W_0, \mathbf{d}_0); \mathcal{Y}\big)$, then we have

$$\text{either } x_i^k \vee c_i^0 \vee d_j^0 = y_j^k \leq w_{ij}^0 \vee d_j^0 \text{ or, } x_i^k \vee c_i^0 \vee d_j^0 > y_j^k.$$

And $y_j^k \leq w_{ij}^0 \vee d_j^0$. Also by the definitions of w_{ij}^0 and d_j^0, it follows that $w_{ij}^0 \vee d_j^0 \leq y_j^k$. Hence

$$w_{ij}^0 \vee d_j^0 = y_j^k,\ \ (x_i^k \vee c_i^0 \vee d_j^0) \wedge (w_{ij}^0 \vee d_j^0) = y_j^k.$$

That is, the following equality holds:

$$\bigvee_{i|k\in TS^G_{ij}\left((W_0,\mathbf{d}_0);\mathcal{Y}\right)} \left\{(x_i^k\vee c_i^0\vee d_j^0)\wedge(w_{ij}^0\vee d_j^0)\right\}=y_j^k.$$

And hence the lemma is proved. □

Theorem 2.5 *For the given fuzzy pattern pair family* $\{(\mathbf{x}_k, \mathbf{y}_k)|k\in \mathrm{P}\}$, *the set* $M^{wcd}\neq\emptyset$ *if and only if* $\forall j\in \mathrm{M}$, $\bigcup\limits_{i\in \mathrm{N}} TS^G_{ij}\left((W_0,\mathbf{d}_0);\mathcal{Y}\right)=\mathrm{P}$.

Proof. Necessity: Let $W=(w_{ij})_{n\times m}\in\mu_{n\times m}$, $\mathbf{c}=(c_1,...,c_n)\in[0,\,1]^n$, $\mathbf{d}=(d_1,...,d_m)\in[0,1]^m$, so that $(W,\mathbf{c},\mathbf{d})\in M^{wcd}$. If the conclusion is false, there is $j_0\in\mathrm{M}$, satisfying $\bigcup\limits_{i\in\mathrm{N}} TS^G_{ij_0}\left((W_0,\mathbf{d}_0);\mathcal{Y}\right)\neq\mathrm{P}$. Thus, there is $k\in\mathrm{P}$, so that $\forall i\in\mathrm{N}$, $k\notin TS^G_{ij_0}\left((W_0,\mathbf{d}_0);\mathcal{Y}\right)$. Therefore, $\forall i\in\mathrm{N}$, either $k\in TL_{ij_0}\left((\mathcal{X},\mathbf{c}_0);\mathbf{d}_0,\mathcal{Y}\right)$ or, $k\in TGE_{ij_0}\left((\mathcal{X},\mathbf{c}_0);\mathbf{d}_0,\mathcal{Y}\right)$, $y^k_{j_0}>w^0_{ij_0}\vee d^0_j$. So

$$\begin{aligned}&\bigvee_{i\in\mathrm{N}}\left\{(x_i^k\vee c_i^0\vee d_j^0)\wedge(w_{ij}^0\vee d_j^0)\right\}\\&=\Big(\bigvee_{i|k\in TGE_{ij_0}\left((W_0,\mathbf{c}_0);\mathbf{d}_0,\mathcal{Y}\right)}\left\{(x_i^k\vee c_i^0\vee d_j^0)\wedge(w_{ij}^0\vee d_j^0)\right\}\Big)\vee\\&\quad\vee\Big(\bigvee_{i|k\in TL_{ij_0}\left((W_0,\mathbf{c}_0);\mathbf{d}_0,\mathcal{Y}\right)}\left\{(x_i^k\vee c_i^0\vee d_j^0)\wedge(w_{ij}^0\vee d_j^0)\right\}\Big)<y^k_{j_0}.\end{aligned}\tag{2.14}$$

By the assumption and Theorem 2.4, there is a fuzzy vector $\mathbf{c}_1=(c^1_1,...,c^1_n)$: $\forall i\in\mathrm{N}$, $c^1_i\leq c^0_i$, so that $(W,\,\mathbf{c}_1,\,\mathbf{d}_0)\in M^{wcd}$. So by (2.14) and Theorem 2.3 it follows that

$$\begin{aligned}y^k_{j_0}&=\bigvee_{i\in\mathrm{N}}\left\{(x_i^k\vee c_i\vee d_j)\wedge(w_{ij}\vee d_j)\right\}=\bigvee_{i\in\mathrm{N}}\left\{(x_i^k\vee c_i^1\vee d_j^0)\wedge(w_{ij}\vee d_j^0)\right\}\\&\leq\bigvee_{i\in\mathrm{N}}\left\{(x_i^k\vee c_i^0\vee d_j^0)\wedge(w_{ij}^0\vee d_j^0)\right\}<y^k_{j_0},\end{aligned}$$

Which is a contradiction. The necessity is proved.

Sufficiency: $\forall j\in\mathrm{M}$, $k\in\mathrm{P}$, there is $i\in\mathrm{N}$, so that $k\in TS^G_{ij}\left((W_0,\mathbf{d}_0);\mathcal{Y}\right)$. Hence if let $w_{ij}=w^0_{ij}$, $c_i=c^0_i$, $d_j=d^0_j$, then Lemma 2.1 implies that

$$\begin{aligned}&\bigvee_{i\in\mathrm{N}}\left\{(x_i^k\vee c_i\vee d_j)\wedge(w_{ij}\vee d_j)\right\}\\&=\Big(\bigvee_{i|k\in TS^G_{ij}\left((W_0,\mathbf{d}_0);\mathcal{Y}\right)}\left\{(x_i^k\vee c_i^0\vee d_j^0)\wedge(w_{ij}^0\vee d_j^0)\right\}\Big)\vee\\&\quad\vee\Big(\bigvee_{i|k\notin TS^G_{ij}\left((W_0,\mathbf{d}_0);\mathcal{Y}\right)}\left\{(x_i^k\vee c_i^0\vee d_j^0)\wedge(w_{ij}^0\vee d_j^0)\right\}\Big)\\&=\bigvee_{i|k\in TS^G_{ij}\left((W_0,\mathbf{d}_0);\mathcal{Y}\right)}\left\{(x_i^k\vee c_i^0\vee d_j^0)\wedge(w_{ij}^0\vee d_j^0)\right\}=y_j^k.\end{aligned}$$

Then put $W_0 = (w_{ij}^0)$, $\mathbf{c}_0 = (c_1^0, ..., c_n^0)$, $\mathbf{d}_0 = (d_1^0, ..., d_m^0)$. Thus, $(W_0, \mathbf{c}_0, \mathbf{d}_0) \in M^{wcd}$. That is, $M^{wcd} \neq \emptyset$. □

Theorem 2.6 *For a given fuzzy pattern family* $\{(\mathbf{x}_k, \mathbf{y}_k)|k \in \mathrm{P}\}$, *let* $M^w \neq \emptyset$, *i.e. there is* $W \in \mu_{n\times m}$, *so that* $\forall k \in \mathrm{P}$, $\mathbf{x}_k \circ W = \mathbf{y}_k$. *Let* $W_0 = (w_{ij}^0)$, $\mathbf{c}_0 = (c_1^0, ..., c_n^0)$, $\mathbf{d}_0 = (d_1^0, ..., d_m^0)$. *Then* $(W_0, \mathbf{c}_0, \mathbf{d}_0) \in M^{wcd}$.

Proof. By Theorem 2.1 we get,

$$\forall j \in \mathrm{M}, \bigcup_{i\in \mathrm{N}} S_{ij}^G(W_0, \mathcal{Y}) = \mathrm{P}.$$

Since $S_{ij}^G(W_0, \mathcal{Y}) \subset TS_{ij}^G((W_0, \mathbf{d}_0); \mathcal{Y})$, it follows that $\bigcup_{i\in \mathrm{N}} TS_{ij}^G((W_0, \mathbf{d}_0); \mathcal{Y}) = \mathrm{P}$. Theorem 2.4 implies, $M^{wcd} \neq \emptyset$, and $(W_0, \mathbf{c}_0, \mathbf{d}_0) \in M^{wcd}$. □

2.1.2 Simulation example

In the subsection we demonstrate that FAM (2.8) with threshold possesses good storage capacity by a simulation example. Let $\mathrm{N} = \{1, 2, 3, 4, 5\}$, $\mathrm{M} = \{1, 2, 3\}$, and $\mathrm{P} = \{1, ..., 8\}$. By Table 2.1 we give a fuzzy pattern pair family as $\{(\mathbf{x}_k, \mathbf{y}_k)|k \in \mathrm{P}\}$. Using the following steps we can realize the algorithm (2.9).

Table 2.1 Fuzzy pattern pair family

k	$\mathbf{x}_k$	$\mathbf{y}_k$
1	(0.5, 0.5, 0.4, 0.4, 0.3)	(0.5, 0.6, 0.3)
2	(0.1, 0.3, 0.3, 0.4, 0.4)	(0.5, 0.6, 0.4)
3	(0.8, 0.4, 0.6, 0.7, 0.4)	(0.6, 0.8, 0.4)
4	(0.3, 0.4, 0.4, 0.3, 0.4)	(0.5, 0.6, 0.4)
5	(0.6, 0.4, 0.7, 0.7, 0.5)	(0.7, 0.7, 0.5)
6	(0.1, 0.1, 0.2, 0.2, 0.1)	(0.5, 0.6, 0.3)
7	(0.7, 0.2, 0.4, 0.3, 0.2)	(0.5, 0.7, 0.3)
8	(0.8, 0.4, 0.3, 0.4, 0.2)	(0.5, 0.8, 0.3)

Step 1. For any $i \in \mathrm{N}$, $j \in \mathrm{M}$, calculate the sets $G_{ij}(\mathcal{X}, \mathcal{Y})$, $LE_{ij}(\mathcal{X}, \mathcal{Y})$, and establish w_{ij}^0, c_i^0, d_j^0;

Step 2. For $i \in \mathrm{N}$, $j \in \mathrm{M}$, calculate and determine the following sets:

$$TG_{ij}((\mathcal{X}, \mathbf{c}_0); \mathbf{d}_0, \mathcal{Y}),\ TE_{ij}((\mathcal{X}, \mathbf{c}_0); \mathbf{d}_0, \mathcal{Y}),$$
$$TGE_{ij}((\mathcal{X}, \mathbf{c}_0); \mathbf{d}_0, \mathcal{Y}),\ TS_{ij}^G((W_0, \mathbf{d}_0); \mathcal{Y});$$

Step 3. For any $j \in \mathrm{M}$, discriminate the following equality:

$$\bigcup_{i\in \mathrm{N}} TS_{ij}^G((W_0, \mathbf{d}_0); \mathcal{Y}) = \mathrm{P}?$$

if yes go to the following step, otherwise go to Step 5;

Step 4. Put $W_0 = (w_{ij}^0)$, $\mathbf{c}_0 = (c_1^0, ..., c_n^0)$, $\mathbf{d}_0 = (d_1^0, ..., d_m^0)$;

Step 5. Stop.

By above steps we get, $\mathbf{c}_0 = (0.3, 0.3, 0.3, 0.3, 0.3)$, $\mathbf{d}_0 = (0.5, 0.6, 0.3)$, the threshold vectors of input units and output units, respectively, the connection weight matrix W_0 as follows:

$$W_0^{\mathrm{T}} = \begin{pmatrix} 0.5 & 1.0 & 1.0 & 0.6 & 1.0 \\ 1.0 & 1.0 & 1.0 & 1.0 & 1.0 \\ 0.3 & 0.3 & 0.3 & 0.3 & 1.0 \end{pmatrix}$$

we may easily show that the given fuzzy pattern pair family $\{(\mathbf{x}_k, \mathbf{y}_k)|k \in \mathrm{P}\}$ satisfies the conditions that for each $j \in \mathrm{M}$, $\bigcup_{i \in \mathrm{N}} TS_{ij}^G((W_0, \mathbf{d}_0); \mathcal{Y}) = \mathrm{P}$. So by Theorem 2.5, all fuzzy pattern pairs in Table 2.1 can be stored in FAM (2.8). If we use the Hebbian learning rule, easily we may imply that only $(\mathbf{x}_5, \mathbf{y}_5)$ can be stored in FAM (2.1) [38]. Therefore, we can improve a FAM in storage capacity by introducing suitable thresholds and learning algorithms.

§2.2 Fuzzy δ−learning algorithm

Introducing threshold to units and designing analytic learning algorithm can improve a FAM as (2.1) in its storage capacity. However, analytic learning algorithm can not show the adaptivity and self-adjustability of FAM's. To overcome such defects we in this section develop a dynamic learning scheme, the fuzzy δ−learning algorithm, and present its convergence.

2.2.1 FAM's based on '$\vee - \wedge$'

Give a fuzzy pattern pair family $\{(\mathbf{x}_k, \mathbf{y}_k)|k \in \mathrm{P}\}$, and by the matrices X, Y we denote $X = (\mathbf{x}_1, ..., \mathbf{x}_p)^{\mathrm{T}}$, $Y = (\mathbf{y}_1, ..., \mathbf{y}_p)^{\mathrm{T}}$, that is

$$X = \begin{pmatrix} x_1^1 & x_2^1 & \cdots & x_n^1 \\ x_1^2 & x_2^2 & \cdots & x_n^2 \\ \cdots & \cdots & & \\ x_1^p & x_2^p & \cdots & x_n^p \end{pmatrix}; \quad Y = \begin{pmatrix} y_1^1 & y_2^1 & \cdots & y_m^1 \\ y_1^2 & y_2^2 & \cdots & y_m^2 \\ \cdots & \cdots & & \\ y_1^p & y_2^p & \cdots & y_m^p \end{pmatrix}.$$

Then all $(\mathbf{x}_1, \mathbf{y}_1), ..., (\mathbf{x}_p, \mathbf{y}_p)$ can be stored in FAM (2.1) if and only if there is a fuzzy matrix $W = (w_{ij})_{n \times m}$, satisfying

$$X \circ W = Y. \tag{2.15}$$

(2.15) is a fuzzy relational equation based on '$\vee - \wedge$' composition [27–29, 41, 42]. All $(\mathbf{x}_1, \mathbf{y}_1), ..., (\mathbf{x}_p, \mathbf{y}_p)$ are memory patterns of FAM (2.1) if and only if the solutions of (2.15) exist. Moreover, using the following algorithm we can demonstrate the learning procedure for the connection weight matrix W of FAM (2.1), and establish a solution of (2.15).

Algorithm 2.1 Fuzzy δ−learning algorithm. With the following steps we can realize the iteration of w_{ij} for $i \in \mathrm{N}$, $j \in \mathrm{M}$:

Step 1. Initialization: $\forall i \in \mathrm{N}$, $j \in \mathrm{M}$, put $w_{ij}(0) = 1$ and $t = 0$;

Step 2. let $W(t) = (w_{ij}(t))_{n \times m}$;

Step 3. Calculate the real output: $Y(t) = X \circ W(t)$, that is

$$\forall k \in \mathrm{P}, \forall j \in \mathrm{M}, \;\; y_j^k(t) = \bigvee_{i=1}^{n} \{x_i^k \wedge w_{ij}(t)\}.$$

Step 4. Adjust the connection weights: Let $\eta \in (0, 1]$ be a learning constant, denote

$$w_{ij}(t+1) = \begin{cases} w_{ij}(t) - \eta \cdot (y_j^k(t) - y_j^k), & w_{ij}(t) \wedge x_i^k > y_j^k, \\ w_{ij}(t), & \text{otherwise.} \end{cases} \tag{2.16}$$

Step 5. $\forall i \in \mathrm{N}$, $j \in \mathrm{M}$, discriminate $w_{ij}(t+1) = w_{ij}(t)$? if yes stop; otherwise let $t = t + 1$, go to Step 2.

Preceding to analyze the convergence of the fuzzy δ−learning algorithm 2.1, we present an example to demonstrate the realizing procedure of the algorithm. To this end let $\mathrm{P} = \mathrm{N} = \{1, 2, 3, 4\}$, and $\mathrm{M} = \{1, 2, 3\}$. Give the fuzzy pattern pair family $\{(\mathbf{x}_k, \mathbf{y}_k) | k \in \mathrm{P}\}$ for training as follows:

$$\begin{aligned} \mathbf{x}_1 &= (0.3, 0.4, 0.5, 0.6), & \mathbf{y}_1 &= (0.6, 0.4, 0.5), \\ \mathbf{x}_2 &= (0.7, 0.2, 1.0, 0.1), & \mathbf{y}_2 &= (0.7, 0.7, 0.7), \\ \mathbf{x}_3 &= (0.4, 0.3, 0.9, 0.8), & \mathbf{y}_3 &= (0.8, 0.4, 0.5), \\ \mathbf{x}_4 &= (0.2, 0.1, 0.2, 0.3), & \mathbf{y}_4 &= (0.3, 0.3, 0.3). \end{aligned}$$

Then we can establish the fuzzy matrices X, Y in (2.15) as

$$X = \begin{pmatrix} 0.3 & 0.4 & 0.5 & 0.6 \\ 0.7 & 0.2 & 1.0 & 0.1 \\ 0.4 & 0.3 & 0.9 & 0.8 \\ 0.2 & 0.1 & 0.2 & 0.3 \end{pmatrix}; \quad Y = \begin{pmatrix} 0.6 & 0.4 & 0.5 \\ 0.7 & 0.7 & 0.7 \\ 0.8 & 0.4 & 0.5 \\ 0.3 & 0.3 & 0.3 \end{pmatrix}.$$

Choose $\eta = 0.8$, and with 40 iterations, the sequence of connection weight matrices $\{W(t)\}$ converges to the matrix W:

$$W^{\mathrm{T}} = \begin{pmatrix} 1.000000 & 1.000000 & 0.700000 & 1.000000 \\ 1.000000 & 1.000000 & 0.400000 & 0.400000 \\ 1.000000 & 1.000000 & 0.500000 & 0.500000 \end{pmatrix}.$$

Obviously (2.15) is true for W.

Theorem 2.7 *Suppose the fuzzy matrix sequence* $\{W(t)|t = 1, 2...\}$ *is obtained by Algorithm 2.1. Then*

(i) $\{W(t)|t = 1, 2, ...\}$ *is a non-increasing sequence of fuzzy matrices;*

(ii) $\{W(t)|t = 1, 2, ...\}$ *converges.*

Proof. (i) Let t mean the iteration step. For any $i \in \mathrm{N},\ j \in \mathrm{M}, k \in \mathrm{P}$, by (2.16), if $x_i^k \wedge w_{ij}(t) > y_j^k$, we get, $y_j^k(t) \geq x_i^k \wedge w_{ij}(t) > y_j^k$. Then $w_{ij}(t+1) = w_{ij}(t) - \eta(y_j^k(t) - y_j^k) < w_{ij}(t)$. If $x_i^k \wedge w_{ij}(t) \leq y_j^k$, then $w_{ij}(t+1) = w_{ij}(t)$. Therefore, for $i \in \mathrm{N},\ j \in \mathrm{M},\ w_{ij}(t+1) \leq w_{ij}(t), \Longrightarrow W(t+1) \subset W(t)$. That is, $\{W(t)|t = 1, 2, ...\}$ is a non-increasing fuzzy matrix sequence.

(ii) Since $\forall t = 1,\ 2, ...,\ w_{ij}(t) \in [0,\ 1]$, we have, $\forall i \in \mathrm{N},\ j \in \mathrm{M}$, the limit $\lim\limits_{t \to +\infty} w_{ij}(t)$ exists, that is, the matrix sequence $\{W(t)|t = 1,\ 2, ...\}$ converges. □

Theorem 2.7 can guarantee the convergence of Algorithm 2.1. If the solution set of (2.15) is non-empty, we in the following prove the limit matrix of the matrix sequence in Algorithm 2.1 is the maximum solution of (2.15).

Theorem 2.8 *For a given fuzzy pattern pair family* $\{(\mathbf{x}_k,\ \mathbf{y}_k)|k \in \mathrm{P}\}$, *the fuzzy matrix sequence* $\{W(t)|t = 1, 2, ...\}$ *defined by (2.16) converges to* $W_0 = (w_{ij}^0)_{n\times m}$, *where* w_{ij}^0 *can be defined by (2.9). Moreover, if* $M^w \neq \emptyset$, *then* $W_0 \in M^w$ *is the maximum element of* M^w; *if* $M^w = \emptyset$, *then* W_0 *is the maximum element of the set* $\{W \big| X \circ W \subset Y\}$.

Proof. For any $i \in \mathrm{N},\ j \in \mathrm{M}$, if $G_{ij}(\mathcal{X},\ \mathcal{Y}) = \emptyset$, then $\forall k \in \mathrm{P},\ x_i^k \leq y_j^k$. In Algorithm 2.1, for any iteration step t, we have, $w_{ij}(t) \wedge x_i^k \leq y_j^k$. Then by (2.16) it follows that

$$G_{ij}(\mathcal{X},\ \mathcal{Y}) = \emptyset, \Longrightarrow w_{ij}(t+1) = w_{ij}(t) = \cdots = w_{ij}(1) = w_{ij}(0) = 1. \quad (2.17)$$

If $G_{ij}(\mathcal{X},\ \mathcal{Y}) \neq \emptyset$, there is $k_0 \in G_{ij}(\mathcal{X},\ \mathcal{Y})$, so that $y_j^{k_0} = \bigwedge\limits_{k \in G_{ij}(\mathcal{X},\mathcal{Y})} \{y_j^k\}$. Then $w_{ij}(0) \wedge x_i^{k_0} = x_i^{k_0} > y_j^{k_0}$. Next let us use (2.16) to show that

$$\forall t \in \{1, 2, ...\},\ w_{ij}(t) > y_j^{k_0}. \quad (2.18)$$

In fact, if $t = 1$, considering $1 \geq x_i^{k_0} > y_j^{k_0}$ and (2.16) we get, $w_{ij}(1) = w_{ij}(0) - \eta \cdot \big(y_j^{k_0}(0) - y_j^{k_0}\big) = 1 - \eta \cdot \big(x_{i_0}^{k_0} - y_j^{k_0}\big) > y_j^{k_0}$, where $i_0 \in \mathrm{N}$ satisfies the following condition:

$$y_j^{k_0} = \bigvee_{i' \in \mathrm{N}} \{w_{i'j}(0) \wedge x_{i'}^{k_0}\} = \bigvee_{i' \in \mathrm{N}} \{x_{i'}^{k_0}\} = x_{i_0}^{k_0}.$$

If (2.18) is false, there is $t' \in \mathbb{N}$, so that $w_{ij}(t') \leq y_j^{k_0}$. Let $t_0 \in \mathbb{N}:\ t_0 = \max\{t \in \mathbb{N}|w_{ij}(t) > y_j^{k_0}\}$. Then we get, $t_0 \geq 1$, and $w_{ij}(t_0+1) \leq y_j^{k_0}$. So by (2.16) it follows that $x_i^{k_0} \wedge w_{ij}(t_0+1) \leq y_j^{k_0}, \Longrightarrow w_{ij}(t_0+2) = 1 > y_j^{k_0} \geq$

$w_{ij}(t_0+1)$, which is a contradiction, since by (i) of Theorem 2.7, the fuzzy matrix sequence $\{W(t)|t=1,2,...\}$ is non-increasing. Hence (2.18) is true. Therefore, $\lim_{t\longrightarrow+\infty} w_{ij}(t) \geq y_j^{k_0}$. If $\lim_{t\longrightarrow+\infty} w_{ij}(t) \triangleq l_{ij} > y_j^{k_0}$, then

$$y_j^{k_0}(t) = \bigvee_{i'\in \mathrm{N}} \{x_{i'}^{k_0} \wedge w_{i'j}(t)\}, \Longrightarrow \lim_{t\longrightarrow+\infty} y_j^{k_0}(t) = \bigvee_{i'\in \mathrm{N}} \{x_{i'}^{k_0} \wedge l_{i'j}\} > y_j^{k_0}. \quad (2.19)$$

Also by (2.16), $w_{ij}(t+1) = w_{ij}(t) - \eta\cdot\left(y_j^{k_0}(t) - y_j^{k_0}\right)$. Therefore, $l_{ij} = l_{ij} - \eta\cdot\left(\lim_{t\longrightarrow+\infty} y_j^{k_0}(t) - y_j^{k_0}\right)$, and $\lim_{t\longrightarrow+\infty} y_j^{k_0}(t) = y_j^{k_0}$, which contradicts (2.19). So $G_{ij}(\mathcal{X},\mathcal{Y}) \neq \emptyset, \Longrightarrow \lim_{t\longrightarrow+\infty} w_{ij}(t) = y_j^{k_0}$. Considering (2.17) we can conclude that

$$\lim_{t\longrightarrow+\infty} w_{ij}(t) = \left\{\begin{array}{ll} \bigwedge_{k\in G_{ij}(\mathcal{X},\mathcal{Y})} \{y_j^k\}, & G_{ij}(\mathcal{X},\mathcal{Y}) \neq \emptyset, \\ 1, & G_{ij}(\mathcal{X},\mathcal{Y}) = \emptyset, \end{array}\right\} = w_{ij}^0.$$

So the first part of the theorem holds. And the other part of the theorem is a direct result of Theorem 2.1. □

In Algorithm 2.1 if we choose the learning constant η as an adjustable value changing with the iteration step t, that is $\eta(t)$, then the convergence speed of the algorithm can speed up, strikingly. We choose

$$\eta = \eta(t) = \frac{w_{ij}(t) - w_{ij}(t)\wedge y_j^k}{y_j^k(t) - y_j^k}.$$

Then (2.16) is transformed into the following iteration scheme:

$$w_{ij}(t+1) = \begin{cases} w_{ij}(t)\wedge y_j^k, & w_{ij}(t)\wedge x_i^k > y_j^k, \\ w_{ij}(t), & \text{otherwise.} \end{cases} \quad (2.20)$$

By Theorem 2.8 the following theorem is trivial.

Theorem 2.9 *Let* $\{(\mathbf{x}_k, \mathbf{y}_k)|k\in \mathrm{P}\}$ *be a fuzzy pattern pair family, and* $W(t) = (w_{ij}(t))_{n\times m}$ *be a fuzzy matrix defined by (2.20). Then the sequence* $\{W(t)|t=1,2,...\}$ *converges to* $W_0 = (w_{ij}^0)_{n\times m}$ *as* $t\longrightarrow+\infty$, *where* w_{ij}^0 *is defined by (2.9). Moreover, if* $M^w \neq \emptyset$, *then* $W_0\in M^w$ *is a maximum element of* M^w; *if* $M^w = \emptyset$, *then* W_0 *is a maximum element of the* $\{W\big|X\circ W \subset Y\}$.

2.2.2 FAM's based on '$\vee - *$'

Since the fuzzy operator pair '$\vee - \wedge$' can not treat many real problems, it is necessary to study the FAM's based on other fuzzy operator pairs. To this end we at first present the following definition [18, 35, 42, 52].

Definition 2.1 We call the mapping $T:\ [0,1]^2 \longrightarrow [0,1]$ a fuzzy operator, if the following conditions hold:

(1) $T(0,0)=0,\ T(1,1)=1$;
(2) If $a,\ b,\ c,\ d\in[0,1]$, then $a\le c,\ b\le d,\Longrightarrow T(a,b)\le T(c,d)$;
(3) $\forall a,\ b\in[0,1],\ T(a,b)=T(b,a)$;
(4) $\forall a,\ b,\ c\in[0,1],\ T(T(a,b),c)=T(a,T(b,c))$.

If T is a fuzzy operator, and $\forall a\in[0,1],\ T(a,1)=a$, we call T a t−norm; If the fuzzy operator T satisfies: $\forall a\in[0,1],\ T(0,a)=a$, we call T a t−conorm. From now on, we denote $T(a,b)=a\,T\,b$, and write the t−norm T as '$*$'.

For $a,\ b\in[0,1]$, define $a\,\alpha_* b\in[0,1]:\ a\,\alpha_*\,b=\sup\{x\in[0,1]|a\,T\,x\le b\}$. Let us now present some useful properties of the operator 'α_*', the further discussions can see [18, 40, 52].

Lemma 2.2 *Let $a,\ b,\ a_1,\ b_1\in[0,1]$, and T be a t−norm. Then*
(i) $a*(a\,\alpha_*\,b)\le b,\ a\,\alpha_*\,(a*b)\ge b,\ (a\,\alpha_*\,b)\,\alpha_*\,b\ge a$;
(ii) $a\le a_1,\Longrightarrow a\,\alpha_*\,b\ge a_1\,\alpha_*\,b$;
(iii) $b\le b_1,\Longrightarrow a\,\alpha_*\,b\le a\,\alpha_*\,b_1$;

Proof. (i) By the definition of the operator 'α_*' it follows that $a*(a\,\alpha_*\,b)\le b$. Since $a*b\le a*b$, also using the definition of 'α_*' we get, $a\,\alpha_*\,(a*b)\ge b$. Moreover, considering that $a*(a\,\alpha_*\,b)\le b,\Longrightarrow(a\,\alpha_*\,b)*a\le b$, we can conclude that, $(a\,\alpha_*\,b)\,\alpha_*\,b\ge a$. (i) is true. As for (ii) (iii), they are also the direct results of the definition of 'α_*'. □

In (2.1) we substitute the t−norm '$*$' for '$\wedge$', and get a FAM based on the fuzzy operator pair '$\vee-*$':

$$y_j=\bigvee_{i\in\mathrm{N}}\{x_i*w_{ij}\}\ (j\in\mathrm{M}). \tag{2.21}$$

If choose '$\circledast$' as the '$\vee-*$' composition operation, then (2.21) becomes as $\mathbf{y}=\mathbf{x}\circledast W$. Similar with (2.1), we can in (2.21) develop some analytic learning algorithms and iterative learning algorithms for the connection weight matrix W. For a given fuzzy pattern pair family $\{(\mathbf{x}_k,\ \mathbf{y}_k)|k\in\mathrm{P}\}$, we can obtain a conclusion for the FAM (2.21) being similar with Theorem 2.1. To this end we at first design an analytic learning algorithm for W. Define $W_*=(w^*_{ij})_{n\times m}\in\mu_{n\times m}$ as follows:

$$w^*_{ij}=\bigwedge_{k\in\mathrm{P}}\{x^k_i\,\alpha_*\,y^k_j\}\ (i\in\mathrm{N},\ j\in\mathrm{M}). \tag{2.22}$$

Recalling $S^G_{ij}(W_0,\mathcal{Y})$ and M^w we may introduce the sets $S^{*G}_{ij}(W_*,\mathcal{X},\mathcal{Y})$ $(i\in\mathrm{N},\ j\in\mathrm{M})$ and M^w_* respectively as follows:

$$S^{*G}_{ij}(W_*,\mathcal{X},\mathcal{Y})=\{k\in\mathrm{P}|x^k_i*w^*_{ij}\ge y^k_j\},$$
$$M^w_*=\{W\in\mu_{n\times m}|\forall k\in\mathrm{P},\ \mathbf{x}_k\circledast W=\mathbf{y}_k\}.$$

Theorem 2.10 *Given a fuzzy pattern pair family $(\mathcal{X},\ \mathcal{Y})\{(\mathbf{x}_k,\ \mathbf{y}_k)|k\in\mathrm{P}\}$, and $W_*=(w^*_{ij})_{n\times m}$ is defined by (2.22). Then* *(i)* $\forall k \in \mathrm{P}, \mathbf{x}_k \circledast W_* \subset \mathbf{y}_k$, *and if the fuzzy matrix* W *satisfies:* $\forall k \in \mathrm{P}, \mathbf{x}_k \circledast W \subset \mathbf{y}_k$, *we have,* $W \subset W_*$;

(ii) If $M_*^w \neq \emptyset$, *it follows that* $W_* \in M_*^w$, *and* $\forall\, W = (w_{ij})_{n\times m} \in M_*^w$, $W \subset W_*$, *i.e.* $\forall i \in \mathrm{N}, j \in \mathrm{M}, w_{ij} \leq w_{ij}^*$;

(iii) The set $M_*^w \neq \emptyset$ *if and only if* $\forall j \in \mathrm{M}, \bigcup\limits_{i\in \mathrm{N}} S_{ij}^{*G}(W_*, \mathcal{X}, \mathcal{Y}) = \mathrm{P}$.

Proof. (i) For any $k \in \mathrm{P}$, and $j \in \mathrm{M}$, by Lemma 2.2 and (2.22) it follows that the following inequalities hold:

$$\bigvee_{i\in \mathrm{N}} \{x_i^k * w_{ij}^*\} = \bigvee_{i\in \mathrm{N}} \Big\{x_i^k * \Big(\bigwedge_{k'\in \mathrm{P}} \{x_i^{k'} \alpha_* y_j^{k'}\}\Big)\Big\} \leq \bigvee_{i\in \mathrm{N}} \{x_i^k * (x_i^k \alpha_* y_j^k)\} \leq y_j^k,$$

that is, $\mathbf{x}_k \circledast W_* \subset \mathbf{y}_k$. And if $W = (w_{ij})_{n\times m} \in \mu_{n\times m}$ satisfies: $\mathbf{x}_k \circledast W \subset \mathbf{y}_k$ $(k \in \mathrm{P})$, for any $k \in \mathrm{P}$, $j \in \mathrm{M}$, we can conclude that

$$\bigvee_{i\in \mathrm{N}} \{x_i^k * w_{ij}\} \leq y_j^k, \Longrightarrow x_i^k * w_{ij} \leq y_j^k, \Longrightarrow w_{ij} \leq x_i^k \alpha_* y_j^k.$$

Therefore, $w_{ij} \leq \bigwedge\limits_{k\in \mathrm{P}} \{x_i^k \alpha_* y_j^k\} = w_{ij}^*$. So $W \subset W_*$. Thus, (i) is true.

(ii) Let $W = (w_{ij})_{n\times m} \in M_*^w$. Then $\forall k \in \mathrm{P}$, $\mathbf{x}_k \circledast W = \mathbf{y}_k$. Similarly with (i) we can show, $W \subset W_*$. And for any $j \in \mathrm{M}$, $k \in \mathrm{P}$, the following fact holds:

$$\bigvee_{i\in \mathrm{N}} \{x_i^k * w_{ij}\} = y_j^k, \Longrightarrow \bigvee_{i\in \mathrm{N}} \{x_i^k * w_{ij}^*\} \geq y_j^k.$$

Lemma 2.2 and (2.22) can imply the following conclusion:

$$x_i^k * w_{ij}^* \leq x_i^k * (x_i^k \alpha_* y_j^k) \leq y_j^k, \Longrightarrow \bigvee_{i\in \mathrm{N}} \{x_i^k * w_{ij}^*\} \leq y_j^k.$$

Therefore, $\bigvee\limits_{i\in \mathrm{N}} \{x_i^k * w_{ij}^*\} = y_j^k$. That is, $W_* \in M_*^w$. (ii) is proved.

(iii) At first assume $M_*^w \neq \emptyset$, and $W = (w_{ij})_{n\times m} \in M_*^w$. By (i) we have, $W_* = (w_{ij}^*)_{n\times m} \in M_*^w$, moreover $W \subset W_*$. Then suppose there is $j_0 \in \mathrm{M}$, satisfying $\bigcup\limits_{i\in \mathrm{N}} S_{ij_0}^{*G}(W_*, \mathcal{X}, \mathcal{Y}) \neq \mathrm{P}$. So there is $k_0 \in \mathrm{P}$, so that $\forall i \in \mathrm{N}$, $k_0 \notin S_{ij_0}^{*G}(W_*, \mathcal{X}, \mathcal{Y})$. Then we can imply the following facts: $w_{ij_0} * x_i^{k_0} < y_{j_0}^{k_0}$; $\bigvee\limits_{i\in \mathrm{N}} \{x_i^{k_0} * w_{ij_0}\} < y_{j_0}^{k_0}$, which is a contradiction since $W \in M^w$. Thus, $\forall j \in \mathrm{M}$, $\bigcup\limits_{i\in \mathrm{N}} S_{ij}^{*G}(W_*, \mathcal{X}, \mathcal{Y}) = \mathrm{P}$. Conversely, let $\bigcup\limits_{i\in \mathrm{N}} S_{ij}^{*G}(W_*, \mathcal{X}, \mathcal{Y}) = \mathrm{P}$ $(j \in \mathrm{M})$. For any $j \in \mathrm{M}$, $k \in \mathrm{P}$, there exists $i_0 \in \mathrm{N}$, satisfying $k \in S_{i_0j}^{*G}(W_*, \mathcal{X}, \mathcal{Y})$. And so $x_{i_0}^k * w_{i_0j}^* \geq y_j^k, \Longrightarrow \bigvee\limits_{i\in \mathrm{N}} \{x_i^k * w_{ij}^*\} \geq y_j^k$. By the definition of w_{ij}^*, (2.22) and Lemma 2.2 it is easy to show, $\forall i \in \mathrm{N}$, $j \in \mathrm{M}$, $k \in \mathrm{P}$, we have $x_i^k * w_{ij}^* = x_i^k * \big(\bigwedge\limits_{k'\in \mathrm{P}} \{x_i^{k'} \alpha_* y_j^{k'}\}\big) \leq x_i^k * (x_i^k \alpha_* y_j^k) \leq y_j^k$. Thus, $\bigvee\limits_{i\in \mathrm{N}} \{x_i^k \wedge w_{ij}^*\} \leq y_j^k$. Therefore, $W_0 \in M_*^w$, $\Longrightarrow M_*^w \neq \emptyset$. (iii) is true. $\square$

Next let us illustrate the application of Theorem 2.10 by a simulation example. Using the analytic learning algorithm (2.22) to establish the connection weight matrix W_*. Let P = N = {1, 2, 3, 4}, M = {1, 2, 3}, and the t−norm $*$ is defined as follows: $\forall a,\, b \in [0,1]$, $a * b = \max\{a+b-1,\, 0\}$. Give the fuzzy pattern pair family $\{(\mathbf{x}_k,\, \mathbf{y}_k)|k \in \mathrm{P}\}$:

$$\begin{aligned}\mathbf{x}_1 &= (0.6, 0.5, 0.4, 0.3), \quad \mathbf{y}_1 = (0.3, 0.6, 0.1);\\ \mathbf{x}_2 &= (0.5, 0.7, 0.8, 0.6), \quad \mathbf{y}_2 = (0.6, 0.5, 0.4);\\ \mathbf{x}_3 &= (0.4, 0.7, 0.6, 0.4), \quad \mathbf{y}_3 = (0.4, 0.5, 0.2);\\ \mathbf{x}_4 &= (0.8, 0.9, 0.7, 0.3), \quad \mathbf{y}_4 = (0.5, 0.8, 0.4).\end{aligned}$$

For $i \in \mathrm{N}$, $j \in \mathrm{M}$, $k \in \mathrm{P}$, by the definition of '$*$' it is easy to show

$$\begin{aligned}x_i^k\, \alpha_*\, y_j^k &= \sup\{x \in [0,1] | x_i^k * x \le y_j^k\}\\ &= \sup\{x \in [0,1] | x_i^k + x - 1 \le y_j^k\} = \min\{1 + y_j^k - x_i^k,\, 1\}.\end{aligned}$$

Therefore, by (2.22), $w_{ij}^* = \bigwedge_{k\in\mathrm{P}} \{x_i^k\, \alpha_*\, y_j^k\} = \bigwedge_{k\in\mathrm{P}} \{\min\{1 + y_j^k - x_i^k,\, 1\}\}$. So we get $W_* = (w_{ij}^*)_{4\times 3}$:

$$W_*^{\mathrm{T}} = \begin{pmatrix} 0.7 & 0.6 & 0.8 & 1.0 \\ 1.0 & 0.8 & 0.7 & 0.9 \\ 0.5 & 0.5 & 0.6 & 0.8 \end{pmatrix}$$

Moreover, for $i \in \mathrm{N}$, $j \in \mathrm{M}$, $S_{ij}^{*G}(W_*, \mathcal{X}, \mathcal{Y}) = \{k \in \mathrm{P} | x_i^k * w_{ij}^* \ge y_j^k\}$. Easily we have, $\forall j \in \mathrm{M}$, $\bigcup_{i\in\mathrm{N}} S_{ij}^{*G}(W_*, \mathcal{X}, \mathcal{Y}) = \mathrm{P}$. Therefore, by Theorem 2.10, Each pattern pair in $\{(\mathbf{x}_k,\, \mathbf{y}_k)|k \in \mathrm{P}\}$ can be stored in the FAM (2.21), and the corresponding connection weight matrix is W_*.

Similarly with Algorithm 2.1, we can develop an iteration scheme for learning the connection weight matrix W of the FAM (2.21), that is

Algorithm 2.2 Fuzzy δ−learning algorithm based on t−norm. With the following steps we can establish the connection weights of FAM (2.21):

Step 1. Initialization: put $t = 0$, and $w_{ij}(t) = 1$;

Step 2. Let $W(t) = (w_{ij}(t))_{n\times m}$;

Step 3. Calculate the real output: $\mathbf{y}_k(t) = \mathbf{x}_k \circledast W(t)$, i.e.

$$y_j^k(t) = \bigvee_{i\in\mathrm{N}} \{x_i^k * w_{ij}(t)\} \;\; (j \in \mathrm{M}, k \in \mathrm{P}).$$

Step 4. Iteration scheme: The connection weights iterate with the following scheme (where $\eta \in (0,1]$ is a learning constant):

$$w_{ij}(t+1) = \begin{cases} w_{ij}(t) - \eta \cdot (y_j^k(t) - y_j^k), & w_{ij}(t) * x_i^k > y_j^k, \\ w_{ij}(t), & \text{otherwise}. \end{cases} \tag{2.23}$$

Step 5. Discriminate $W(t+1)=W(t)$? if yes, stop; otherwise let $t=t+1$ go to Step 2.

Similarly with the equation (2.15), the FAM (2.21) can store each fuzzy pattern pair in $\{(\mathbf{x}_k,\ \mathbf{y}_k)|k\in \mathrm{P}\}$ if and only if the following equalities hold:

$$\begin{pmatrix} x_1^1 & x_2^1 & \cdots & x_n^1 \\ x_1^2 & x_2^2 & \cdots & x_n^2 \\ \cdots & \cdots & & \\ x_1^p & x_2^p & \cdots & x_n^p \end{pmatrix} \circledast \begin{pmatrix} w_{11} & w_{12} & \cdots & w_{1m} \\ w_{21} & w_{22} & \cdots & w_{2m} \\ \cdots & \cdots & & \\ w_{n1} & w_{n2} & \cdots & w_{nm} \end{pmatrix} = \begin{pmatrix} y_1^1 & y_2^1 & \cdots & y_m^1 \\ y_1^2 & y_2^2 & \cdots & y_m^2 \\ \cdots & \cdots & & \\ y_1^p & y_2^p & \cdots & y_m^p \end{pmatrix}, \tag{2.24}$$

that is, $X\circledast W=Y$. By Theorem 2.8 and Theorem 2.9, we can get the following result.

Theorem 2.11 *Let $\{(\mathbf{x}_k,\mathbf{y}_k)|k\in\mathrm{P}\}$ be a given fuzzy pattern pair family, and $\{W(t)|t=1,2,...\}$ be a sequence of fuzzy matrices obtained by Algorithm 2.2. The t−norm '$*$' is continuous as a two-variate function. Then*

(i) $\forall\, t=1,2,...,$ $W(t+1)\subset W(t)$, and so $\{W(t)|t=1,2,...\}$ converges;

(ii) If there is $W\in\mu_{n\times m}$, so that (2.24) holds, then $\{W(t)|t=1,2,...\}$ converges to the maximum solution of (2.24), i.e. $\lim\limits_{t\to+\infty}W(t)=W_*=(w_{ij}^*)_{n\times m}$;

(iii) If $\forall\, W\in\mu_{n\times m}$, (2.24) is false, then $\{W(t)|t=1,2,...\}$ converges to the maximum solution of $X\circledast W\subset Y$ as $t\to+\infty$.

Proof. it suffices to prove (ii), since the proofs of (i) (iii) are similar with ones of Theorem 2.8 by Theorem 2.10.

There is $k_0\in G_{ij}(\mathcal{X},\,\mathcal{Y})$, satisfying $x_i^{k_0}\,\alpha_*\,y_j^{k_0}=\bigwedge\limits_{k\in\mathrm{P}}\{x_i^k\,\alpha_*\,y_j^k\}$. If $x_i^{k_0}\le y_j^{k_0}$, then we can conclude that

$$\forall t=1,2,...,\ x_i^{k_0}\,*\,w_{ij}(t)\le y_j^{k_0},\Longrightarrow w_{ij}(1)=w_{ij}(2)=\cdots=1. \tag{2.25}$$

if $x_i^{k_0}>y_j^{k_0}$, by the continuity of the t−norm '$*$', $x_i^{k_0}\,\alpha_*\,y_j^{k_0}<1$. Similarly with (2.18), let us next to prove by (2.16) that

$$\forall t\in\{0,1,2,...\},\ w_{ij}(t)>x_i^{k_0}\,\alpha_*\,y_j^{k_0}. \tag{2.26}$$

In fact, if $t=0$, then $w_{ij}(0)=1>x_i^{k_0}\,\alpha_*\,y_j^{k_0}$. And if there is $t_1\in\mathbb{N}$, so that $w_{ij}(t_1)\le x_i^{k_0}\,\alpha_*\,y_j^{k_0}$, let $t_0=\max\{t\in\{0,1,...\}|w_{ij}(t)\le x_i^{k_0}\,\alpha_*\,y_j^{k_0}\}$. Then $t_0>0$, and $w_{ij}(t_0+1)\le x_i^{k_0}\,\alpha_*\,y_j^{k_0}$. By Lemma 2.2 we get, $w_{ij}(t_0+1)\,*\,x_i^{k_0}\le x_i^{k_0}\,*\,(x_i^{k_0}\,\alpha_*\,y_j^{k_0})\le y_j^{k_0}$. So (2.23) may imply, $w_{ij}(t_0+2)=1>w_{ij}(t_0+1)$, which contradicts (i), i.e. $\{W(t)|t=0,1,...\}$ is non-increasing. Therefore (2.26) is true. Thus, $\lim\limits_{t\to+\infty}w_{ij}(t)\ge x_i^{k_0}\,\alpha_*\,y_j^{k_0}=w_{ij}^*$. If $\lim\limits_{t\to+\infty}w_{ij}(t)\triangleq l_{ij}>x_i^{k_0}\,\alpha_*\,y_j^{k_0}=w_{ij}^*$, then by the definition of 'α_*', $x_i^{k_0}\,*\,l_{ij}>y_j^{k_0}$, so

$$y_j^{k_0}(t)=\bigvee_{i'\in\mathrm{N}}\{x_{i'}^{k_0}\,*\,w_{i'j}(t)\},\Longrightarrow\lim_{t\to+\infty}y_j^{k_0}(t)=\bigvee_{i'\in\mathrm{N}}\{x_{i'}^{k_0}\,*\,l_{i'j}\}>y_j^{k_0}. \tag{2.27}$$

And by (2.23) (2.26) and the definition of 'α_*' it follows that

$$w_{ij}(t+1) = w_{ij}(t) - \eta \cdot \big(y_j^{k_0}(t) - y_j^{k_0}\big),$$
$$\Longrightarrow l_{ij} = l_{ij} - \eta \cdot \big(\lim_{t\to+\infty} y_j^{k_0}(t) - y_j^{k_0}\big), \Longrightarrow \lim_{t\to+\infty} y_j^{k_0}(t) = y_j^{k_0},$$

which contradicts (2.27). Thus, $x_i^{k_0} > y_j^{k_0}, \Longrightarrow \lim_{t\to+\infty} w_{ij}(t) = x_i^{k_0}\,\alpha_*\,y_j^{k_0}$. Hence by (2.25), $\lim_{t\to+\infty} w_{ij}(t) = x_i^{k_0}\,\alpha_*\,y_j^{k_0} = w_{ij}^*$. □

Next we discuss an application of Algorithm 2.2. Define the t−norm $*$ as follows [27–29]:

$$a * b = \max\{0, a+b-1\} \ (a,\, b \in [0,\, 1]).$$

And P = {1, 2, 3}, N = {1, 2, 3}, M = {1}. Let

$$X = \begin{pmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \mathbf{x}_3 \end{pmatrix} = \begin{pmatrix} 0.8 & 0.0 & 0.4 \\ 0.5 & 0.1 & 0.3 \\ 1.0 & 0.2 & 0.1 \end{pmatrix}; \ Y = \begin{pmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \\ \mathbf{y}_3 \end{pmatrix} = \begin{pmatrix} 0.5 \\ 0.2 \\ 0.7 \end{pmatrix}.$$

By Theorem 2.10, the fuzzy pattern pair family $\{(\mathbf{x}_k,\, \mathbf{y}_k) | k = 1,2,3\}$ can be stored in the FAM (2.21). So the maximum solution of $X \circledast W = Y$ exists, that is $W_* = (0.7, 1.0, 0.9)^{\mathrm{T}}$. Table 2.2 shows the iteration step number of Algorithm 2.2 with different learning constants η's and the ultimate connection weight matrix W :

Table 2.2 Simulation results of Algorithm 2.2

No.	learning constant (η)	iteration (t)	converged matrix (W)
1	0.90000	5	$(0.64000, 1.00000, 0.90000)^{\mathrm{T}}$
2	0.80000	7	$(0.68000, 1.00000, 0.90000)^{\mathrm{T}}$
3	0.50000	16	$(0.69531, 1.00000, 0.90000)^{\mathrm{T}}$
4	0.30000	30	$(0.69948, 1.00000, 0.90000)^{\mathrm{T}}$
5	0.10000	86	$(0.69999, 1.00000, 0.90000)^{\mathrm{T}}$
6	0.01000	594	$(0.70000, 1.00000, 0.90000)^{\mathrm{T}}$

By Table 2.2 we may see, the larger the learning constant η is , the quicker the convergent speed of the matrix sequence $\{W(t)|t = 1, 2, ...\}$ is. The limit value W and the maximum solution W^* of (2.20) are not completely identical, for example, when $\eta = 0.9$ and $\eta = 0.8$, the difference between W and W^* is obvious. As η becomes smaller and smaller, W is close to W^*, gradually. When $\eta = 0.01$, we get $W = W^*$. Therefore, a meaningful problem related to Algorithm 2.2 is how to determine η so that the convergent speed and the sufficient closeness between W and W^* can be guaranteed, simultaneously.

§2.3 BP learning algorithm of FAM's

In the section we present the back propagation (BP) algorithm for the connection weight matrix W of the FAM (2.1). Since for a given $a \in [0,\ 1]$, the functions $a\vee x$ and $a\wedge x$ are not differentiable on $[0,\ 1]$ (see [51]), as a preliminary for the BP algorithm of FAM's we at first define the differentiable functions 'La' and 'Sm', by which the fuzzy operators '$\vee$' and '$\wedge$' can be approximated, respectively and the derivatives in the BP algorithm can be calculated.

2.3.1 Two analytic functions

Next we build the approximately analytic representations of the fuzzy operators '$\vee$' and '$\wedge$', respectively, and establish the partial derivatives related. Define $d-$variate functions $La,\ Sm:\ \mathbb{R}_+ \times [0,1]^d \longrightarrow [0,1]$ as

$$La(s;x_1,...,x_d) = \frac{\sum\limits_{i=1}^{d} x_i \cdot \exp\{sx_i\}}{\sum\limits_{i=1}^{d} \exp\{sx_i\}};\quad Sm(s;x_1,...,x_d) = \frac{\sum\limits_{i=1}^{d} x_i \cdot \exp\{-sx_i\}}{\sum\limits_{i=1}^{d} \exp\{-sx_i\}}. \tag{2.28}$$

By (2.28) it is easy to show, $\forall x_1,...,x_d \in [0,1]$, the following facts hold:

$$\forall s>0,\ La(s;x_1,...,x_d),\ \ Sm(s;x_1,...,x_d) \in \big[x_1\wedge\cdots\wedge x_d,\ x_1\vee\cdots\vee x_d\big].$$

For $x_1,...,x_d \in [0,1]$, denote $x_{\min} \triangleq x_1\wedge\cdots\wedge x_d,\ x_{\max} \triangleq x_1\vee\cdots\vee x_d$. Moreover

$$x_\delta = \begin{cases} \max\{x\in\{x_1,...,x_d\}\big|x<x_{\max}\}, & \text{this set nonempty,}\\ x_{\max}, & \text{this set empty;}\end{cases}$$

$$x_\rho = \begin{cases} \min\{x\in\{x_1,...,x_d\}\big|x>x_{\min}\}, & \text{this set nonempty,}\\ x_{\min}, & \text{this set empty;}\end{cases}$$

Lemma 2.3 *Suppose $d>1,\ s>o$, and $x_1,...,x_d \in [0,1]$. Then we have the following estimations:*

$$\Big|\bigvee_{i=1}^{d}\{x_i\} - La(s;x_1,...,x_d)\Big| \le (d-1)\cdot\exp\big\{-s(x_{\max}-x_\delta)\big\};$$
$$\Big|\bigwedge_{i=1}^{d}\{x_i\} - Sm(s;x_1,...,x_d)\Big| \le (d-1)\cdot\exp\big\{-s(x_\rho-x_{\min})\big\}.$$

Therefore, $\lim\limits_{s\to+\infty} La(s;x_1,...,x_d) = \bigvee\limits_{i=1}^{d}\{x_i\},\ \lim\limits_{s\to+\infty} Sm(s;x_1,...,x_d) = \bigwedge\limits_{i=1}^{d}\{x_i\}.$

Proof. At first let $x_{\max} > x_{\min}$, and $x_1 = \cdots = x_q = x_{\max}$. Then $\forall i \in$

$\{q+1,...,d\}$, $x_i < x_{\max}$. Thus, when $s > 0$, we can conclude that

$$\left|\bigvee_{i=1}^{d}\{x_i\} - La(s;x_1,...,x_d)\right| = \left|\frac{\sum_{i=1}^{d} x_i \exp\{sx_i\}}{\sum_{i=1}^{d}\exp\{sx_i\}} - x_{\max}\right|$$

$$= \left|\frac{\sum_{i=1}^{d}(x_i - x_{\max})\exp\{sx_i\}}{\sum_{i=1}^{d}\exp\{sx_i\}}\right| \le \frac{\sum_{i=1}^{d}|x_i - x_{\max}|\exp\{sx_i\}}{\sum_{i=1}^{d}\exp\{sx_i\}}$$

$$= \frac{\sum_{i=1}^{d}|x_i - x_{\max}|\exp\{s(x_i - x_{\max})\}}{\sum_{i=1}^{d}\exp\{s(x_i - x_{\max})\}} = \frac{\sum_{i=q+1}^{d}|x_i - x_{\max}|\exp\{s(x_i - x_{\max})\}}{\sum_{i=1}^{d}\exp\{s(x_i - x_{\max})\}}$$

$$= \frac{\sum_{i=q+1}^{d}|x_i - x_{\max}|\exp\{s(x_i - x_{\max})\}}{q + \sum_{i=q+1}^{d}\exp\{s(x_i - x_{\max})\}} < \frac{|x_{\max} - x_{\min}|}{q}\sum_{i=q+1}^{d}\exp\{s(x_i - x_{\max})\}$$

$$\le \frac{1}{q}\sum_{i=q+1}^{d}\exp\{-s\cdot(x_{\max} - x_\delta)\} \le \frac{d-q}{q}\cdot\exp\{-s\cdot(x_{\max} - x_\delta)\}$$

$$\le (d-q)\cdot\exp\{-s\cdot(x_{\max} - x_\delta)\} \le (d-1)\cdot\exp\{-s\cdot(x_{\max} - x_\delta)\}.$$

So if $x_{\max} > x_{\min}$, then we can obtain the following limit:

$$\lim_{s\to+\infty}\left|La(s;x_1,...,x_d) - \bigvee_{i=1}^{d}\{x_i\}\right| \le \lim_{s\to+\infty}(d-1)\cdot\exp\{-s\cdot(x_{\max} - x_\delta)\} = 0$$

when $x_{\max} > x_\delta$. And therefore, $\lim_{s\to+\infty} La(s;x_1,...,x_d) = \bigvee_{i=1}^{d}\{x_i\}$; If $x_{\max} = x_{\min}$, then $x_1 = \cdots = x_d, \Longrightarrow La(s;x_1,...,x_d) = x_1 = \bigvee_{i=1}^{d}\{x_i\}$. Hence the limit of $La(s;x_1,...,x_d)$ as $s \to +\infty$ exists, and $\lim_{s\to+\infty} La(s;x_1,...,x_d) = \bigvee_{i=1}^{d}\{x_i\}$. The first part of the theorem is proved. Similarly we can prove the other conclusions. □

Lemma 2.4 *The functions $La(s;x_1,...,x_d)$ and $Sm(s;x_1,...,x_d)$ are continuously differentiable on $[0,1]^d$. Moreover, for $j \in \{1,...,d\}$, we have*

(i) $\dfrac{\partial La(s;x_1,...,x_d)}{\partial x_j}=\dfrac{-\exp(sx_j)}{\Big(\sum\limits_{i=1}^{d}\exp(sx_i)\Big)^2}\Big\{\sum\limits_{i=1}^{d}(sx_i-sx_j-1)\exp(sx_i)\Big\};$

(ii) $\dfrac{\partial Sm(s;x_1,...,x_d)}{\partial x_j}=\dfrac{\exp(-sx_j)}{\Big(\sum\limits_{i=1}^{d}\exp(-sx_i)\Big)^2}\Big\{\sum\limits_{i=1}^{d}(sx_i-sx_j+1)\exp(-sx_i)\Big\}.$

Proof. It suffices to show (i) since the proof of (ii) is similar. By the definition of $La(s;x_1,...,x_d)$ we can prove

$$\frac{\partial La(s;x_1,...,x_d)}{\partial x_j}=\sum_{i=1,i\neq j}^{d}\frac{\partial}{\partial x_j}\left(\frac{x_i\cdot\exp(sx_i)}{\sum\limits_{i=1}^{d}\exp(sx_i)}\right)+\frac{\partial}{\partial x_j}\left(\frac{x_j\cdot\exp(sx_j)}{\sum\limits_{i=1}^{d}\exp(sx_i)}\right)$$
$$=\frac{-\exp(sx_j)}{\Big(\sum\limits_{i=1}^{d}\exp(sx_i)\Big)^2}\Big\{sx_j\exp(sx_j)+\sum_{i=1,i\neq j}^{d}sx_i\exp(sx_i)-(1+sx_j)\sum_{i=1}^{d}\exp(sx_i)\Big\}$$
$$=\frac{-\exp(sx_j)}{\Big(\sum\limits_{i=1}^{d}\exp(sx_i)\Big)^2}\Big\{\sum_{i=1}^{d}(sx_i-sx_j-1)\exp(sx_i)\Big\}.$$

So (i) is true. □

By Lemma 2.4, we may conclude that the following facts hold for the constant $a\in[0,1]$:

$$\begin{cases}\dfrac{\mathrm{d}La(s;x,a)}{\mathrm{d}x}=\dfrac{1}{(1+\exp(s(a-x)))^2}\big\{1-(sa-sx-1)\exp(s(a-x))\big\};\\[2ex] \dfrac{\mathrm{d}Sm(s;x,a)}{\mathrm{d}x}=\dfrac{1}{(1+\exp(-s(a-x)))^2}\big\{1+(sa-sx+1)\exp(-s(a-x))\big\}.\end{cases}\tag{2.29}$$

Therefore, $x>a,\Longrightarrow\lim\limits_{s\to+\infty}\big(\mathrm{d}La(s;x,a)/\mathrm{d}x\big)=1,\ \lim\limits_{s\to+\infty}\big(\mathrm{d}Sm(s;x,a)/\mathrm{d}x\big)=0$; and $x<a,\Longrightarrow\lim\limits_{s\to+\infty}\big(\mathrm{d}La(s;x,a)/\mathrm{d}x\big)=0;\ \lim\limits_{s\to+\infty}\big(\mathrm{d}Sm(s;x,a)/\mathrm{d}x\big)=1$. So for a given constant $a\in[0,\ 1]$, it follows that

$$\lim_{s\to+\infty}\frac{\mathrm{d}La(s;x,a)}{\mathrm{d}x}=\begin{cases}\dfrac{\mathrm{d}(a\vee x)}{\mathrm{d}x}, & x\neq a,\\[1ex] \dfrac{1}{2}, & x=a;\end{cases}\tag{2.30}$$

$$\lim_{s\to+\infty}\frac{\mathrm{d}Sm(s;x,a)}{\mathrm{d}x}=\begin{cases}\dfrac{\mathrm{d}(a\wedge x)}{\mathrm{d}x}, & x\neq a,\\[1ex] \dfrac{1}{2}, & x=a.\end{cases}\tag{2.31}$$

2.3.2 BP learning algorithm

To develop the BP learning algorithm for the connection weight matrix W of (2.1), firstly we define a suitable error function. Suppose $\{(\mathbf{x}_k,\ \mathbf{y}_k)|k \in \mathrm{P}\}$ is a fuzzy pattern pair family for training. And for the input pattern $\mathbf{x}_k$ of (2.1), let the corresponding real output pattern be $\mathbf{o}_k = (o_1^k, ..., o_m^k) : \mathbf{o}_k = \mathbf{x}_k \circ W$, that is

$$o_j^k = \bigvee_{i \in \mathrm{N}} \{x_i^k \wedge w_{ij}\} \ (k \in \mathrm{P}, j \in \mathrm{M}).$$

Define the error function $E(W)$ as follows:

$$E(W) = \frac{1}{2}\sum_{k=1}^{p} \|\mathbf{o}_k - \mathbf{y}_k\|^2 = \frac{1}{2}\sum_{k=1}^{p}\sum_{j=1}^{m}\big(o_j^k - y_j^k\big)^2. \tag{2.32}$$

Since $E(W)$ is non-differentiable with respect to w_{ij}, we can not design the BP algorithm directly using $E(W)$. So we utilize the functions La and Sm to replace the fuzzy operators $\vee$ and $\wedge$, respectively. By Lemma 2.3, when s is sufficiently large, we have

$$E(W) \approx e(W) \triangleq \frac{1}{2}\sum_{k=1}^{p}\sum_{j=1}^{m}\big(La(s; Sm(s; x_1^k, w_{1j}), ..., Sm(s; x_n^k, w_{nj})) - y_j^k\big)^2. \tag{2.33}$$

$e(W)$ is a differentiable function, so we can employ the partial derivative $\partial e(W)/\partial w_{ij}$ to develop a BP algorithm of (2.1).

Theorem 2.12 *Give the fuzzy pattern pair family* $\{(\mathbf{x}_k,\ \mathbf{y}_k)|k \in \mathrm{P}\}$. *Then* $e(W)$ *is continuously differentiable with respect to* w_{ij} *for* $i \in \mathrm{N},\ j \in \mathrm{M}$. *And*

$$\frac{\partial e(W)}{\partial w_{ij}} = \sum_{k=1}^{p} \frac{-\exp\big(s \cdot \triangle(i,k)\big)\Gamma(s)}{\big(\sum\limits_{p=1}^{d} \exp\big(s \cdot \triangle(p,k)\big)\big)^2} \cdot \frac{1 + (sx_i^k - sw_{ij} + 1)\exp(-s(x_i^k - w_{ij}))}{(1 + \exp(-s(x_i^k - w_{ij})))^2}.$$

where $\Gamma(s) = \sum\limits_{p=1}^{d}\big\{s \cdot \exp\big(s \cdot \triangle(p,k)\big) - \exp\big(s \cdot \triangle(i,k)\big) - 1\big\}\exp\big(s \cdot \triangle(p,k)\big)$, and $\triangle(i,k) = Sm(s; x_i^k, w_{ij})$.

Proof. By Lemma 2.4, considering $\triangle(i,k) = Sm(s; x_i^k, w_{ij})$ for $i \in \mathrm{N},\ j \in \mathrm{M}$, we have

$$\frac{\partial La(s; Sm(s; x_1^k, w_{1j}), ..., Sm(s; x_n^k, w_{nj}))}{\partial Sm(s; x_i^k, w_{ij})} = \frac{-\exp\big(s \cdot \triangle(i,k)\big) \cdot \Gamma(s)}{\big(\sum\limits_{p=1}^{d} \exp\big(s \cdot \triangle(p,k)\big)\big)^2}.$$

And by (2.29) easily we can show

$$\frac{\partial Sm(s; x_i^k, w_{ij})}{\partial w_{ij}} = \frac{1 + (sx_i^k - sw_{ij} + 1)\exp(-s(x_i^k - w_{ij}))}{(1 + \exp(-s(x_i^k - w_{ij})))^2}.$$

By $\partial e(W)/\partial w_{ij} = \sum_{k=1}^{p}\big(\partial e(W)/\partial Sm(s;x_i^k,w_{ij})\big)\cdot\big(\partial Sm(s;x_i^k,w_{ij})/\partial w_{ij}\big)$, it follows that

$$\frac{\partial e(W)}{\partial w_{ij}}=\sum_{k=1}^{p}\frac{-\exp\big(s\cdot\triangle(i,k)\big)\Gamma(s)}{\big(\sum_{p=1}^{d}\exp\big(s\cdot\triangle(p,k)\big)\big)^2}\cdot\frac{1+(sx_i^k-sw_{ij}+1)\exp(-s(x_i^k-w_{ij}))}{(1+\exp(-s(x_i^k-w_{ij})))^2}.$$

Therefore, $e(W)$ is continuously differentiable with respect to w_{ij}. □

Using the partial derivatives in Theorem 2.12 we can design a BP algorithm for W of (2.1).

Algorithm 2.3 BP learning algorithm of FAM's.

Step 1. Initialization. Put $w_{ij}(0) = 0$, and let $W(0) = (w_{ij}(0))_{n\times m}$, set $t = 1$.

Step 2. Denote $W(t) = (w_{ij}(t))_{n\times m}$.

Step 3. Iteration scheme. $W(t)$ iterates with the following law:

$$\Omega = w_{ij}(t) - \delta\cdot\frac{\partial e(W(t))}{\partial w_{ij}(t)} + \alpha\cdot\triangle w_{ij}(t),\quad w_{ij}(t+1) = (\Omega\vee 0)\wedge 1.$$

Step 4. Stop condition. Discriminate $|e(W(t+1))| < \varepsilon$? If yes, output $w_{ij}(t+1)$; otherwise, let $t = t+1$ go to Step 2.

In the following we illustrate Algorithm 2.3 by a simulation to train FAM (2.1). To this end, Give a fuzzy pattern pair family as shown in Table 2.3.

Table 2.3 Fuzzy pattern pair family for training

No.	Input pattern	Desired output	Real pattern
1	(0.64, 0.50, 0.70, 0.60)	(0.64, 0.70)	(0.6400, 0.7000)
2	(0.40, 0.45, 0.80, 0.65)	(0.65, 0.80)	(0.6500, 0.7867)
3	(0.75, 0.70, 0.35, 0.25)	(0.75, 0.50)	(0.7250, 0.5325)
4	(0.33, 0.67, 0.35, 0.50)	(0.67, 0.50)	(0.6700, 0.5000)
5	(0.65, 0.70, 0.90, 0.75)	(0.75, 0.80)	(0.7500, 0.7867)
6	(0.95, 0.30, 0.45, 0.60)	(0.80, 0.60)	(0.7250, 0.6000)
7	(0.80, 1.00, 0.85, 0.70)	(0.80, 0.80)	(0.7864, 0.7867)
8	(0.10, 0.50, 0.70, 0.65)	(0.65, 0.70)	(0.6500, 0.7000)
9	(0.70, 0.70, 0.25, 0.56)	(0.70, 0.56)	(0.7000, 0.5600)

Choose $\alpha = 0.05$, $\eta = 0.3$. Let $s = 100$. With 1000 iterations, by Algorithm 2.3 we can establish the real output of (2.1), as shown Table 2.3. By comparison we know, Algorithm 2.3 possesses a quicker convergent speed and higher convergent accuracy.

The further subjects for FAM's include designing the learning algorithms related based on GA [5, 49], analysis on fault–tolerance of systems [22, 33–35, 37, 38], and applying the results obtained to many real fields, such as, signal processing [42], system modeling and identification [21, 44, 50], system control [31, 32] and so on. These researches are at their infancy, and so they have a great prospect for the future research.

§2.4 Fuzzy ART and fuzzy ARTMAP

Through the learning of a FAM, a given family of fuzzy patterns may be stored in the FAM, and the connection weight matrix W is established. If a new fuzzy pattern is presented to the FAM and asked to be stored in W, the FAM has to be trained to violate the original W. Thus, FAM's as competitive networks do not have stable learning in response to arbitrary input patterns. The learning instability occurs because of the network's adaptivity, which causes prior learning to be eroded by more recent learning. How can a system be receptive to significant new patterns and yet remain stable in response to irrelevant patterns? Adaptive resonance theory (ART) developed by Carpenter et al addresses such a dilemma [6]. As each input pattern is presented to ART, it is compared with the prototype vector that it most closely matches. If the match between the prototype and the input vector is no adequate, a prototype is selected. In this way previously learned memories are nor eroded by new learning. ART1 can process patterns expressed as vectors whose components are either 0 or 1. Fuzzy ART is a fuzzy version of ART1 [7], so let us now recall ART1 and its architecture.

2.4.1 ART1 architecture

An ART1 network consists of five parts, two subsystems which are called the attentional subsystem C (comparing layer) and the orienting subsystem R (recognition layer), respectively, and three controllers, two gain controllers G_1 and G_2, which generate the controlling signals G_1, G_2, respectively. and reset controller 'Reset'. The five components act together to form an efficient pattern classifying model. The ART1 has an architecture as shown in Figure 2.3.

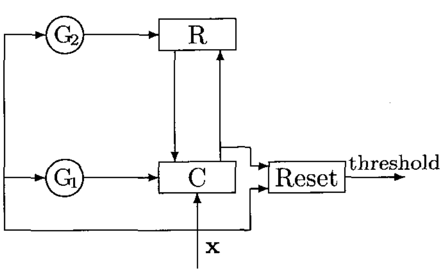


Figure 2.3 Architecture of ART1

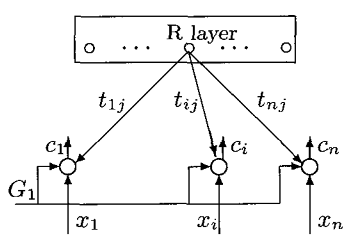


Figure 2.4 Attentional subsystem

Let us now describe the respective functions of five parts of ART1 in Figure 2.3. When an input $\mathbf{x} = (x_1, ..., x_n) \in \{0, 1\}^n$ is presented the network, the gain controller G_2 tests whether it is 0, and the corresponding controlling signal $G_2 = x_1 \vee \cdots \vee x_n$, that is, $\forall i \in \{1, ..., n\}$, $x_i = 0, \Longrightarrow G_2 = 0$, otherwise $G_2 = 1$. Suppose the output of the recognition layer R is $\mathbf{r} = (r_1, ..., r_m)$, and $R_0 = r_1 \vee \cdots \vee r_m$. Then the controlling signal G_1 of the gain controller G_1 is the product of G_2 and the complementary of R_0, that is, $G_1 = G_2 \cdot (1 - R_0) = G_2 \cdot R_0^c$. Therefore, $\mathbf{r} = \mathbf{0}$, $\mathbf{x} \neq \mathbf{0}, \Longrightarrow G_1 = 1$; otherwise $G_1 = 0$. The reset controller 'Reset' makes the wining neuron in competition in layer R lose efficacy.

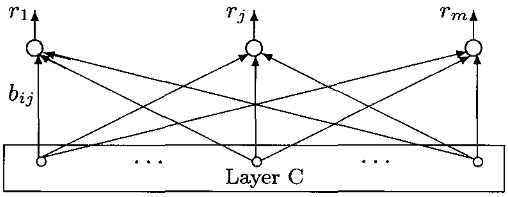


Figure 2.5 Orienting subsystem

There exist n units (nodes) in the comparing layer C, which are connected respectively with each node in the recognition layer, as shown in Figure 2.4. Each node in C accepts three signals, the input x_i, gain controlling signal G_1 and the feedback signal of the wining node in the recognition R, $t_{ij^*} \in \{0, 1\}$. Outputs c_i's of the nodes in C are determined by the 2/3 criterion, i.e. the majority criterion: The value of c_i is identical to the common one of the majority in G_i, t_{ij^*}, x_i.

There exist m nodes in the recognition layer R, each of which is connected with the nodes in C to form a feedforward competing network, as shown in Figure 2.5. m means the number of classified fuzzy patterns. By R the new fuzzy patterns can be added to the set of the patterns classified, dynamically. Suppose the connection weight between the i−th node in C and the j−th node in R is b_{ij}. And the output vector of C is $\mathbf{c} = (c_1, ..., c_n)$, which propagates forwardly to R, and the nodes in R compete to generate a wining node j^*. All the components of the output vector $\mathbf{r} = (r_1, ..., r_m)$ of R are zero but $r_{j^*} = 1$. In the following we will explain how an ART1 works.

First step—matching. When there is no input signal the network is in a waiting state, and let $\mathbf{x} = \mathbf{0}$. Thus $G_2 = R_0 = 0$, and there is no competition in the recognition layer R, and consequently $t_{ij} = 0$; When the signal $\mathbf{x} \neq \mathbf{0}$ is presented to the network, $G_2 = 1$, $R_0 = 0, \Longrightarrow G_1 = G_2 \cdot R_0^c = 1$, and so by the 2/3 criterion, $\mathbf{c} = \mathbf{x}$. And we get the input of the j−th node in R as follows:

$$P_j = \sum_{i=1}^{n} b_{ij} \cdot x_i \; (j = 1, ..., m),$$

We call P_j the matching degree between $\mathbf{x}$ and $\mathbf{b}_j = (b_{1j}, ..., b_{nj})$. Choose such a node j^*, whose matching degree is maximum, that is, $P_{j^*} = \bigvee_{1 \le j \le m} \{P_j\}$. The node is called a wining node, and so $r_{j^*} = 1$, $r_j = 0\,(j \neq j^*)$.

Second step—comparing. The output vector $\mathbf{r} = (r_1, ..., r_m)$ of R return to C through the connection weight matrix $T = (t_{ij})_{n\times m}$. The fact that $r_{j^*} = 1$ results in the weight vector $\mathbf{t}_{j^*} = (t_{1j^*}, ..., t_{nj^*})$ being active, and others being inactive. Thus, $R_0 = 1, \Longrightarrow G_1 = G_2 \cdot R_0^c = 0$. The output $\mathbf{c} = (c_1, ..., c_n)$ of C characterizes the matching degree M_0 between $\mathbf{t}_{j^*}$ and the input $\mathbf{x}$:

$$M_0 = \langle \mathbf{x}, \mathbf{t}_{j^*}^{\mathrm{T}} \rangle = \sum_{i=1}^{n} t_{ij^*} \cdot x_i = \sum_{i=1}^{n} c_i.$$

Since $x_i \in \{0, 1\}$, M_0 is the number of overlapping nonzero components between $\mathbf{t}_{j^*}$ and $\mathbf{x}$. Suppose there exist M_1 nonzero components in $\mathbf{x}$, i.e. $M_1 = x_1 + \cdots + x_n$. M_0/M_1 also reflect the similarity between $\mathbf{x}$ and $\mathbf{t}_{j^*}$. Give $\rho \in [0, 1]$ as a minimum similarity vigilance of the input pattern $\mathbf{x}$ and the template $\mathbf{t}_j$ corresponding to a wining node. If $M_0/M_1 < \rho$, then $\mathbf{x}$ and $\mathbf{t}_j$ can not satisfy the similarity condition, and through the reset signal 'Reset' let the match finished in the first step lose its efficacy, and the wining node become invalid. Go to third step, searching; If $M_0/M_1 > \rho$, then $\mathbf{x}$ and $\mathbf{t}_j$ are close enough, and the 'resonance' between $\mathbf{x}$ and $\mathbf{t}_j$ takes place, the match in the first step is effective. Go to fourth step, learning.

Third step—searching. Reset signal makes the wining node established by the first step keep restrained, and the restraining state is kept until the ART1 network receives a new pattern. And we have, $R_0 = 0$, $G_1 = 1$. The network returns the matching state in the first step, and go to first step. If the circulating procedure does not stop until all patterns in R are used, then $(m + 1)-$ th node have to be added to store the current pattern as a new template, and let $t_{i(m+1)} = 1$, $b_{i(m+1)} = x_i\,(i = 1, ..., n)$.

Fourth step—learning. b_{ij} and t_{ij} iterate according to the following Algorithm 2.4, so that the stronger 'resonance' between $\mathbf{x}$ and $\mathbf{t}_{j^*}$ takes places.

Algorithm 2.4 The connection weight matrices $B = (b_{ij})_{n\times m}$, $T = (t_{ij})_{n\times m}$ iterate with the following steps:

Step 1. Initialization: let $t = 0$, and choose the initial values of b_{ij} and t_{ij} :

$$b_{ij}(0) = \frac{1}{n+1};\ \ t_{ij}(0) = 1\ (i = 1, ..., n;\ j = 1, ..., m).$$

Step 2. Receive an input: Give the input pattern $\mathbf{x} = (x_1, ..., x_n) \in \{0, 1\}^n$.

Step 3. Determine the wining node j^* : Calculate the matching degree P_j, and compute $j^* : P_{j^*} = \bigvee_{j=1}^{m} \{P_j\}$.

Step 4. Compute the similarity degree. By $M_0 = \sum_{i=1}^{n} x_i \cdot t_{ij^*} = \sum_{i=1}^{n} c_i$ we may establish the similarity degree between the vector $\mathbf{t}_{j^*} = (t_{1j^*}, ..., t_{nj^*})$, corresponding to the wining node and the input pattern $\mathbf{x}$.

Step 5. Vigilance test. Calculate $M_1 = \sum_{i=1}^{n} x_i$. If $M_0/M_1 < \rho$, let the wining node j^* invalid, and put $r_{j^*} = 0$. Go to Step 6; if $M_0/M_1 > \rho$, we classify $\mathbf{x}$ into the pattern class that includes $\mathbf{t}_{j^*}$, and go to Step 7.

Step 6. Search pattern class. If the wining node number is less than m, go to Step 3; If the invalid node number equals to m, then in R add the $(m+1)-$th node, and let $b_{i(m+1)} = x_i$, $t_{i(m+1)} = 1\,(i = 1, ..., n)$. Put $m = m + 1$, go to Step 2.

Step 7. Adjust connection weights. For $i = 1, ..., n$, t_{ij^*} and b_{ij^*} are adjusted with the following scheme:

$$\begin{cases} t_{ij^*}(t+1) = t_{ij^*} \cdot x_i, \\ b_{ij^*}(t+1) = \dfrac{t_{ij^*}(t) \cdot x_i}{0.5 + \sum\limits_{i'=1}^{n} t_{i'j^*}(t) \cdot x_{i'}} = \dfrac{t_{ij^*}(t+1)}{0.5 + \sum\limits_{i'=1}^{n} t_{i'j^*}(t+1)}. \end{cases}$$

Set $t = t + 1$, go to Step 2.

Algorithm 2.4 for the ART1 network is a on-line learning. In the ART1 network, the nodes stand for the classified pattern classes respectively, each of which includes some similar patterns. By the vigilance ρ we can establish the number of the patterns classified. The larger ρ is, the more the classified patterns are.

2.4.2 Fuzzy ART

Like an ART1 network, a fuzzy ART consist also of two subsystems [7, 13, 14], one is the attentional subsystem, and another is the orienting subsystem, as shown in Figure 2.6. The attentional subsystem is a two-layer network architecture, in which $F_1^{\mathbf{x}}$ is the input layer, accepting the input fuzzy patterns, and $F_2^{\mathbf{x}}$ is a pattern expressing layer. All fuzzy patterns in $F_2^{\mathbf{x}}$ constitute a classification of the input patterns. Orienting subsystem consists of a reset node 'Reset', which accepts all information coming from $F_1^{\mathbf{x}}$ layer, $F_2^{\mathbf{x}}$ layer and $F_0^{\mathbf{x}}$ layer using for transforming the input patterns.

By $F_0^{\mathbf{x}}$ layer we can also complete the complement code of the input fuzzy pattern $\mathbf{x}$, that is, the output $\mathbf{I}$ of $F_0^{\mathbf{x}}$ is determined as follows:

$$\mathbf{I} = (\mathbf{x}, \mathbf{x}^c) = (x_1, ..., x_n, x_1^c, ..., x_n^c) = (x_1, ..., x_n, 1 - x_1, ..., 1 - x_n),$$

where $x_i \in [0, 1]$ $(i = 1, ..., n)$. From now on, we take the pattern $\mathbf{I}$ as an input of a fuzzy ART. Thus, $F_1^{\mathbf{x}}$ includes $2n$ nodes, and suppose $F_2^{\mathbf{x}}$ includes m nodes. Let the connection weight between the node i in $F_1^{\mathbf{x}}$ and the node j in $F_2^{\mathbf{x}}$ be $W_{ij}^{\mathbf{x}}$, and the connection weight between the node j in $F_2^{\mathbf{x}}$ and the node i in $F_1^{\mathbf{x}}$ be $w_{ji}^{\mathbf{x}}$. Denote

$$\mathbf{W}_j^{\mathbf{x}} = \left(W_{1j}^{\mathbf{x}}, ..., W_{(2n)j}^{\mathbf{x}}\right),\ \ \mathbf{w}_j^{\mathbf{x}} = \left(w_{j1}^{\mathbf{x}}, ..., w_{j(2n)}^{\mathbf{x}}\right),$$

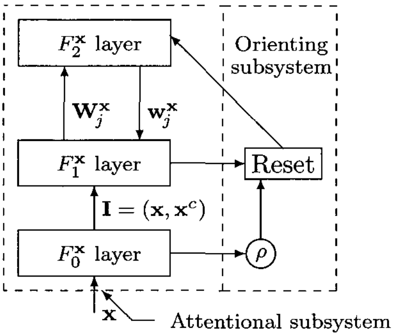


Figure 2.6 Fuzzy ART architecture

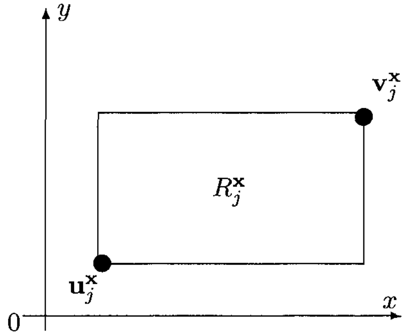


Figure 2.7 Geometry form of pattern $\mathbf{w}_j^{\mathbf{x}}$

We call $\mathbf{w}_j^{\mathbf{x}}$ a template, where $j = 1, ..., m$. Suppose the initial values of the connection weights $W_{ij}^{\mathbf{x}}$ and $w_{ji}^{\mathbf{x}}$ are $W_{ij}^{\mathbf{x}}(0)$, $w_{ji}^{\mathbf{x}}(0)$, respectively:

$$W_{ij}^{\mathbf{x}}(0) = \frac{1}{\alpha_{\mathbf{x}} + M_{\mathbf{x}}}, \quad w_{ji}^{\mathbf{x}}(0) = 1 \ (i = 1, ..., 2n; \ j = 1, ..., m),$$

where $\alpha_{\mathbf{x}}$, $M_{\mathbf{x}}$ are parameters of the fuzzy ART, $\alpha_{\mathbf{x}} \in (0, +\infty)$ is a selection parameter, and $M_{\mathbf{x}} \in [2n, +\infty)$ is a uncommitted node parameter [14]. Denote $\mathbf{W}_j^{\mathbf{x}}(0) = \big(W_{1j}^{\mathbf{x}}(0), ..., W_{(2n)j}^{\mathbf{x}}(0)\big)$, $\mathbf{w}^{\mathbf{x}}(0) = \big(w_{j1}^{\mathbf{x}}(0), ..., w_{j(2n)}^{\mathbf{x}}(0)\big)$. Then $\mathbf{W}_j^{\mathbf{x}}(0)$, $\mathbf{w}_j^{\mathbf{x}}(0)$ correspond, respectively to the $j-$th connection weight vectors before the input fuzzy pattern $\mathbf{x}$ is expressed in $F_2^{\mathbf{x}}$.

Before discussing the I/O relationship of the fuzzy ART, we introduce some notations. Suppose $\mathbf{y}_q = (y_1^q, ..., y_{2n}^q) \in [0,1]^{2n}\,(q = 1, 2)$, denote $|\mathbf{y}^1| = \sum_{i=1}^{2n} y_i^1$, and we call $\text{dis}(\mathbf{y}_1, \mathbf{y}_2) = \sum_{i=1}^{2n} |y_i^1 - y_i^2|$ the metric between the fuzzy patterns $\mathbf{y}_1$ and $\mathbf{y}_2$. Denote

$$\mathbf{y}_1 \vee \mathbf{y}_2 = (y_1^1 \vee y_1^2, ..., y_{2n}^1 \vee y_{2n}^2); \ \ \mathbf{y}_1 \wedge \mathbf{y}_2 = (y_1^1 \wedge y_1^2, ..., y_{2n}^1 \wedge y_{2n}^2).$$

We call the node j a uncommitted node in $F_2^{\mathbf{x}}$, if $\mathbf{w}_j^{\mathbf{x}} = \mathbf{w}_j^{\mathbf{x}}(0) = (1, ..., 1)$, otherwise the node j is called a committed node. For $j \in \{1, ..., m\}$, define

$$\begin{cases} \mathbf{W}_j^{\mathbf{x},\text{old}} = \mathbf{W}_j^{\mathbf{x}}, & \mathbf{I} \text{ is not classified into } F_2^{\mathbf{x}}, \\ \mathbf{W}_j^{\mathbf{x},\text{new}} = \mathbf{W}_j^{\mathbf{x}}, & \mathbf{I} \text{ is classified into } F_2^{\mathbf{x}}; \end{cases}$$

$$\begin{cases} \mathbf{w}_j^{\mathbf{x},\text{old}} = \mathbf{w}_j^{\mathbf{x}}, & \mathbf{I} \text{ is not classified into } F_2^{\mathbf{x}}, \\ \mathbf{w}_j^{\mathbf{x},\text{new}} = \mathbf{w}_j^{\mathbf{x}}, & \mathbf{I} \text{ is classified into } F_2^{\mathbf{x}}. \end{cases}$$

For the input $\mathbf{I}$ of $F_1^{\mathbf{x}}$, by the upward connection weight matrix $\left(\mathbf{W}_1^{\mathbf{x}}, ..., \mathbf{W}_{2n}^{\mathbf{x}}\right)^{\mathrm{T}}$ we can establish an input of $F_2^{\mathbf{x}}$ layer as $\mathbf{t}(\mathbf{I}) = (t_1(\mathbf{I}), ..., t_m(\mathbf{I}))$:

$$t_j(\mathbf{I}) = \begin{cases} \dfrac{|\mathbf{I}|}{\alpha_{\mathbf{x}} + M_{\mathbf{x}}}, & \text{node } j \text{ is an uncommitted node,} \\ \dfrac{|\mathbf{I} \wedge \mathbf{w}_j^{\mathbf{x},\mathrm{old}}|}{\alpha_{\mathbf{x}} + |\mathbf{w}_j^{\mathbf{x},\mathrm{old}}|}, & \text{node } j \text{ is a committed node,} \end{cases} \tag{2.34}$$

where $j \in \{1, ..., m\}$. Let $j^* \in \{1, ..., m\} : t_{j^*}(\mathbf{I}) = \bigvee_{j=1}^{m} \{t_j(\mathbf{I})\}$. And in $F_2^{\mathbf{x}}$ only the node j^* is active, and others are inactive. $\mathbf{w}_{j^*}^{\mathbf{x}}$ may be taken as a candidacy of a standard pattern class, into which the input pattern $\mathbf{I}$ will be classified. The matching degree between $\mathbf{w}_{j^*}^{\mathbf{x}}$ and $\mathbf{I}$ is $\lambda_{\mathbf{x}} = \left(|\mathbf{I} \wedge \mathbf{w}_{j^*}^{\mathbf{x},\mathrm{old}}|\right)/|\mathbf{I}|$. For a given vigilance $\rho \in [0, 1]$, if $\lambda_{\mathbf{x}} > \rho$, then $\mathbf{I}$ is classified into $\mathbf{w}_{j^*}^{\mathbf{x}}$. Similarly with Algorithm 2.4, the connection weight vectors can be trained as follows:

$$\begin{cases} \mathbf{w}_{j^*}^{\mathbf{x},\mathrm{new}} = \mathbf{I} \wedge \mathbf{w}_{j^*}^{\mathbf{x},\mathrm{old}}; \\ \mathbf{W}_{j^*}^{\mathbf{x},\mathrm{new}} = \dfrac{\mathbf{I} \wedge \mathbf{w}_{j^*}^{\mathbf{x},\mathrm{old}}}{\alpha_{\mathbf{x}} + |\mathbf{I} \wedge \mathbf{w}_{j^*}^{\mathbf{x},\mathrm{old}}|} = \dfrac{\mathbf{w}_{j^*}^{\mathbf{x},\mathrm{new}}}{\alpha_{\mathbf{x}} + |\mathbf{w}_{j^*}^{\mathbf{x},\mathrm{new}}|}. \end{cases}$$

If $\lambda_{\mathbf{x}} < \rho$, then the reset node 'Reset' in the orienting subsystem generates a signal to enable the node j^* inhibitory. And we choose a second-maximum $t_j(\mathbf{I})$ as the new active node j^*. Repeat this procedure. If one by one each $j \in \{1, ..., m\}$ is taken as j^*, and it can not ensure $\lambda_{\mathbf{x}} > \rho$, then $\mathbf{I}$ is stored in $F_2^{\mathbf{x}}$ as a representative of a new fuzzy pattern class, and in $F_2^{\mathbf{x}}$ we add a new node according to $\mathbf{I}$.

Next let us show the geometry sense of fuzzy pattern in the fuzzy ART [14]. For $j \in$ M, the weight vector $\mathbf{w}_j^{\mathbf{x}}$ can be expressed by two n dimensional vectors $\mathbf{u}_j^{\mathbf{x}}$ and $\mathbf{v}_j^{\mathbf{x}}$:

$$\mathbf{w}_j^{\mathbf{x}} = \left(\mathbf{u}_j^{\mathbf{x}}, \{\mathbf{v}_j^{\mathbf{x}}\}^c\right).$$

where $\mathbf{u}_j^{\mathbf{x}} \leq \mathbf{v}_j^{\mathbf{x}}$. If let $n = 2$, then $\mathbf{w}_j^{\mathbf{x}}$ can be established by two vertices $\mathbf{u}_j^{\mathbf{x}}$ $\mathbf{v}_j^{\mathbf{x}}$ of the rectangle $R_j^{\mathbf{x}}$, as shown in Figure 2.7. Give an input pattern $\mathbf{I}^{\mathbf{x}} = (\mathbf{x}, \mathbf{x}^c)$, at first it is not classified by the fuzzy ART. The rectangle $R_j^{\mathbf{x},\mathrm{old}}$ is a geometry representation of the template vector $\mathbf{w}_j^{\mathbf{x},\mathrm{old}}$. If $\mathbf{I}^{\mathbf{x}} \in R_j^{\mathbf{x},\mathrm{old}}$, then $\mathbf{u}_j^{\mathbf{x},\mathrm{old}} \leq \mathbf{x} \leq \mathbf{v}_j^{\mathbf{x},\mathrm{old}}$, and the the template vector keeps unchanged, that is

$$\mathbf{w}_j^{\mathbf{x},\mathrm{new}} = \mathbf{w}_j^{\mathbf{x},\mathrm{old}} \wedge \mathbf{I}^{\mathbf{x}} = \left(\mathbf{u}_j^{\mathbf{x},\mathrm{old}} \wedge \mathbf{x}, \{\mathbf{v}_j^{\mathbf{x},\mathrm{old}} \vee \mathbf{x}\}^c\right) = \mathbf{w}_j^{\mathbf{x},\mathrm{old}}.$$

If $\mathbf{I}^{\mathbf{x}} \notin R_j^{\mathbf{x},\mathrm{old}}$, then easily we can show, $\mathbf{w}_j^{\mathbf{x},\mathrm{new}} \neq \mathbf{w}_j^{\mathbf{x},\mathrm{old}}$. And the weight vectors change, and the number of the rectangles increases, correspondingly. Its maximum value is determined by the vigilance ρ. For the input fuzzy pattern $\mathbf{I}^{\mathbf{x}}$, if $|\mathbf{I}^{\mathbf{x}}| = M$, and

$$\left|\mathbf{I}^{\mathbf{x}} \wedge \mathbf{w}_j^{\mathbf{x},\mathrm{old}}\right| \geq M \cdot \rho, \tag{2.35}$$

then in $F_2^{\mathbf{x}}$ layer, $\mathbf{I}^{\mathbf{x}}$ can be classified into a pattern class that includes the weight vector $\mathbf{w}_j^{\mathbf{x},\text{old}}$. By computation we can see

$$\begin{aligned} \left|\mathbf{I}^{\mathbf{x}} \wedge \mathbf{w}_j^{\mathbf{x},\text{old}}\right| &= \left|\left(\mathbf{u}_j^{\mathbf{x},\text{old}} \wedge \mathbf{x},\ \{\mathbf{v}_j^{\mathbf{x},\text{old}} \vee \mathbf{x}\}^c\right)\right| \\ &= \sum_{i=1}^{n}\left(x_i \wedge u_{ij}^{\mathbf{x},\text{old}}\right) + \sum_{i=1}^{n}\left(a_i \vee v_{ij}^{\mathbf{x},\text{old}}\right)^c \\ &= n - \left|\left(\mathbf{x} \vee \mathbf{v}_j^{\mathbf{x},\text{old}}\right) - \left(\mathbf{x} \wedge \mathbf{u}_j^{\mathbf{x},\text{old}}\right)\right| = n - \left|R_j^{\mathbf{x},\text{new}}\right|. \end{aligned} \tag{2.36}$$

So by (2.35) (2.36) it follows that $\left|R_j^{\mathbf{x},\text{new}}\right| \leq n(1-\rho)$. Therefore, if ρ is very small, i.e. $\rho \approx 0$, then the input space is filled with small rectangles; If $\rho \approx 1$, then there are only a few of rectangles.

Now we present the classifying order of the input pattern $\mathbf{I} = (\mathbf{x},\ \mathbf{x}^c)$ by the fuzzy ART when choosing the parameter $\alpha_{\mathbf{x}}$ as 'very small' 'medium' and 'very large', respectively. To this end we at first give three lemmas.

Lemma 2.5 *Let* $\mathbf{I} = (\mathbf{x}, \mathbf{x}^c)$ *be an input fuzzy pattern of the fuzzy ART, and* $\mathbf{I} \in R_{j_1}^{\mathbf{x},\text{old}} \cap R_{j_2}^{\mathbf{x},\text{old}}$, $\left|R_{j_1}^{\mathbf{x},\text{old}}\right| < \left|R_{j_2}^{\mathbf{x},\text{old}}\right|$. *Then the rectangle* $R_{j_1}^{\mathbf{x},\text{old}}$ *is chosen precede* $R_{j_2}^{\mathbf{x},\text{old}}$, *that is,* $t_{j_1}(\mathbf{I}) > t_{j_2}(\mathbf{I})$.

Proof. By the assumption we have, j_1, j_2 are committed nodes in $F_2^{\mathbf{x}}$ layer. So by (2.34) it follows that

$$\left\{\begin{aligned} t_{j_1}(\mathbf{I}) &= \frac{\left|\mathbf{I} \wedge \mathbf{w}_{j_1}^{\mathbf{x},\text{old}}\right|}{\alpha_{\mathbf{x}} + \left|\mathbf{w}_{j_1}^{\mathbf{x},\text{old}}\right|} = \frac{\left|\mathbf{w}_{j_1}^{\mathbf{x},\text{old}}\right|}{\alpha_{\mathbf{x}} + \left|\mathbf{w}_{j_1}^{\mathbf{x},\text{old}}\right|} = \frac{n - \left|R_{j_1}^{\mathbf{x},\text{old}}\right|}{\alpha_{\mathbf{x}} + n - \left|R_{j_1}^{\mathbf{x},\text{old}}\right|}; \\ t_{j_2}(\mathbf{I}) &= \frac{\left|\mathbf{I} \wedge \mathbf{w}_{j_2}^{\mathbf{x},\text{old}}\right|}{\alpha_{\mathbf{x}} + \left|\mathbf{w}_{j_1}^{\mathbf{x},\text{old}}\right|} = \frac{\left|\mathbf{w}_{j_2}^{\mathbf{x},\text{old}}\right|}{\alpha_{\mathbf{x}} + \left|\mathbf{w}_{j_2}^{\mathbf{x},\text{old}}\right|} = \frac{n - \left|R_{j_2}^{\mathbf{x},\text{old}}\right|}{\alpha_{\mathbf{x}} + n - \left|R_{j_2}^{\mathbf{x},\text{old}}\right|}. \end{aligned}\right. \tag{2.37}$$

Using the assumption we get, $n - \left|R_{j_1}^{\mathbf{x},\text{old}}\right| > n - \left|R_{j_2}^{\mathbf{x},\text{old}}\right|$. Hence (2.37) implies that $t_{j_1}(\mathbf{I}) > t_{j_2}(\mathbf{I})$. Thus, by the classifying rule of the fuzzy ART, the rectangle $R_{j_1}^{\mathbf{x},\text{old}}$ is chosen precede $R_{j_2}^{\mathbf{x},\text{old}}$. □

Lemma 2.6 *Suppose an input fuzzy pattern* $\mathbf{I} = (\mathbf{x},\ \mathbf{x}^c)$ *is presented to the fuzzy ART, and* $\mathbf{I} \in R_{j_2}^{\mathbf{x},\text{old}} \setminus R_{j_1}^{\mathbf{x},\text{old}}$. *Then the rectangle* $R_{j_1}^{\mathbf{x},\text{old}}$ *is chosen precede* $R_{j_2}^{\mathbf{x},\text{old}}$ *if and only if*

$$\text{dis}\left(\mathbf{I},\ R_{j_1}^{\mathbf{x},\text{old}}\right) < \left(n + \alpha_{\mathbf{x}} - \left|R_{j_1}^{\mathbf{x},\text{old}}\right|\right)\left\{\frac{n - \left|R_{j_1}^{\mathbf{x},\text{old}}\right|}{\alpha_{\mathbf{x}} + n - \left|R_{j_1}^{\mathbf{x},\text{old}}\right|} - \frac{n - \left|R_{j_2}^{\mathbf{x},\text{old}}\right|}{\alpha_{\mathbf{x}} + n - \left|R_{j_2}^{\mathbf{x},\text{old}}\right|}\right\}.$$

Proof. By the assumption and (2.34) easily we can show

$$\left\{\begin{aligned} t_{j_1}(\mathbf{I}) &= \frac{\left|\mathbf{I} \wedge \mathbf{w}_{j_1}^{\mathbf{x},\text{old}}\right|}{\alpha_{\mathbf{x}} + \left|\mathbf{w}_{j_1}^{\mathbf{x},\text{old}}\right|} = \frac{\left|\mathbf{w}_{j_1}^{\mathbf{x},\text{new}}\right|}{\alpha_{\mathbf{x}} + \left|\mathbf{w}_{j_1}^{\mathbf{x},\text{old}}\right|} = \frac{n - \left|R_{j_1}^{\mathbf{x},\text{new}}\right|}{\alpha_{\mathbf{x}} + n - \left|R_{j_1}^{\mathbf{x},\text{old}}\right|}; \\ t_{j_2}(\mathbf{I}) &= \frac{\left|\mathbf{I} \wedge \mathbf{w}_{j_2}^{\mathbf{x},\text{old}}\right|}{\alpha_{\mathbf{x}} + \left|\mathbf{w}_{j_1}^{\mathbf{x},\text{old}}\right|} = \frac{\left|\mathbf{w}_{j_2}^{\mathbf{x},\text{old}}\right|}{\alpha_{\mathbf{x}} + \left|\mathbf{w}_{j_2}^{\mathbf{x},\text{old}}\right|} = \frac{n - \left|R_{j_2}^{\mathbf{x},\text{old}}\right|}{\alpha_{\mathbf{x}} + n - \left|R_{j_2}^{\mathbf{x},\text{old}}\right|}. \end{aligned}\right.$$

Then

$$t_{j_1}(\mathbf{I}) > t_{j_2}(\mathbf{I}) \Longleftrightarrow \frac{n - |R_{j_1}^{\mathbf{x},\text{new}}|}{\alpha_{\mathbf{x}} + n - |R_{j_1}^{\mathbf{x},\text{old}}|} > \frac{n - |R_{j_2}^{\mathbf{x},\text{old}}|}{\alpha_{\mathbf{x}} + n - |R_{j_2}^{\mathbf{x},\text{old}}|}. \tag{2.38}$$

But $|R_{j_1}^{\mathbf{x},\text{new}}| = |R_{j_1}^{\mathbf{x},\text{old}}| + \text{dis}(\mathbf{I},\ R_{j_1}^{\mathbf{x},\text{old}})$, which is replace into (2.38) we get, $t_{j_1}(\mathbf{I}) > t_{j_2}(\mathbf{I})$ if and only if the following fact holds:

$$\text{dis}(\mathbf{I},\ R_{j_1}^{\mathbf{x},\text{old}}) < \left(n + \alpha_{\mathbf{x}} - |R_{j_1}^{\mathbf{x},\text{old}}|\right)\left\{\frac{n - |R_{j_1}^{\mathbf{x},\text{old}}|}{\alpha_{\mathbf{x}} + n - |R_{j_1}^{\mathbf{x},\text{old}}|} - \frac{n - |R_{j_2}^{\mathbf{x},\text{old}}|}{\alpha_{\mathbf{x}} + n - |R_{j_2}^{\mathbf{x},\text{old}}|}\right\},$$

which implies the lemma. □

Lemma 2.7 *Suppose* $\mathbf{I} = (\mathbf{x}, \mathbf{x}^c)$ *is an input pattern presented to the fuzzy ART, and* $\mathbf{I} \in R_{j_1}^{\mathbf{x},\text{old}} \cap R_{j_2}^{\mathbf{x},\text{old}}$. *Then the rectangle* $R_{j_1}^{\mathbf{x},\text{old}}$ *is chosen precede* $R_{j_2}^{\mathbf{x},\text{old}}$ *if and only if*

$$\text{dis}(\mathbf{I},\ R_{j_1}^{\mathbf{x},\text{old}}) < \text{dis}(\mathbf{I}, R_{j_2}^{\mathbf{x},\text{old}}) \cdot \frac{n + \alpha_{\mathbf{x}} - |R_{j_1}^{\mathbf{x},\text{old}}|}{\alpha_{\mathbf{x}} + n - |R_{j_2}^{\mathbf{x},\text{old}}|} + \frac{\alpha_{\mathbf{x}}(|R_{j_2}^{\mathbf{x},\text{old}}| - |R_{j_1}^{\mathbf{x},\text{old}}|)}{\alpha_{\mathbf{x}} + n - |R_{j_2}^{\mathbf{x},\text{old}}|}. \tag{2.39}$$

Proof. Using the assumption and (2.34) we can conclude that

$$t_{j_1}(\mathbf{I}) = \frac{|\mathbf{I} \wedge \mathbf{w}_{j_1}^{\mathbf{x},\text{old}}|}{\alpha_{\mathbf{x}} + |\mathbf{w}_{j_1}^{\mathbf{x},\text{old}}|} = \frac{|\mathbf{w}_{j_1}^{\mathbf{x},\text{new}}|}{\alpha_{\mathbf{x}} + |\mathbf{w}_{j_1}^{\mathbf{x},\text{old}}|} = \frac{n - |R_{j_1}^{\mathbf{x},\text{new}}|}{\alpha_{\mathbf{x}} + n - |R_{j_1}^{\mathbf{x},\text{old}}|}.$$

$$t_{j_1}(\mathbf{I}) = \frac{|\mathbf{I} \wedge \mathbf{w}_{j_2}^{\mathbf{x},\text{old}}|}{\alpha_{\mathbf{x}} + |\mathbf{w}_{j_2}^{\mathbf{x},\text{old}}|} = \frac{|\mathbf{w}_{j_2}^{\mathbf{x},\text{new}}|}{\alpha_{\mathbf{x}} + |\mathbf{w}_{j_2}^{\mathbf{x},\text{old}}|} = \frac{n - |R_{j_2}^{\mathbf{x},\text{new}}|}{\alpha_{\mathbf{x}} + n - |R_{j_2}^{\mathbf{x},\text{old}}|}.$$

Therefore the following fact holds:

$$t_{j_1}(\mathbf{I}) > t_{j_2}(\mathbf{I}) \Longleftrightarrow \frac{n - |R_{j_1}^{\mathbf{x},\text{new}}|}{\alpha_{\mathbf{x}} + n - |R_{j_1}^{\mathbf{x},\text{old}}|} > \frac{n - |R_{j_2}^{\mathbf{x},\text{new}}|}{\alpha_{\mathbf{x}} + n - |R_{j_2}^{\mathbf{x},\text{old}}|}. \tag{2.40}$$

But for $k = 1, 2$, $|R_{j_k}^{\mathbf{x},\text{new}}| = |R_{j_k}^{\mathbf{x},\text{old}}| + \text{dis}(\mathbf{I},\ R_{j_k}^{\mathbf{x},\text{old}})$. Consequently by (2.40) it follows that $t_{j_1}(\mathbf{I}) > t_{j_2}(\mathbf{I})$ if and only if (2.39) is true. □

In order to utilize the parameter $\alpha_{\mathbf{x}}$ to establish the classifying order of the input fuzzy pattern $\mathbf{I}$ by the fuzzy ART, we define the function

$$\begin{cases} \phi_0(x) = \left(n + x - |R_{j_1}^{\mathbf{x},\text{old}}|\right)\left\{\dfrac{n - |R_{j_1}^{\mathbf{x},\text{old}}|}{x + n - |R_{j_1}^{\mathbf{x},\text{old}}|} - \dfrac{n - |R_{j_2}^{\mathbf{x},\text{old}}|}{x + n - |R_{j_2}^{\mathbf{x},\text{old}}|}\right\}; \\[2ex] \phi_1(x) = \text{dis}(\mathbf{I}, R_{j_2}^{\mathbf{x},\text{old}}) \cdot \dfrac{n + x - |R_{j_1}^{\mathbf{x},\text{old}}|}{x + n - |R_{j_2}^{\mathbf{x},\text{old}}|} + \dfrac{x(|R_{j_2}^{\mathbf{x},\text{old}}| - |R_{j_1}^{\mathbf{x},\text{old}}|)}{x + n - |R_{j_2}^{\mathbf{x},\text{old}}|}. \end{cases}$$

Theorem 2.13 *Suppose an input pattern* $\mathbf{I} = (\mathbf{x}, \mathbf{x}^c)$ *is presented to the fuzzy ART, and the parameter* $\alpha_{\mathbf{x}} \approx 0$. *Then*

(i) If $\mathbf{I} \in R_{j_1}^{\mathbf{x},\text{old}} \cap R_{j_2}^{\mathbf{x},\text{old}}$, *then* $\mathbf{I}$ *chooses first the rectangle between* $|R_{j_1}^{\mathbf{x},\text{old}}|$ *and* $|R_{j_2}^{\mathbf{x},\text{old}}|$ *with smaller size;*

(ii) If $\mathbf{I} \in R_{j_2}^{\mathbf{x},\text{old}} \setminus R_{j_1}^{\mathbf{x},\text{old}}$, *then* $\mathbf{I}$ *chooses first* $R_{j_1}^{\mathbf{x},\text{old}}$;

(iii) If $\mathbf{I} \notin R_{j_1}^{\mathbf{x},\text{old}} \cup R_{j_2}^{\mathbf{x},\text{old}}$, *then the pattern* $\mathbf{I}$ *chooses first* $R_{j_1}^{\mathbf{x},\text{old}}$ *if and only if*

$$\text{dis}(\mathbf{I}, R_{j_1}^{\mathbf{x},\text{old}})(n - |R_{j_2}^{\mathbf{x},\text{old}}|) < \text{dis}(\mathbf{I}, R_{j_2}^{\mathbf{x},\text{old}})(n - |R_{j_1}^{\mathbf{x},\text{old}}|).$$

Proof. (i) is a direct corollary of Lemma 2.5, so it suffices to show (ii) and (iii).

(ii) Since $\lim\limits_{x\to 0} \phi_0(x) = 0$, when $\alpha_{\mathbf{x}} \approx 0$, $\phi_0(\alpha_x) \approx 0$. Therefore, $\phi_0(\alpha_{\mathbf{x}}) \leq \text{dis}(\text{M}, \mathbf{I}, R_{j_1}^{\mathbf{x},\text{old}})$. Then by Lemma 2.6 it follows that $\mathbf{I}$ chooses the rectangle $R_{j_2}^{\mathbf{x},\text{old}}$ firstly.

(iii) Obviously, the following fact holds:

$$\lim_{x\to 0} \phi_1(x) = \phi_1(0) = \frac{\left(\text{dis}(\mathbf{I}, R_{j_2}^{\mathbf{x},\text{old}})(n - |R_{j_1}^{\mathbf{x},\text{old}}|)\right)}{n - |R_{j_2}^{\mathbf{x},\text{old}}|}.$$

And when $\alpha_{\mathbf{x}} \approx 0$, we have, $\text{dis}(\mathbf{I}, R_{j_1}^{\mathbf{x},\text{old}}) < \phi_1(\alpha_{\mathbf{x}}) \iff \text{dis}(\mathbf{I}, R_{j_1}^{\mathbf{x},\text{old}}) < \phi_1(0)$. So by Lemma 2.7, $\mathbf{I}$ will first choose $R_{j_1}^{\mathbf{x},\text{old}} \iff \text{dis}(\mathbf{I}, R_{j_1}^{\mathbf{x},\text{old}}) < \phi_1(0)$, that is, $\text{dis}(\mathbf{I}, R_{j_1}^{\mathbf{x},\text{old}})(n - |R_{j_2}^{\mathbf{x},\text{old}}|) < \text{dis}(\mathbf{I}, R_{j_2}^{\mathbf{x},\text{old}})(n - |R_{j_1}^{\mathbf{x},\text{old}}|)$. □

Next let us proceed to discuss the choosing order of the input fuzzy pattern by the fuzzy ART when the parameter $\alpha_{\mathbf{x}}$ is 'medium' or 'sufficiently large'. By Lemma 2.7 and Theorem 2.13, if $\mathbf{I} \in R_{j_1}^{\mathbf{x},\text{old}} \cap R_{j_2}^{\mathbf{x},\text{old}}$, the conclusion (i) of Theorem 2.13 holds when $\alpha_{\mathbf{x}}$ is 'medium' or 'sufficiently large'. So it suffices to solve our problem for $\mathbf{I} \in R_{j_2}^{\mathbf{x},\text{old}} \setminus R_{j_1}^{\mathbf{x},\text{old}}$ and $\mathbf{I} \notin R_{j_1}^{\mathbf{x},\text{old}} \cup R_{j_2}^{\mathbf{x},\text{old}}$.

Theorem 2.14 *Suppose the fuzzy pattern* $\mathbf{I} = (\mathbf{x}, \mathbf{x}^c)$ *is presented to the fuzzy ART, and* $\alpha_{\mathbf{x}}$ *is 'medium', i.e.* $0 < \alpha_{\mathbf{x}} < +\infty$, *moreover,* $|R_{j_1}^{\mathbf{x},\text{old}}| < |R_{j_2}^{\mathbf{x},\text{old}}|$. *Then*

(i) If $\mathbf{I} \in R_{j_2}^{\mathbf{x},\text{old}} \setminus R_{j_1}^{\mathbf{x},\text{old}}$, *we have,* $\mathbf{I}$ *will first choose* $R_{j_1}^{\mathbf{x},\text{old}}$ *if and only if*

$$\text{dis}\left(\mathbf{I}, R_{j_1}^{\mathbf{x},\text{old}}\right) < \phi_0(\alpha_{\mathbf{x}}). \tag{2.41}$$

Moreover, the function $\phi_0(\cdot)$ *is nondecreasing on* $\mathbb{R}$, *and*

$$0 < \phi_0(\alpha_{\mathbf{x}}) < |R_{j_2}^{\mathbf{x},\text{old}}| - |R_{j_1}^{\mathbf{x},\text{old}}|. \tag{2.42}$$

(ii) If $\mathbf{I} \notin R_{j_2}^{\mathbf{x},\text{old}} \cup R_{j_1}^{\mathbf{x},\text{old}}$, *then* $\mathbf{I}$ *will first choose* $R_{j_1}^{\mathbf{x},\text{old}}$ *if and only if*

$$\text{dis}\left(\mathbf{I}, R_{j_1}^{\mathbf{x},\text{old}}\right) < \phi_1(\alpha_{\mathbf{x}}). \tag{2.43}$$

And the function $\phi_1(\cdot)$ is nondecreasing on $\mathbb{R}$, moreover

$$\mathrm{dis}\big(\mathbf{I}, R_{j_2}^{\mathbf{x},\mathrm{old}}\big)\cdot\frac{n-|R_{j_1}^{\mathbf{x},\mathrm{old}}|}{n-|R_{j_2}^{\mathbf{x},\mathrm{old}}|} < \phi_1(\alpha_{\mathbf{x}}) < \mathrm{dis}\big(\mathbf{I}, R_{j_2}^{\mathbf{x},\mathrm{old}}\big)+\big|R_{j_2}^{\mathbf{x},\mathrm{old}}\big|-\big|R_{j_1}^{\mathbf{x},\mathrm{old}}\big|. \tag{2.44}$$

Proof. (i) At first, by Lemma 2.6, **I** will first choose $R_{j_1}^{\mathbf{x},\mathrm{old}}$ if and only if (2.41) is true. And we can conclude by computation that

$$\phi_0'(x) = \frac{\mathrm{d}\phi_0(x)}{\mathrm{d}x} = \frac{(n-|R_{j_2}^{\mathbf{x},\mathrm{old}}|)(|R_{j_2}^{\mathbf{x},\mathrm{old}}|-|R_{j_1}^{\mathbf{x},\mathrm{old}}|)}{(x+n-|R_{j_2}^{\mathbf{x},\mathrm{old}}|)^2}.$$

Using the fact $|R_{j_2}^{\mathbf{x},\mathrm{old}}|\leq n$, and the assumption we get, $|R_{j_2}^{\mathbf{x},\mathrm{old}}|-|R_{j_1}^{\mathbf{x},\mathrm{old}}|> 0$. Then $\phi_0'(x) \geq 0$, that is, $\phi_0(\cdot)$ is nondecreasing on $\mathbb{R}$. Moreover, $\phi_0(0) = 0$, $\phi_0(+\infty) \triangleq \lim_{x\to+\infty}\phi_0(x) = \big|R_{j_2}^{\mathbf{x},\mathrm{old}}\big|-\big|R_{j_1}^{\mathbf{x},\mathrm{old}}\big|$. Therefore by $\alpha_{\mathbf{x}} > 0$, $0 < \phi_0(\alpha_{\mathbf{x}}) < \phi_0(+\infty)$ it follows that (2.42) is true.

(ii) Using Lemma 2.7 we imply the first part of (ii) holds. it follows by computation that

$$\phi_1'(x) = \frac{\mathrm{d}\phi_0(x)}{\mathrm{d}x} = \frac{(n-|R_{j_2}^{\mathbf{x},\mathrm{new}}|)(|R_{j_2}^{\mathbf{x},\mathrm{old}}|-|R_{j_1}^{\mathbf{x},\mathrm{old}}|)}{(x+n-|R_{j_2}^{\mathbf{x},\mathrm{old}}|)^2}.$$

By assumption and the fact $\big|R_{j_2}^{\mathbf{x},\mathrm{new}}\big|\leq n$ we get, $\phi_1(x) \geq 0$. So $\phi_1(\cdot)$ is nondecreasing on $\mathbb{R}$. Denote $\phi_1(+\infty) \triangleq \lim_{x\to+\infty}\phi_1(x)$. It is easy to show

$$\phi_1(0)=\mathrm{dis}\big(\mathbf{I}, R_{j_2}^{\mathbf{x},\mathrm{old}}\big)\cdot\frac{n-|R_{j_1}^{\mathbf{x},\mathrm{old}}|}{n-|R_{j_2}^{\mathbf{x},\mathrm{old}}|},\ \phi_1(+\infty)=\mathrm{dis}\big(\mathbf{I}, R_{j_2}^{\mathbf{x},\mathrm{old}}\big)+\big|R_{j_2}^{\mathbf{x},\mathrm{old}}\big|-\big|R_{j_1}^{\mathbf{x},\mathrm{old}}\big|,$$

moreover $\phi_1(0) < \phi_1(\alpha_{\mathbf{x}}) < \phi_1(+\infty)$. Thus, (2.44) is true. □

When $\alpha_{\mathbf{x}} \to +\infty$, we have, $\phi_0(\alpha_{\mathbf{x}}) \to \big|R_{j_2}^{\mathbf{x},\mathrm{old}}\big|-\big|R_{j_1}^{\mathbf{x},\mathrm{old}}\big|$, and $\phi_1(\alpha_{\mathbf{x}}) \to \mathrm{dis}\big(\mathbf{I}, R_{j_2}^{\mathbf{x},\mathrm{old}}\big)+\big|R_{j_2}^{\mathbf{x},\mathrm{old}}\big|-\big|R_{j_1}^{\mathbf{x},\mathrm{old}}\big|$. Then by Lemma 2.6 and Lemma 2.7 easily we can obtain the following conclusion.

Theorem 2.15 *Suppose the fuzzy pattern* $\mathbf{I} = (\mathbf{x}, \mathbf{x}^c)$ *is presented to the fuzzy ART. The parameter* $\alpha_{\mathbf{x}} \approx +\infty$. *Then*

(i) If $\mathbf{I} \in R_{j_2}^{\mathbf{x},\mathrm{old}} \setminus R_{j_1}^{\mathbf{x},\mathrm{old}}$, *then* **I** *will first choose* $R_{j_1}^{\mathbf{x},\mathrm{old}}$ *if and only if* $\mathrm{dis}\big(\mathbf{I}, R_{j_1}^{\mathbf{x},\mathrm{old}}\big)< \big|R_{j_2}^{\mathbf{x},\mathrm{old}}\big|-\big|R_{j_1}^{\mathbf{x},\mathrm{old}}\big|$.

(ii) If $\mathbf{I} \notin R_{j_2}^{\mathbf{x},\mathrm{old}} \cup R_{j_1}^{\mathbf{x},\mathrm{old}}$, *then* **I** *will first choose* $R_{j_1}^{\mathbf{x},\mathrm{old}}$ *if and only if* $\mathrm{dis}\big(\mathbf{I}, R_{j_1}^{\mathbf{x},\mathrm{old}}\big)< \mathrm{dis}\big(\mathbf{I}, R_{j_2}^{\mathbf{x},\mathrm{old}}\big)+\big|R_{j_2}^{\mathbf{x},\mathrm{old}}\big|-\big|R_{j_1}^{\mathbf{x},\mathrm{old}}\big|$.

In the subsection we discuss how to choose committed nodes in the learning of the fuzzy ART. The main results come from [7, 14]. As for how to choose the uncommitted nodes by the fuzzy ART some tentative researches are presented in [14], which is also a meaningful problem for future research in the field related. Based on geometric interpretations of the vigilance test and the $F_2^{\mathbf{x}}$ layer competition of committed nodes with uncommitted ones, Anagnostopoulo and Georgiopoulos in [1] build a geometric concept related to fuzzy ART, category regions. It is useful for analyzing the learning of fuzzy ART, especially the stability of learning in fuzzy ART. Lin, Lin and Lee utilize the fuzzy ART learning algorithm as a main component to addresses the structure and the associated on-line learning algorithms of a feedforward multilayered network for realizing the basic elements and functions of a traditional fuzzy logic controller [30–32]. The network structure can be constructed from training examples by fuzzy ART learning techniques to find proper fuzzy partitions, membership functions, and fuzzy logic rules. Another important problem related fuzzy ART is how to find the appropriate vigilance range to improve its performance. We may build some robust and invariant pattern recognition models by solving such a problem [24].

2.4.3 Fuzzy ARTMAP

As a supervised learning neural network model the fuzzy ARTMAP consist of three modules, the fuzzy $\mathrm{ART}_{\mathbf{x}}$, fuzzy $\mathrm{ART}_{\mathbf{y}}$ and the inter-ART module, shown as in Figure 2.8.

Both fuzzy $\mathrm{ART}_{\mathbf{x}}$ and fuzzy $\mathrm{ART}_{\mathbf{y}}$ are the fuzzy ART's, accepting the inputs $\mathbf{x}$, $\mathbf{y}$, respectively. The inter-ART module consists of the field $F_{\mathbf{xy}}$ and the reset node 'Reset'. The main propose of $F_{\mathbf{xy}}$ is to classify a fuzzy pattern into the given class, or re-begin the matching procedure. For instance, when the fuzzy $\mathrm{ART}_{\mathbf{y}}$ generates a wrong match, by the fuzzy $\mathrm{ART}_{\mathbf{x}}$ the vigilance $\rho_{\mathbf{x}}$ increases, so that the maximum matching degree between the resonance fuzzy pattern and the input pattern can achieve.

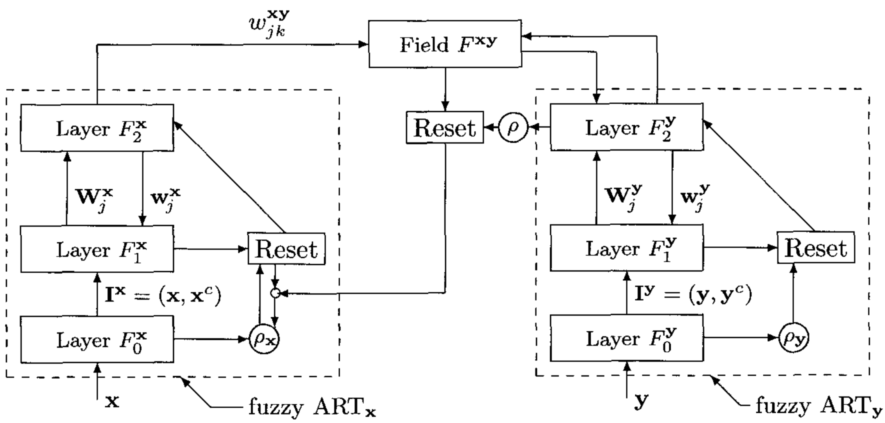


Figure 2.8 Fuzzy ARTMAP architecture

As the respective input patterns of fuzzy $\text{ART}_{\mathbf{x}}$ and fuzzy $\text{ART}_{\mathbf{y}}$, $\mathbf{I}^{\mathbf{x}} = (\mathbf{x}, \mathbf{x}^c)$, $\mathbf{I}^{\mathbf{y}} = (\mathbf{y}, \mathbf{y}^c)$ are two complement code, where $\mathbf{x}$ is a stimulus fuzzy pattern, and $\mathbf{y}$ is a response fuzzy pattern, which is a prediction of the fuzzy pattern $\mathbf{I}^{\mathbf{x}}$. In the fuzzy $\text{ART}_{\mathbf{x}}$, the output of layer $F_1^{\mathbf{x}}$ is $\mathbf{a}^{\mathbf{x}} = (a_1^{\mathbf{x}}, ..., a_{2n^{\mathbf{x}}}^{\mathbf{x}})$, and $\mathbf{b}^{\mathbf{x}} = (b_1^{\mathbf{x}}, ..., b_{m^{\mathbf{x}}}^{\mathbf{x}})$ is the output of layer $F_2^{\mathbf{x}}$. Let the $j-$th connection weight vector from $F_2^{\mathbf{x}}$ down to $F_1^{\mathbf{x}}$ be $\mathbf{w}_j^{\mathbf{x}} = (w_{j1}^{\mathbf{x}}, ..., w_{j(2n^{\mathbf{x}})}^{\mathbf{x}})$. Similarly, in the fuzzy $\text{ART}_{\mathbf{y}}$, suppose $\mathbf{a}^{\mathbf{y}} = (a_1^{\mathbf{y}}, ..., a_{2n^{\mathbf{y}}}^{\mathbf{y}})$ and $\mathbf{b}^{\mathbf{y}} = (b_1^{\mathbf{y}}, ..., b_{m^{\mathbf{y}}}^{\mathbf{y}})$ are the output patterns of $F_1^{\mathbf{y}}$ and $F_2^{\mathbf{y}}$, respectively. And let $\mathbf{w}_k^{\mathbf{y}} = (w_{k1}^{\mathbf{y}}, ..., w_{k(2n^{\mathbf{y}})}^{\mathbf{y}})$ be the $k-$th connection weight vector from $F_2^{\mathbf{y}}$ down to $F_1^{\mathbf{y}}$. Also we suppose $\mathbf{a}^{\mathbf{xy}} = (a_1^{\mathbf{xy}}, ..., a_{m^{\mathbf{y}}}^{\mathbf{xy}})$ is a output pattern of field $F^{\mathbf{xy}}$, and $\mathbf{w}_j^{\mathbf{xy}} = (w_{j1}^{\mathbf{xy}}, ..., w_{jm^{\mathbf{y}}}^{\mathbf{xy}})$ is the $j-$th connection weight vector of $F_2^{\mathbf{x}}$ to $F^{\mathbf{xy}}$.

In the inter-ART, field $F^{\mathbf{xy}}$ is called a map field, which accepts the outputs coming from the fuzzy $\text{ART}_{\mathbf{x}}$ and fuzzy $\text{ART}_{\mathbf{y}}$. Map field activation is governed by the activity of the fuzzy $\text{ART}_{\mathbf{x}}$ and the fuzzy $\text{ART}_{\mathbf{y}}$, in the following way:

$$\mathbf{a}^{\mathbf{xy}} = \begin{cases} \mathbf{b}^{\mathbf{y}} \wedge \mathbf{w}_{j^*}^{\mathbf{xy}}, & \text{the } j^* - \text{th } F_2^{\mathbf{x}} \text{ node is active and } F_2^{\mathbf{y}} \text{ is active,} \\ \mathbf{w}_{j^*}^{\mathbf{xy}}, & \text{the } j^* - \text{th } F_2^{\mathbf{x}} \text{ node is active and } F_2^{\mathbf{y}} \text{ is inactive,} \\ \mathbf{b}^{\mathbf{y}}, & F_2^{\mathbf{x}} \text{ is inactive and } F_2^{\mathbf{y}} \text{ is active,} \\ 0, & F_2^{\mathbf{x}} \text{ is inactive and } F_2^{\mathbf{y}} \text{ is inactive.} \end{cases}$$

If in $F_2^{\mathbf{x}}$ the j^*-th node is active, then its output can be transported to the field $F^{\mathbf{xy}}$ through the weight vector $\mathbf{w}_{j^*}^{\mathbf{xy}}$. And $\mathbf{w}_{j^*}^{\mathbf{xy}}$ may be classified into a defined fuzzy pattern class. If $F_2^{\mathbf{y}}$ is active, then only when a identical fuzzy pattern class is obtained by fuzzy $\text{ART}_{\mathbf{x}}$ and fuzzy $\text{ART}_{\mathbf{y}}$, respectively $F^{\mathbf{xy}}$ is active. If a mis-match between $\mathbf{b}^{\mathbf{y}}$ and $\mathbf{w}_{j^*}^{\mathbf{xy}}$ takes place, $\mathbf{a}^{\mathbf{xy}} = \mathbf{0}$, and the search procedure is active.

Searching match. When the system accepts an input pattern, the vigilance $\rho_{\mathbf{x}}$ of $\text{ART}_{\mathbf{x}}$ equals to the minimum value $\overline{\rho}_{\mathbf{x}}$, and the vigilance of $F_{\mathbf{xy}}$ is ρ. If $|\mathbf{a}^{\mathbf{xy}}| < \rho \cdot |\mathbf{b}^{\mathbf{y}}|$, then we increase $\rho_{\mathbf{x}}$, so that $\rho_{\mathbf{x}} > |\mathbf{I}^{\mathbf{x}} \wedge \mathbf{w}_{j^*}^{\mathbf{x}}| \cdot |\mathbf{I}^{\mathbf{x}}|^{-1}$, and thus

$$|\mathbf{a}^{\mathbf{x}}| = |\mathbf{I}^{\mathbf{x}} \wedge \mathbf{w}_{j^*}^{\mathbf{x}}| < \rho_{\mathbf{x}} \cdot |\mathbf{I}^{\mathbf{x}}|,$$

where j^* means an active node in $F_2^{\mathbf{x}}$. Thus, through the search procedure of $\text{ART}_{\mathbf{x}}$ we can obtain the fact: either there is an active node j^* in $F_2^{\mathbf{x}}$, satisfying

$$\begin{cases} |\mathbf{a}^{\mathbf{x}}| = |\mathbf{I}^{\mathbf{x}} \wedge \mathbf{w}_{j^*}^{\mathbf{x}}| \geq \rho_{\mathbf{x}} \cdot |\mathbf{I}^{\mathbf{x}}|; \\ |\mathbf{a}^{\mathbf{xy}}| = |\mathbf{b}^{\mathbf{y}} \wedge \mathbf{w}_{j^*}^{\mathbf{xy}}| \geq \rho \cdot |\mathbf{b}^{\mathbf{y}}|, \end{cases}$$

or there is no such a node, then $F_2^{\mathbf{x}}$ stop the expressing procedure of the input patterns.

Learning of map field. The connection weight $w_{jk}^{\mathbf{xy}}$ of $F_2^{\mathbf{x}} \longrightarrow F^{\mathbf{xy}}$ is trained with the following scheme:

Step 1. Initialize: $w_{jk}^{\mathbf{xy}}(0) = 1$;

Step 2. Resonance: If the resonance corresponding to the active node j^* of $\text{ART}_\mathbf{x}$ takes place, let $\mathbf{w}_{j^*}^{\mathbf{xy}} = \mathbf{a}^{\mathbf{xy}}$;

Step 3. Prediction: If the fuzzy pattern corresponding to the node j^* in $\text{ART}_\mathbf{x}$ can predict the fuzzy pattern corresponding to the node k^* in $\text{ART}_\mathbf{y}$ we let $w_{j^*k^*}^{\mathbf{xy}} = 1$.

All equations of the subsection 2.4.2 for fuzzy ART module are valid for the $\text{ART}_\mathbf{x}$ and $\text{ART}_\mathbf{y}$ modules of fuzzy ARTMAP. If we are only focusing on pattern classification tasks, the templates formed in $\text{ART}_\mathbf{y}$ are not very interesting. To enforce this type of clustering in $\text{ART}_\mathbf{y}$ the vigilance parameter $\rho_\mathbf{y}$ in $\text{ART}_\mathbf{y}$ is chosen equal to one. The templates formed in $\text{ART}_\mathbf{x}$ are a different story. The discussion in the subsection 2.4.2 about templates in fuzzy ART is still valid for templates in the $\text{ART}_\mathbf{x}$ module of the fuzzy ARTMAP. Moreover, the definition of a distance in fuzzy ART, mentioned in 2.4.2, is also valid for the $\text{ART}_\mathbf{x}$ module of the fuzzy ARTMAP.

In the following, we explain the classification results of fuzzy ARTMAP about the input fuzzy patterns. Lemma 2.5, Lemma 2.6, Lemma 2.7 and Theorems 2.13, 2.14, 2.15 are applicable without any modification for the $\text{ART}_\mathbf{x}$ module of the fuzzy ARTMAP. The order of search, established for fuzzy ART is also applicable for the $\text{ART}_\mathbf{x}$ module of the fuzzy ARTMAP. Similarly with with the fuzzy ART, Theorem 2.13, Theorem 2.14 and Theorem 2.15 explain also how the fuzzy ARTMAP chooses among the committed nodes in the $\text{ART}_\mathbf{x}$ module.

Fuzzy ARTMAP can also be employed to deal with noisy data. For example, Marriott and Harrison use it to approximate a noisy continuous mapping, and a fuzzy ARTMAP variant is built [40]. Some meaningful and important problems related to this subject include building novel fuzzy ARTMAP models to improve the classification performances [1, 17], designing some efficient learning algorithms related to accelerate the classifying procedures [17] and analyzing the convergence of learning algorithms of fuzzy ARTMAP [15, 16] and so on.

2.4.4 Real examples

Now let us take some real examples to illustrate the effectiveness of Theorems 2.13, 2.14 and 2.15. To justify the large $\alpha_\mathbf{x}$ values of the aforementioned fuzzy ART variant, i.e. corresponding to $M_\mathbf{x} \to +\infty$, $\alpha_\mathbf{x} \to +\infty$, we compare its performance with one of the original fuzzy ART algorithm ($M_\mathbf{x} = 2n$, $\alpha_\mathbf{x}$ is reasonably small). The criterion for the comparison is the average clustering performance of the algorithms related. To this end we choose five databases coming from [14], they are, heart the disease database, the diabetes database, the wine recognition database, the ionosphere database and the sonar database. Then, for different vigilance values ρ's the average clustering performance of the fuzzy ART variant is evaluated for each database. We also calculate the average clustering performance of the original fuzzy ART algorithm for the same databases, the same vigilance value and a wide range of $\alpha_\mathbf{x}$ values.

Table 2.4 Comparison of clustering performance

			Av-l	St-l	Av-n	St-n
Heart Disease	$\rho = 0.3$	$\alpha_{\mathbf{x}} \to \infty$	57.7%	1.64%	5.80	0.00
		$\alpha_{\mathbf{x}} = 0.6$	57.9%	1.44%	5.50	0.53
	$\rho = 0.4$	$\alpha_{\mathbf{x}} \to \infty$	62.7%	2.34%	10.3	0.79
		$\alpha_{\mathbf{x}} = 13.0$	65.1%	2.04%	15.6	1.35
		$\alpha_{\mathbf{x}} = 0.01$	61.4%	1.75%	4.00	1.18
	$\rho = 0.8$	$\alpha_{\mathbf{x}} \to \infty$	77.8%	1.57%	60.4	2.72
		$\alpha_{\mathbf{x}} = 6.0$	74.6%	2.08%	55.0	2.89
Diabetes	$\rho = 0.2$	$\alpha_{\mathbf{x}} \to \infty$	65.7%	0.83%	2.90	0.32
		$\alpha_{\mathbf{x}} = 0.8$	65.2%	0.62%	2.50	0.53
	$\rho = 0.4$	$\alpha_{\mathbf{x}} \to \infty$	71.4%	3.40%	9.90	0.74
		$\alpha_{\mathbf{x}} = 7.4$	72.4%	3.10%	14.7	0.95
		$\alpha_{\mathbf{x}} = 0.01$	69.4%	1.80%	6.30	0.48
	$\rho = 0.9$	$\alpha_{\mathbf{x}} \to \infty$	91.1%	1.05%	245.7	2.97
		$\alpha_{\mathbf{x}} = 2.6$	92.9%	0.88%	234.3	4.22
Wine Recog.	$\rho = 0.5$	$\alpha_{\mathbf{x}} \to \infty$	68.2%	6.95%	2.00	0.32
		$\alpha_{\mathbf{x}} = 5.0$	63.2%	6.98%	3.00	0.00
	$\rho = 0.6$	$\alpha_{\mathbf{x}} \to \infty$	82.8%	7.73%	5.60	0.92
		$\alpha_{\mathbf{x}} = 24.0$	90.2%	4.06%	8.80	0.45
		$\alpha_{\mathbf{x}} = 0.01$	87.8%	2.74%	5.20	0.45
	$\rho = 0.9$	$\alpha_{\mathbf{x}} \to \infty$	98.2%	0.86%	53.8	1.23
		$\alpha_{\mathbf{x}} = 0.8$	98.8%	0.75%	53.6	1.52
Ionosph.	$\rho = 0.2$	$\alpha_{\mathbf{x}} \to \infty$	93.0%	1.76%	15.0	1.00
		$\alpha_{\mathbf{x}} = 12.6$	78.7%	2.99%	17.0	0.94
	$\rho = 0.4$	$\alpha_{\mathbf{x}} \to \infty$	94.6%	1.07%	31.7	2.03
		$\alpha_{\mathbf{x}} = 34.0$	87.0%	2.24%	37.1	1.45
		$\alpha_{\mathbf{x}} = 0.01$	91.9%	2.69%	26.3	1.64
	$\rho = 0.7$	$\alpha_{\mathbf{x}} \to \infty$	97.2%	1.53%	63.6	1.88
		$\alpha_{\mathbf{x}} = 0.6$	94.7%	2.70%	60.6	1.63
Sonar	$\rho = 0.4$	$\alpha_{\mathbf{x}} \to \infty$	54.9%	2.24%	2.00	0.00
		$\alpha_{\mathbf{x}} = 0.8$	54.3%	2.73%	2.00	0.00
	$\rho = 0.6$	$\alpha_{\mathbf{x}} \to \infty$	86.9%	5.59%	10.2	1.03
		$\alpha_{\mathbf{x}} = 30.0$	87.0%	3.30%	9.50	1.08
		$\alpha_{\mathbf{x}} = 0.01$	93.7%	2.28%	9.80	0.79
	$\rho = 0.8$	$\alpha_{\mathbf{x}} \to \infty$	92.6%	1.64%	34.4	2.17
		$\alpha_{\mathbf{x}} = 0.6$	95.4%	1.91%	33.2	2.30

The testing results are reported in Table 2.4, where for ease of presentation we report the average clustering performance of the original fuzzy ART only for selective $\alpha_{\mathbf{x}}$ values. Also we give the average number of $F_2^{\mathbf{x}}$ nodes created by the fuzzy ART variant and the original fuzzy ART algorithm. Moreover, standard deviation of the clustering performance, and the number of nodes are also reported in Table 2.4.

In Table 2.4, 'Av-l' means average percentage of correct clustering; 'St-l' means standard deviation of correct clustering; 'Av-n' means average number of $F_2^{\mathbf{x}}$ nodes, and 'St-n' means standard deviation of $F_2^{\mathbf{x}}$ nodes. The training list of each database consists of inputs and corresponding output patterns. For the training of fuzzy ART only the input patterns of the training list are used. In the heart disease database, $n = 13$, and there are 203 input patterns in the training list, which belong to five different classes. In the diabetes database $n = 8$, and the training list consists of 576 input patterns belonging to two different classes. In the wine recognition database, $n = 12$ and there exist 120 input patterns in the training list, which belong to three different classes. In the ionosphere database $n = 34$, and the training list consists of 200 input patterns belonging to two different classes. In the wine recognition database, $n = 60$ and there exist 104 input patterns in the training list, which belong to two different classes.

After the training is over, we assign a label to each category formed in $F_2^{\mathbf{x}}$ layer of the fuzzy ART, by the output pattern to which most of the input patterns that are expressed by this category are mapped. For each input pattern in the training list, fuzzy ART chooses a category in $F_2^{\mathbf{x}}$ layer. If the label of this category is the output pattern that this pattern corresponds to in the training list, then we say that fuzzy ART makes a correct clustering. If, on the other hand, the label of this category is different from the output pattern that this pattern corresponds to in the training list, then we say that fuzzy ART clusters this input pattern erroneously. From Table 2.4 we can know, the order of search results for fuzzy ART are of practical significance for small, intermediate and large $\alpha_{\mathbf{x}}$ values. However, the very large choices for the choice parameter value $\alpha_{\mathbf{x}}$ and the parameter $M^{\mathbf{x}}$ are not a good combination of parameter values for fuzzy ARTMAP. Fuzzy ARTMAP simulation results conducted with these parameter choices indicate that fuzzy ARTMAP creates too many clusters in $\mathrm{ART}_{\mathbf{x}}$ for solving the pattern classifications corresponding to the five databases mentioned above, and as a result they are not worth mentioning here. Thus, the fuzzy ARTMAP simulations with very large $M^{\mathbf{x}}$ and $\alpha_{\mathbf{x}}$ values are not practical. So the classifications obtained by fuzzy ARTMAP are of practical significance only for small, intermediate $\alpha_{\mathbf{x}}$ values.

Above evaluating procedure is motivated by Dubes and Jain [11], where a number of clustering techniques are compared with each other. The average performance shown in Table 2.4 are calculated by training fuzzy ART with ten different orders of pattern presentations from the training list.

References

[1] Anagnostopoulo G. C. & Georgiopoulos M., Category regions as new geometrical concepts in fuzzy-ART and fuzzy-ARTMAP, *Neural Networks,* 2002, **15**(10): 1205–1221.

[2] Blanco A., Delgado M. & Requena I., Identification of relational equations by fuzzy neural networks, *Fuzzy Sets and Systems,* 1995, **71**: 215–226.

[3] Blanco A., Delgado M. & Requena I., Improved fuzzy neural networks for solving relational equations, *Fuzzy Sets and Systems,* 1995, **72**: 311–322.

[4] Buckley J. J. & Hayashi Y., Fuzzy neural networks: A survey, *Fuzzy Sets and Systems,* 1994, **66**: 1–13.

[5] Buckley J. J. & Hayashi Y., Fuzzy genetic algorithm and applications, *Fuzzy Sets and Systems,* 1994, **61**: 129–136.

[6] Carpenter G. A. & Grossberg S., A massively parallel architecture for a self-organizing neural pattern recognition, *Computer Vision, Graphics, and Image Processing,* 1987, **37**: 54–115.

[7] Carpenter G. A., Grossberg S. & Rosen D. B., Fuzzy ART: fast stable learning and categorization of analog patterns by an adaptive resonance system, *Neural Networks*, 1991, **4**(6): 759–771.

[8] Carpenter G. A., Grossberg S. & Markuzon S., et al, Fuzzy ARTMAP: A neural network architecture for incremental supervised learning of analog multi-dimensional maps, *IEEE Trans. on Neural Networks*, 1992, **3**(5): 698–713.

[9] Chung F. L. & Lee T., On fuzzy associative memory with multiple-rule storage capacity, *IEEE Trans. on Fuzzy Systems,* 1996, **4**: 375-384.

[10] Chung F. L. & Lee T., Fuzzy Competitive learning, *Neural Networks,* 1994, **7**: 539-552.

[11] Dubes R. & Jain A., Clustering techniques: the user's dilemma, *Pattern Recognition,* 1976, **8**: 247–260.

[12] Fan Junjie, Jin Fan & Shi Yan, An efficient learning algorithm of fuzzy associative memory, *Acta Elect. Sinica*(In Chinese), 1996, **24**(1): 112–114.

[13] Frank T. & Kraiss K. -F., Comparative analysis of Fuzzy ART and ART-2A network clustering performance, *IEEE Trans. on Neural Networks,* 1998, **9**(3): 544–559.

[14] Georgiopoulos M., Fernlund H. & Bebis G. et al, Order of search in fuzzy ART and fuzzy ARTMAP: effect of the choice parameter, *Neural Networks*, 1996, **9**(9): 1541–1559.

[15] Georgiopoulos M., Heileman G. L. & Huang J., Convergence properties of learning in ART1, *Neural Networks,* 1990, **2**(4): 502–509.

[16] Georgiopoulos M., Huang J. & Heileman G. L., Properties of learning in ARTMAP, *Neural Networks,* 1994, **7**(4): 495–506.

[17] Gomez-Sanchez E., Dimitriadis Y. A. & Cano-Izquierdo J., et al, μARTMAP: use of mutual information for category reduction in fuzzy ARTMAP, *IEEE Trans. on Neural Networks,* 2002, **13**(1): 58–69.

[18] Gupta M. M. & Fao D. H., On the principles of fuzzy neural networks, *Fuzzy Sets and Systems,* 1994, **61**: 1–18.

[19] Gupta M. M. & Qi J., Design of fuzzy logic controllers based on generalized T-operators, *Fuzzy Sets and Systems,* 1991, **40**: 473–489.

[20] Hassoun M. H. & Watta P. B., Exact associative neural memory dynamics utilizing Boolean matrices, *IEEE Trans. on Neural Networks,* 1991, **2**: 437–448.

[21] Huang Jiejun & Xie Weixin, Regular fuzzy associative memory, *Acta Electron. Sinica*(In Chinese), 1997, **25**(7): 68–71.

[22] Jang J. S. R., Sun C. T. & Mizutani E., *Neuro-fuzzy and soft computing,* Prentice Hall, 1997.

[23] Kang Hoon, Multilayer associative neural networks (MANN's): storage capacity versus perfect recall, *IEEE Trans.on Neural Networks,* 1994, **5**: 812–822.

[24] Kim M. -H., Jang D. -S. & Yang Y. -K., A robust-invariant pattern recognition model using Fuzzy ART, *Pattern Recognition,* 2001, **34**(8): 1685–1696.

[25] Kosko B., Fuzzy associative memories. In Kandel A.(ed.), *Fuzzy Expert Systems Reading,* MA: Addison-Weley, 1987.

[26] Lee C. Y., Wang J. S. & Lee R. C. T., Characteristics of the Hopfield associative memory utilizing isomorphism relations, *IEEE Trans. on Neural Networks,* 1994, **5**: 523–526.

[27] Li X. Z. & Ruan Da, Novel neural algorithms based on fuzzy δ rules for solving fuzzy relation equations: Part I, *Fuzzy Sets and Systems,* 1997, **90**: 11–23.

[28] Li X. Z. & Ruan Da, Novel neural algorithms based on fuzzy δ rules for solving fuzzy relation equations: Part II, *Fuzzy Sets and Systems,* 1999, **103**: 473–486.

[29] Li X. Z. & Ruan Da, Novel neural algorithms based on fuzzy δ rules for solving fuzzy relation equations: Part III, *Fuzzy Sets and Systems,* 2000, **109**: 355–362.

[30] Lin C. -J. & Lin C. -T., An ART-based Fuzzy adaptive learning control network, *IEEE Transactions on Fuzzy Systems,* 1997, **5**(4): 477–496.

[31] Lin C. T. & Lee C. J. G., Neural network based fuzzy logic control and decision system, *IEEE Trans. on Computer,* 1991, **40**(12).

[32] Lin C. -T., Lin C. -J. & Lee C. S. G., Fuzzy adaptive learning control network with on-line neural learning, *Fuzzy Sets and Systems,* 1995, **71**(1): 25–45.

[33] Liu Puyin, Fuzzy associative memory of fuzzy neural network based on a class of norm operations, *Systems Engineering and Elect.*(In Chinese), 1997, **19**(11): 54–59.

[34] Liu Puyin, Stability and fault–tolerance of the fuzzy bidirectional associative memory, *Control and Decision*(In Chinese), 1999, **14**(5): 443–447.

[35] Liu Puyin, A learning algorithm based on fault-tolerance for Min-max fuzzy Hopfield networks, *J. National Uni. Defence Tech.*(In Chinese), 1998, **20**(1): 116–123.

[36] Liu Puyin, Zhang Hanjiang & Wu Mengda,et al, Fuzzy neural network theory–a survey, *Fuzzy System and Math.*(In Chinese), 1998, **12**(1): 77–87.

[37] Liu Puyin & Zhang Hanjiang, Analysis for stability and fault-tolerance of min-max fuzzy Hopfield network with threshold, *Fuzzy Systems and Math.,* 1998, **12**(4): 47-55.

[38] Liu Puyin, The fuzzy associative memory of max-min fuzzy neural networks with threshold, *Fuzzy Sets and Systems,* 1999, **107**: 147–157.

[39] Liu Zengliang & Liu Youcai, *Fuzzy Logic and neural network—theory and application*, Beijing: Press of Beijing Uni. of Aero. and Astro.,1996.

[40] Marriott S. & Harrison R., A modified fuzzy ARTMAP architecture for the approximation of noise mappings, *Neural Networks,* 1995, **8**(4): 619–641.

[41] Miyakoshi M. & Shimbo M., Solutions of composite fuzzy relational equations with triangular norms, *Fuzzy Sets and Systems,* 1985, **16**(1): 53–63.

[42] Nobuhara H., Pedrycz W. & Hirota K, Fast solving method of fuzzy relational equation and its application to lossy image compression/reconstruction, *IEEE Trans. on Fuzzy Systems,* 2000, **8**: 325–334.

[43] Pal S. K. & Mitra S., Multi-layer perceptron, fuzzy sets, and classification, *IEEE Trans. on Neural Networks,* 1992, **3**: 683–697.

[44] Pedrycz W., Fuzzy neural networks and neuro-computations, *Fuzzy Sets and Systems,* 1993, **56**: 1–28.

[45] Simpson P. K., Fuzzy min-max neural networks-part 1: classification, *IEEE Trans. on Neural Networks,* 1992, **3**: 777-786.

[46] Simpson P.K., Fuzzy Min-Max neural networks-part 2: clustering, *IEEE Trans.Fuzzy Systems,* 1993,**1**:32–45.

[47] Wang L. X., *Adaptive fuzzy systems and control:design stability analysis,* PTR Prentice-Hall, Englewood Cliffs, NJ, 1994.

[48] Wang Z. F., Jin D. M. & Li Z. J., Research on monolithic fuzzy neural networks self-learning problems, *Acta Elect. Sinica* (in Chinese), 1997, **25**(1): 33–38.

[49] Wiggins R., Docking a truck: A genetic fuzzy approach, *AI Expert,* 1992, **7**:28–35.

[50] Zhang Chengfu & Zhao Gang, Some problems related to associative memory neural network, *Acta Automatica Sinica* (In Chinese), 1994, **20**(5): 513–521.

[51] Zhang X. H., Hang C. C. & Tan S. H., et al, The Min-max function differentiation and training of fuzzy neural networks, *IEEE Trans. on Neural Networks,* 1996, **7**: 1139–1150.

[52] Zhang Yirong & Liu Puyin, Stability analysis of a class of fuzzy Hopfield networks based on T–norm, *Acta Elect. Sinica* (in Chinese), 2002, **30**(1): 30–33.

CHAPTER III
Fuzzy Associative Memory—Feedback Networks

Fuzzy associative memory in the preceding chapter is a feedforward system whose information flows from input layer to output layer. So such a neural network is static. The research related is closely associated with fuzzy relation equation theory [2, 3, 40, 43]. In the chapter we proceed to study some further subjects of Chapter II, that is, the feedback versions of FAM's, which is a feedback FNN. So the research of feedback FAM's includes their topological architectures, learning algorithm for connection weights and the dynamical properties related, such as stability of the systems, attractors and attractive basin and so on [28–33, 36, 37, 45, 46]. Bidirectionality, or forward and backward information flow, is utilized to produce a two–way associative search for fuzzy pattern pairs. An important attribute of the feedback FAM's is their ability to retrieve some stored patterns from their noise or partial inputs. The achievements related have found very useful in many application areas, for instance, pattern recognition [13], system forecast [16, 18], control and decision [18], etc. As in Chapter II we focus on the fuzzy operator pair '$\wedge - \vee$' also, and the feedback FAM's based on '$\wedge - \vee$' in the chapter.

In the following we present the systematic study to two classes of FNN's, they are fuzzy Hopfield networks and fuzzy bidirectional associative memory (FBAM). Fuzzy Hopfield networks are the fuzzy versions of the discrete Hopfield networks, in which the operations '$+$' '$\times$' are replaced by the composition of the fuzzy operators '$\vee$' '$\wedge$', and $0-1$ string information is superseded by fuzzy information with the vector form whose components belonging to $[0, 1]$. Similarly, a FBAM can be established based on one bidirectional associative memory.

§3.1 Fuzzy Hopfield networks

To develop a FNN system with real sense, an unavoidable problem that we have to deal with is fault–tolerance of the system, since in practice fuzzy information to be processed is always corrupted or distorted, or some error information is contained. So it is very important and meaningful for pattern recognition, system identification and forecasting and so on how to design a FNN so that right fuzzy patterns can be recalled even with some distorted input patterns. The subjects in this area now begin to attract many scholars' attention [1, 15], while in the past much more attention was focused on the storage capability of FNN's [8, 10, 11]. In the section we aim mainly at the following fuzzy Hopfield network based on the fuzzy operator pair '$\vee - \wedge$':

$$\mathbf{x}(t) = \mathbf{x}(t-1) \circ W \tag{3.1}$$

where $t = 1, 2, ...$ is the iteration step of the system, '$\circ$' is $\vee - \wedge$ composite operation, $W = (w_{ij})_{n\times n}$ is a connection weight matrix, $\mathbf{x}(0)$ is an initial fuzzy pattern. In the following we at first analyze the dynamic properties of (3.1), including the system state when the iteration step $t \to +\infty$, that is, the attractors of the system and the corresponding attractive basins. Then we present an analytic learning algorithm for (3.1). For a given fuzzy pattern family the learning algorithm can lead to such a connection weight matrix that all fuzzy patterns are attractors, the corresponding attractive basins are largest and consequently the system possesses optimal fault–tolerance. Finally these results are illustrated by some simulation examples. As in chapter II we write $\mathrm{N} = \{1, ..., n\}$, $\mathrm{P} = \{1, ..., p\}$, n, p are natural numbers.

3.1.1 Attractor and attractive basin

In the following let us establish the attractors and attractive basins for the system (3.1). As a subset of $[0, 1]^n$ the attractive basins here are larger than ones in [29, 34].

Definition 3.1 The fuzzy pattern $\mathbf{b}$ is called an attractor of (3.1) if $\mathbf{b} = \mathbf{b}\circ W$. Provided $\mathbf{b}$ is an attractor of (3.1), and there is a set $F_q \subset [0,1]^n$, so that $\mathbf{b} \in F_q$, moreover, $\forall \mathbf{x} \in F_q$, Taken $\mathbf{x}$ as an initial fuzzy pattern, the system (3.1) converges to $\mathbf{b}$. Then we call F_q an attractive basin of $\mathbf{b}$. The attractive basin F_q is called non-degenerate if the volume of F_q as a subset of $[0, 1]^n$ is nonzero.

Fault–tolerance of (3.1) means the capability of the system to recall a right fuzzy pattern when the input signal (initial pattern) is corrupted. Obviously, fault–tolerance of a dynamic system can be determined by the sizes of attractive basins of the corresponding attractors. So it is necessary for (3.1) to possess fault–tolerance that the system is trained to guarantee the attractive basins of the corresponding attractors are non-degenerate. To this end we suppose $\mathbf{b} = (b_1, ..., b_n)$ is a fuzzy pattern and $W = (w_{ij})_{n\times n}$ is a connection weight matrix of (3.1). Denote

$$H^G(W, \mathbf{b}, i) = \{j \in \mathrm{N} | w_{ij} > b_j\}, \quad H^E(W, \mathbf{b}, i) = \{j \in \mathrm{N} | w_{ij} = b_j\},$$

where $i \in \mathrm{N}$, and let

$$H_0^G(W, \mathbf{b}) = \{i \in \mathrm{N} \big| H^G(W, \mathbf{b}, i) \neq \emptyset, \ H^E(W, \mathbf{b}, i) = \emptyset\};$$

$$H_0^L(W, \mathbf{b}) = \{i \in \mathrm{N} \big| H^G(W, \mathbf{b}, i) \cup H^E(W, \mathbf{b}, i) = \emptyset\};$$

$$H_0^{Ge}(W, \mathbf{b}) = \{i \in \mathrm{N} \big| H^G(W, \mathbf{b}, i) \neq \emptyset, \ H^E(W, \mathbf{b}, i) \neq \emptyset\};$$

$$H_0^E(W, \mathbf{b}) = \{i \in \mathrm{N} \big| H^G(W, \mathbf{b}, i) = \emptyset, \ H^E(W, \mathbf{b}, i) \neq \emptyset\}.$$

$$b_i^1(W,\mathbf{b})=\begin{cases}\bigvee\limits_{j\in H^E(W,\mathbf{b},i)}\{b_j\}, & H^E(W,\mathbf{b},i)\neq\emptyset;\\ 0, & H^E(W,\mathbf{b},i)=\emptyset.\end{cases}$$

$$b_i^2(W,\mathbf{b})=\begin{cases}\bigwedge\limits_{j\in H^G(W,\mathbf{b},i)}\{b_j\}, & H^G(W,\mathbf{b},i)\neq\emptyset;\\ 1, & H^G(W,\mathbf{b},i)=\emptyset.\end{cases}$$

Let us now introduce other two sets:

$$H_I^E(W,\mathbf{b})=\big\{i\in \mathrm{N}\big|H^E(W,\mathbf{b},i)=\mathrm{N}\big\},$$
$$H_I^{Ge}(W,\mathbf{b})=\big\{i\in H_0^{Ge}(W,\mathbf{b})\big|H^E(W,\mathbf{b},i)\cup H^G(W,\mathbf{b},i)=\mathrm{N}\big\}.$$

Write

$$i_e=\begin{cases}\min\{i|i\in H_I^E(W,\mathbf{b})\}, & H_I^E(W,\mathbf{b})\neq\emptyset,\\ n+1, & H_I^E(W,\mathbf{b})=\emptyset;\end{cases}$$

$$i_{ge}=\begin{cases}\min\{i|i\in H_I^{Ge}(W,\mathbf{b})\}, & H_I^{Ge}(W,\mathbf{b})\neq\emptyset,\\ n+1, & H_I^{Ge}(W,\mathbf{b})=\emptyset.\end{cases}$$

For the set $C\subset \mathrm{N}$, we define the following function:

$$\varpi(C)=\begin{cases}1, & C=\emptyset;\\ 0, & C\neq\emptyset.\end{cases}\tag{3.2}$$

If $i\in\mathrm{N}$, $\forall j_1\in H^E(W,\mathbf{b},i)$, $j_2\in H^G(W,\mathbf{b},i)$, we have $b_{j_1}<b_{j_2}$. Let

$$\begin{aligned}A_t^0(W,\mathbf{b})=\Big\{&(x_1,...,x_n)\in[0,1]^n\Big|x_{i_{ge}}\in\big[b_{i_{ge}}^1(W,\mathbf{b}),\ b_{i_{ge}}^2(W,\mathbf{b})\big],\\ &x_{i_e}\in\big[b_{i_e}^1(W,\mathbf{b}),\ 1\big],\ x_i\in[0,1]\,(i\in H_0^L(W,\mathbf{b})),\\ &x_i\in\big[b_i^1(W,\mathbf{b})\cdot\varpi(H_I^{Ge}(W,\mathbf{b})),\ b_i^2(W,\mathbf{b})\big]\ (i\in H_0^{Ge}(W,\mathbf{b})\setminus\{i_{ge}\}),\\ &x_i\in\big[b_i^1(W,\mathbf{b})\cdot\varpi(H_I^E(W,\mathbf{b})),\ 1\big]\ (i\in H_0^E(W,\mathbf{b})\setminus\{i_e\}),\\ &x_i\in\big[0,\ b_i^2(W,\mathbf{b})\big]\ (i\in H_0^G(W,\mathbf{b}))\Big\}.\end{aligned}\tag{3.3}$$

By (3.3) it follows that $A_t^0(W,\mathbf{b})$ includes the attractive basin obtained in [32].

Definition 3.2 We call (W, $\mathbf{b}$) to satisfy GE condition, if the following facts hold:

(i) $\forall j\in\mathrm{N}$, there exists $i\in\mathrm{N}$, so that $j\in H^E(W,\mathbf{b},i)$;

(ii) $\forall i\in\mathrm{N}$, and $j_2\in H^G(W,\mathbf{b},i)$, $j_1\in H^E(W,\mathbf{b},i)$, we have, $b_{j_1}<b_{j_2}$.

Theorem 3.1 *Suppose the fuzzy pattern* $\mathbf{b}=(b_1,...,b_n)$ *is an attractor of (3.1).* $W=(w_{ij})_{n\times n}$ *is the connection weight matrix. Moreover,* (W, $\mathbf{b}$) *satisfies GE condition. Then for any* $\mathbf{x}=(x_1,...,x_n)\in A_t^0(W,\mathbf{b})$, $\mathbf{x}$ *converges to* $\mathbf{b}$ *by one iteration.*

Proof. By the assumption easily we have, $\mathbf{b} \in A_t^0(W, \mathbf{b})$, and for $i \in$ N, $b_i^1(W, \mathbf{b}) < b_i^2(W, \mathbf{b})$. Given $j \in$ N, and the fuzzy pattern $\mathbf{x} = (x_1, ..., x_n) \in A_t^0(W, \mathbf{b})$. If $H_I^E(W, \mathbf{b}) \neq \emptyset$, then $i_e \in H_0^E(W, \mathbf{b})$ and $H^E(W, \mathbf{b}, i_e) =$ N. Therefore by (3.3) we can get

$$\begin{aligned}\bigvee_{i \in H_0^E(W,\mathbf{b}) \cup H_0^{Ge}(W,\mathbf{b})} \{x_i \wedge w_{ij}\} &\geq \bigvee_{i \in H_0^E(W,\mathbf{b})} \{x_i \wedge w_{ij}\} \\ &\geq x_{i_e} \wedge w_{i_e j} \geq \Big(\bigvee_{k \in \mathrm{N}} \{b_k\}\Big) \wedge w_{i_e j} \geq b_j.\end{aligned} \tag{3.4}$$

If $H_I^{Ge}(W, \mathbf{b}) \neq \emptyset$, with the same reason we get

$$\bigvee_{i \in H_0^E(W,\mathbf{b}) \cup H_0^{Ge}(W,\mathbf{b})} \{x_i \wedge w_{ij}\} \geq b_j.$$

If $H_I^{Ge}(W, \mathbf{b}) \cup H_I^E(W, \mathbf{b}) = \emptyset$, then $\varpi(H_I^E(W, \mathbf{b})) = \varpi(H_I^{Ge}(W, \mathbf{b})) = 1$. Since $(W, \mathbf{b})$ satisfies GE condition, choose $i_0 \in$ N, so that $j \in H^E(W, \mathbf{b}, i_0)$. Thus, $i_0 \in H_0^{Ge}(W, \mathbf{b}) \cup H_0^E(W, \mathbf{b})$. Considering (3.3) and $j \in H^E(W, \mathbf{b}, i_0)$, easily we can imply

$$\begin{aligned}\bigvee_{i \in H_0^E(W,\mathbf{b}) \cup H_0^{Ge}(W,\mathbf{b})} (x_i \wedge w_{ij}) &\geq \bigvee_{i \in H_0^E(W,\mathbf{b}) \cup H_0^{Ge}(W,\mathbf{b})} (b_i^1(W, \mathbf{b}) \wedge w_{ij}) \\ &\geq b_{i_0}^1(W, \mathbf{b}) \wedge w_{i_0 j} \geq b_j.\end{aligned} \tag{3.5}$$

In summary we have, $\bigvee_{i \in H_0^{Ge}(W,\mathbf{b}) \cup H_0^E(W,\mathbf{b})} \{x_i \wedge w_{ij}\} \geq b_j$. So

$$\begin{aligned}\bigvee_{i \in \mathrm{N}} \{x_i \wedge w_{ij}\} &= \Big(\bigvee_{i \in H_0^{Ge}(W,\mathbf{b})} \{x_i \wedge w_{ij}\}\Big) \vee \Big(\bigvee_{i \in H_0^G(W,\mathbf{b})} \{x_i \wedge w_{ij}\}\Big) \\ &\quad \vee \Big(\bigvee_{i \in H_0^E(W,\mathbf{b})} \{x_i \wedge w_{ij}\}\Big) \vee \Big(\bigvee_{i \in H_0^L(W,\mathbf{b})} \{x_i \wedge w_{ij}\}\Big) \\ &\geq \Big(\bigvee_{i \in H_0^E(W,\mathbf{b}) \cup H_0^{Ge}(W,\mathbf{b})} \{x_i \wedge w_{ij}\}\Big) \geq b_j.\end{aligned} \tag{3.6}$$

On the other hand, for $j \in$ N, let

$$l_1(q) \triangleq \bigvee_{i \in H_0^G(W,\mathbf{b})} \{w_{ij} \wedge b_i^2(W, \mathbf{b})\} = \bigvee_{i \in H_0^G(W,\mathbf{b})} \Big\{w_{ij} \wedge \Big(\bigwedge_{k \in H^G(W,\mathbf{b},i)} \{b_k\}\Big)\Big\}.$$

Since $j \in H^G(W, \mathbf{b}, i)$ it follows that $\bigwedge_{k \in H^G(W,\mathbf{b},i)} \{b_k\} \leq b_j$. And by $j \notin H^G(W, \mathbf{b}, i)$ we can conclude that $w_{ij} \leq b_j$. Hence $\bigvee_{i \in H_0^G(W,\mathbf{b})} \{x_i \wedge w_{ij}\} \leq$

$l_1(q) \le b_j$. With the same reason we have

$$b_j \ge l_2(q) \triangleq \bigvee_{i \in H_0^E(W,\mathbf{b})} \{w_{ij} \wedge 1\} \ge \bigvee_{i \in H_0^E(W,\mathbf{b})} \{w_{ij} \wedge x_i\};$$

$$b_j \ge l_3(q) \triangleq \bigvee_{i \in H_0^{Ge}(W,\mathbf{b})} \{w_{ij} \wedge b_i^2(W,\mathbf{b})\} \ge \bigvee_{i \in H_0^{Ge}(W,\mathbf{b})} \{w_{ij} \wedge x_i\}.$$

Therefore we can obtain the following face:

$$\bigvee_{i \in \mathrm{N}} \{x_i \wedge w_{ij}\} \le l_1(q) \vee l_2(q) \vee l_3(q) \vee \Big(\bigvee_{i \in H_0^L(W,\mathbf{b})} \{x_i \wedge w_{ij}\} \Big) \le b_j. \qquad (3.7)$$

Thus synthesizing (3.6) and (3.7) we get, $\bigvee_{i \in \mathrm{N}} (w_{ij} \wedge x_i) = b_j \ (j \in \mathrm{N})$. That is, by one iteration $\mathbf{x}$ converges to $\mathbf{b}$. □

To guarantee certain fault-tolerance of a fuzzy Hopfield network, it is necessary to enlarge the attractive basins of corresponding attractors. Next let us discuss this problem by adjusting suitably the connection weight matrix of (3.1).

Theorem 3.2 *In the system (3.1), suppose the fuzzy pattern* $\mathbf{b}$ *is an attractor of (3.1) when the connection weight matrix is chosen as* W_1 *and* W_2*, respectively. Moreover,* $W_1 \subset W_2$*, and* W_1 *is defined through* W_2 *by letting some nonzero elements of* W_2 *be zero. Both* $(W_1, \mathbf{b})$*,* $(W_2, \mathbf{b})$ *satisfy GE condition. Then* $A_t^0(W_2,\mathbf{b}) \subset A_t^0(W_1,\mathbf{b})$.

Proof. By the assumption and Theorem 3.1, we can show, if choose W in (3.1) respectively as W_1, W_2, then both $A_t^0(W_1,\mathbf{b})$ and $A_t^0(W_2,\mathbf{b})$ are the attractive basins of the attractor $\mathbf{b}$, correspondingly. To prove our theorem, by (3.3) it suffices to show that $\forall i \in \mathrm{N}$, the following facts hold:

$$b_i^1(W_1,\mathbf{b}) \le b_i^1(W_2,\mathbf{b}), \ \ b_i^2(W_2,\mathbf{b}) \le b_i^2(W_1,\mathbf{b}) \qquad (3.8)$$

In fact, let $W_k = (w_{ij}^k)_{n \times n}$ $(k = 1, 2)$. By the assumption, $\forall i, j \in \mathrm{N}$, either $w_{ij}^1 = 0$ or, $w_{ij}^1 = w_{ij}^2$. So for each $i \in \mathrm{N}$, $\forall j \in \mathrm{N}$, $b_j \ne 0$, $j \in H^E(W_1,\mathbf{b},i), \Longrightarrow j \in H^E(W_2,\mathbf{b},i)$. Thus, by the definitions of $b_i^1(W,\mathbf{b})$, $b_i^2(W,\mathbf{b})$ it follows that

$$b_i^1(W_1,\mathbf{b}) = \bigvee_{k \in H^E(W_1,\mathbf{b},i)} \{b_k\} \le \bigvee_{k \in H^E(W_2,\mathbf{b},i)} \{b_k\} = b_i^1(W_2,\mathbf{b}).$$

The first part of (3.8) holds. On the other hand, $W_1 \subset W_2$ implies that for each $i \in \mathrm{N}$, we get, $\forall j \in \mathrm{N}$, $b_j \ne 0$, $j \in H^G(W_1,\mathbf{b},i), \Longrightarrow j \in H^G(W_2,\mathbf{b},i)$. So

$$b_i^2(W_2,\mathbf{b}) = \bigwedge_{k \in H^G(W_2,\mathbf{b},i)} \{b_k\} \le \bigwedge_{k \in H^G(W_1,\mathbf{b},i)} \{b_k\} = b_i^2(W_1,\mathbf{b}).$$

The second part of (3.8) holds. In summary (3.8) is true. □

Using Theorem 3.2 we can get such a fact associated to the system (3.1), that is, provided (W, $\mathbf{b}$) satisfies GE condition, with the fuzzy inclusion '$\subset$' sense, the smaller the matrix W is, the larger the attractive basin of the corresponding attractor is. That provides us with a useful criterion for designing learning algorithm of W to guarantee optimal fault–tolerance.

3.1.2 Learning algorithm based on fault–tolerance

Suppose $\mathcal{B} = \{\mathbf{b}^k = (b_1^k, ..., b_n^k) | k \in \mathrm{P}\}$ is a family of fuzzy pattern. In the subsection we develop a learning algorithm for W, the connection weight matrix of (3.1) to minimize W and to ensure each fuzzy pattern in $\mathcal{B}$ is an attractor of (3.1). Moreover, we utilize some conditions to maximize the attractive basin of each attractor, so that the system can possess optimal fault–tolerance.

Definition 3.3 The fuzzy matrix $W = (w_{ij})_{n\times n}$ is called to be weakly reflexive, if W is a weakly diagonal dominant matrix, that is, $\forall i, j \in \mathrm{N}$, $w_{ij} \leq w_{jj}$.

It is easy to prove that if $W = (w_{ij})_{n\times n}$ is weakly reflexive, then $W \subset W^2$. For a given fuzzy pattern family $\mathcal{B}$, introduce the following notations:

$$H_G(\mathbf{b}^k, j) = \{i \in \mathrm{N} | b_i^k > b_j^k\}. \tag{3.9}$$

Recalling the notations in §2.1, Similarly we define, $G_{ij}^H(\mathcal{B}) = \{k \in \mathrm{P} | b_i^k > b_j^k\}$. And if $i, j \in \mathrm{N}$, we write

$$A_{ij}(\mathcal{B}) = \Big\{i_1 \in \mathrm{N} \Big| i_1 < i,\ G_{i_1 j}^H(\mathcal{B}) \neq \emptyset,\ \bigwedge_{k\in G_{i_1 j}^H(\mathcal{B})} \{b_j^k\} = \bigwedge_{k\in G_{ij}^H(\mathcal{B})} \{b_j^k\}\Big\}.$$

By the following analytic learning we can get the connection weight matrix $W_0 = (w_{ij}^0)_{n\times n}$:

$$w_{ij}^0 = \begin{cases} \bigvee_{k\in \mathrm{P}} \{b_j^k\}, & i = j; \\ \Big(\bigwedge_{k\in G_{ij}^H(\mathcal{B})} \{b_j^k\}\Big)\cdot \varpi\big(A_{ij}(\mathcal{B})\big), & i \neq j,\ i \in \bigcup_{k\in \mathrm{P}} H_G(\mathbf{b}^k, j); \\ 0, & i \neq j,\ i \notin \bigcup_{k\in \mathrm{P}} H_G(\mathbf{b}^k, j). \end{cases} \tag{3.10}$$

By (3.10) easily we have, $\forall j \in \mathrm{N}$, $k \in \mathrm{P}$, there exists uniquely $i \in \mathrm{N}$, so that $w_{ij}^0 = b_j^k$. In the following let us present some useful properties of W_0.

Theorem 3.3 *Suppose $W_0 = (w_{ij}^0)_{n\times n}$ is defined by (3.10). Then the following conclusions hold:*

(i) W_0 is weakly reflexive, consequently $W_0 \subset W_0^2$;

(ii) If in (3.1) let $W = W_0$, then each fuzzy pattern $\mathbf{b}^k$ in $\mathcal{B}$ is an attractor of (3.1).

Proof. (i) By (3.10), obviously it follows that $\forall i,\ j \in \mathrm{N},\ w_{jj}^0 = \bigvee_{k\in \mathrm{P}} \{b_j^k\} \geq w_{ij}^0$. So W_0 is weakly diagonal dominant, i.e. W_0 is weakly reflexive. Therefore, $W_0 \subset W_0^2$.

(ii) Give arbitrarily $k \in \mathrm{P}$, and $i,\ j \in \mathrm{N}$. If $i = j$, we have

$$b_i^k \wedge w_{ij}^0 = b_j^k \wedge w_{jj}^0 = b_j^k \wedge \Big(\bigvee_{k'\in \mathrm{P}} \{b_j^{k'}\}\Big) = b_j^k. \tag{3.11}$$

Assume that $i \neq j$. Then by $i \notin H_G(\mathbf{b}^k, j)$ it follows that $b_i^k \leq b_j^k$. So $b_i^k \wedge w_{ij}^0 \leq b_j^k$; If $i \in H_G(\mathbf{b}^k, j)$, then $i \in \bigcup_{k'\in \mathrm{P}} H_G(\mathbf{b}^{k'}, j)$, and when $A_{ij}(\mathcal{B}) \neq \emptyset$, we get, $b_i^k \wedge w_{ij}^0 = 0 \leq b_j^k$. Also when $A_{ij}(\mathcal{B}) = \emptyset$, considering $k \in G_{ij}^H(\mathcal{B})$ and (3.10) we have

$$w_{ij}^0 \wedge b_i^k = \Big(\bigwedge_{k'\in G_{ij}^H(\mathcal{B})} \{b_j^{k'}\}\Big) \wedge b_i^k \leq b_j^k \wedge b_i^k = b_j^k.$$

In summary the following conclusion holds:

$$i \neq j \Longrightarrow w_{ij}^0 \wedge b_i^k \leq b_j^k. \tag{3.12}$$

So (3.11) and (3.12) imply, $\bigvee_{i\in \mathrm{N}} \{w_{ij}^0 \wedge b_i^k\} = b_j^k$. Thus, $\mathbf{b}^k$ is an attractor of (3.1). □

For $i,\ j \in \mathrm{N}$, denote $E_{\mathcal{B}}(j) = \big\{k \in \mathrm{P} \big| b_j^k = \bigvee_{k'\in \mathrm{P}} \{b_j^{k'}\}\big\}$, and

$$E_{\mathcal{B}}^{\vee}(i,j) = \begin{cases} \big\{k \in \mathrm{P} \big| b_j^k = \bigwedge_{k'\in G_{ij}^H(\mathcal{B})} \{b_j^{k'}\}\big\}, & G_{ij}^H(\mathcal{B}) \neq \emptyset; \\ \emptyset, & G_{ij}^H(\mathcal{B}) = \emptyset. \end{cases} \tag{3.13}$$

Definition 3.4 In (3.1), let $W = W_0 = (w_{ij}^0)_{n\times n}$. The fuzzy pattern family $\mathcal{B} = \big\{\mathbf{b}^k = (b_1^k, ..., b_n^k) \big| k \in \mathrm{P}\big\}$ is called to be correlated, if the following conditions hold:

(i) For any $j \in \mathrm{N},\ \bigcup_{i\in \mathrm{N}} E_{\mathcal{B}}^{\vee}(i,j) \cup E_{\mathcal{B}}(j) = \mathrm{P}$;

(ii) $\forall k \in \mathrm{P},\ i \in \mathrm{N}$, if $j \in H^G(W_0, \mathbf{b}^k, i)$, and $j_0 \in H^E(W_0, \mathbf{b}^k, i)$, then $b_{j_0}^k < b_j^k$.

If the fuzzy pattern family $\mathcal{B}$ is correlated, we can establish a non-degenerate attractive basin for each fuzzy pattern $\mathbf{b}$ in $\mathcal{B}$, where $\mathbf{b}$ is an attractor of (3.1).

Theorem 3.4 *Let $\mathcal{B} = \big\{\mathbf{b}^k = (b_1^k, b_2^k, ..., b_n^k) \big| k \in \mathrm{P}\big\}$ be a fuzzy pattern family, and the following conditions hold:*

(i) $\forall j \in \mathrm{N},\ k \in \mathrm{P},\ 0 < b_j^k < 1$;

(ii) $\mathcal{B}$ is correlated.

Suppose W_0 is defined by (3.10), and in (3.1) $W = W_0$. Then $\forall k \in \mathrm{P}$, $\mathbf{b}^k$ is an attractor of (3.1), whose attractive basin $A_t^0(W_0, \mathbf{b}^k)$ is non-degenerate.

Proof. Give $k \in \mathrm{P}$. By Theorem 3.3 it follows that $W = W_0$ implies $\mathbf{b}^k$ is an attractor of (3.1). In the following let us prove, (W_0, $\mathbf{b}^k$) satisfies GE condition.

In fact, $\mathcal{B}$ is correlated, then $\forall j \in \mathrm{N}$, there is $i_1 \in \mathrm{N}$, satisfying $k \in E_{\mathcal{B}}^{\vee}(i_1, j) \cup E_{\mathcal{B}}(j)$. We write

$$i_0 = \min\{l \in \mathrm{N} | k \in E_{\mathcal{B}}^{\vee}(l, j) \cup E_{\mathcal{B}}(j)\}.$$

Then $k \in E_{\mathcal{B}}^{\vee}(i_0, j) \cup E_{\mathcal{B}}(j)$. If $k \in E_{\mathcal{B}}(j)$, we have, $\forall k' \in \mathrm{P}$, $b_j^{k'} \le b_j^k$. Therefore, $w_{jj} = b_j^k$, and $j \in H^E(W_0, \mathbf{b}^k, j)$. If $k \in E_{\mathcal{B}}^{\vee}(i_0, j)$, by (3.5), it follows that $G_{i_0 j}^H(\mathcal{B}) \neq \emptyset$. Thus, $i_0 \in \bigcup\limits_{k' \in \mathrm{P}} H_G(\mathbf{b}^{k'}, j)$. So we can show, $A(i_0, j) = \emptyset$. Otherwise, we choose $l_0 \in A_{i_0 j}(\mathcal{B})$, then $l_0 < i_0$. Moreover

$$\bigwedge_{l \in G_{i_0 j}^H(\mathcal{B})} \{b_j^l\} = \bigwedge_{l \in G_{i_0 j}^H(\mathcal{B})} \{b_j^l\} = b_j^k, \Longrightarrow k \in E_{\mathcal{B}}^{\vee}(l_0, j),$$

which contradicts the definition of i_0. So $A_{i_0 j}(\mathcal{B}) = \emptyset$. So using (3.10) and (3.13) we can imply, $b_j^k = \bigwedge\limits_{k' \in G_{i_0 j}^H(\mathcal{B})} b_j^{k'} = w_{i_0 j}$, that is, $j \in H^E(W_0, \mathbf{b}^k, i_0)$. Thus, for $j \in \mathrm{N}$, there is $i \in \mathrm{N}$, so that $j \in H^E(W_0, \mathbf{b}^k, i)$. On the other hand, $\forall j_1 \in H^E(W_0, \mathbf{b}^k, i)$, $j_2 \in H^G(W_0, \mathbf{b}^k, i)$, we get, $w_{ij_1}^0 = b_{j_1}^k$, $w_{ij_2}^0 > b_{j_2}^k$. Since $\mathcal{B}$ is correlated, it follows that $b_{j_2}^k > b_{j_1}^k$. Therefore, (W_0, $\mathbf{b}^k$) satisfies GE condition. By Theorem 3.1, for each $\mathbf{x} = (x_1, ..., x_n) \in A_t^0(W_0, \mathbf{b}^k)$, $\mathbf{x}$ converges to $\mathbf{b}^k$ with one iteration. So $A_t^0(W_0, \mathbf{b}^k)$ is the attractive basin of $\mathbf{b}^k$. By the assumption and (3.10), obviously $A_t^0(W_0, \mathbf{b}^k)$ is non-degenerate. □

Let us now present another useful property of W_0, which shows the minimality of W_0.

Theorem 3.5 *Let* $\mathcal{B} = \{\mathbf{b}^k = (b_1^k, b_2^k, ..., b_n^k) | k \in \mathrm{P}\}$ *be a correlated fuzzy pattern family. And suppose* W_0 *is defined by (3.10). Then* W_0 *is a minimum matrix among the fuzzy matrices W's with the following conditions:*

(i) $\forall k \in \mathrm{P}$, $(W, \mathbf{b}^k)$ *satisfies GE condition;*

(ii) For each $k \in \mathrm{P}$, $\mathbf{b}^k$ *is an attractor of W, i.e.* $\mathbf{b}^k \circ W = \mathbf{b}^k$;

(iii) W is weakly reflexive.

Proof. By Theorem 3.4, W_0 defined by (3.10) satisfies the conditions (i) (ii) and (iii). For arbitrary $W = (w_{ij})_{n \times n}$ with the assumptions (i) (ii) (iii). At first, by (ii) (iii) it is trivial to show $w_{jj} = w_{jj}^0$ for each $j \in \mathrm{N}$ if $W \subset W_0$. Next let us use reduction to absurdity to show the other conclusions. Assume the result is false, then we may assume $W \subset W_0$, and there exist i_0, $j_0 \in \mathrm{N}$, satisfying $w_{i_0 j_0} < w_{i_0 j_0}^0$. Then $i_0 \neq j_0$. It is no harm to assume

$$\{i \in \mathrm{N} | w_{ij_0} < w_{ij_0}^0\} = \{i_0, i_1, ..., i_p\} : 1 \le i_0 < i_1 < \cdots < i_p \le n. \qquad (3.14)$$

By the definition (3.10) of W_0, $G^H_{i_0j_0}(\mathcal{B}) \neq \emptyset$, and there is $k_0 \in \mathrm{P}$, so that $b^{k_0}_{j_0} = w^0_{i_0j_0}$. So $w_{i_0j_0} < b^{k_0}_{j_0}$. Since $(W, \mathbf{b}^{k_0})$ satisfies GE condition by the assumption, there is a $i' \in \mathrm{N}$, so that $j_0 \in H^E(W, \mathbf{b}^{k_0}, i')$. Obviously $i' \neq i_0$. If $i_1 < i'$, by (3.14) we have, $w_{i'j_0} = w^0_{i'j_0} = b^{k_0}_{j_0}$. Thus $G^H_{i'j_0}(\mathcal{B}) \neq \emptyset$. By (3.10) it follows that $w^0_{i_0j_0} = 0$, which is a contradiction. So $i' > i_0$. Then by the assumption of W, we get, $w_{i'j_0} \leq w^0_{i'j_0}$. Next let us show $w_{i'j_0} < w^0_{i'j_0}$. Otherwise $b^{k_0}_{j_0} = w_{i'j_0} = w^0_{i'j_0}$, which implies $G^H_{i'j_0}(\mathcal{B}) \neq \emptyset$, and then $w^0_{i_0j_0} = 0$ by (3.10), which is also a contradiction. So it is no harm to assume $i' = i_1$. Similarly we have, there is $k_1 \in G^H_{i_1j_0}(\mathcal{B}) : \ w^0_{i_1j_0} = b^{k_1}_{j_0}$. With the same reason we can imply, there exists $k_2 \in G^H_{i_2j_0}(\mathcal{B})$, so that

$$b^{k_1}_{j_0} = w_{i_2j_0} < w^0_{i_2j_0} = b^{k_2}_{j_0}.$$

Continuing the procedure until the p−th step, we get a $k_p \in G^H_{i_pj_0}(\mathcal{B})$, so that $w_{i_pj_0} < w^0_{i_pj_0} = b^{k_p}_{j_0}$. Since $(W, \mathbf{b}^{k_p})$ satisfies GE condition, there is $i_* \in \mathrm{N}$: $j_0 \in H^E(W, \mathbf{b}^{k_p}, i_*)$. Similarly we can show, $i_* > i_p$, and $w_{i_*j_0} < w^0_{i_*j_0}$, which contradicts (3.14). Thus, W_0 is a minimum matrix with the given conditions. □

By Theorem 3.2, Theorem 3.4 and Theorem 3.5, for a given fuzzy pattern family $\mathcal{B}$ if we train the system (3.1) so that its connection weight matrix W is determined as W_0, then each fuzzy pattern in $\mathcal{B}$ is an attractor of (3.1), moreover we can establish the largest attractive basins of the system, and consequently fault–tolerance of the system is optimal.

3.1.3 Simulation example

To illustrate above results we in the subsection employ a simulation example to establish the attractors, attractive basins of the attractors related to the system (3.1). We may show that this dynamical system possesses and optimal fault–tolerance if we choose the connection weight matrix rationally. To this end let $\mathrm{N} = \{1, 2, 3, 4, 5, 6\}$, $\mathrm{P} = \{1, 2, 3, 4, 5\}$. By Table 3.1 we give a fuzzy pattern family $\mathcal{B}$, which are asked to be the attractors of (3.1).

Table 3.1 Fuzzy pattern family

k	$\mathbf{b}^k$
1	(0.6 0.5 0.6 0.8 0.3 0.6)
2	(0.5 0.7 0.7 0.8 0.7 0.6)
3	(0.6 0.4 0.7 0.3 0.7 0.4)
4	(0.5 0.7 0.4 0.3 0.7 0.6)
5	(0.4 0.7 0.7 0.8 0.3 0.5)

Using (3.13) we can calculate the set $E_{\mathcal{B}}^{\vee}(i,j)$ as shown in Table 3.2.

Table 3.2 The set $E_{\mathcal{B}}^{\vee}(i,j)$

	1	2	3	4	5	6
1	∅	{3}	{4}	{3,4}	{1,5}	{3}
2	{5}	∅	{4}	{3,4}	{1,5}	{5}
3	{5}	{3}	∅	{3,4}	{1,5}	{3}
4	{5}	{1}	{1}	∅	{1,5}	{5}
5	{2,4}	{3}	{4}	{3,4}	∅	{3}
6	{5}	{1}	{4}	{3,4}	{1,5}	∅

Thus we can obtain the following sets:

$$K_1 = \{1,\ 3\},\ K_2 = \{2,\ 4,\ 5\},\ K_3 = \{2,\ 3,\ 5\},$$
$$K_4 = \{1,\ 2,\ 5\},\ K_5 = \{2,\ 3,\ 4\},\ K_6 = \{1,\ 2,\ 4\}.$$

Therefore

$$\forall j \in \mathrm{N},\ \Big(\bigcup_{i\in\mathrm{N}} E_{\mathcal{B}}^{\vee}(i,j)\Big)\cup E_{\mathcal{B}}(j) = \mathrm{P}.$$

So by (3.10) it follows that

$$W_0 = \begin{pmatrix} 0.6 & 0.4 & 0.4 & 0.3 & 0.3 & 0.4 \\ 0.4 & 0.7 & 0 & 0 & 0 & 0.5 \\ 0 & 0 & 0.7 & 0 & 0 & 0 \\ 0 & 0.5 & 0.6 & 0.8 & 0 & 0 \\ 0.5 & 0 & 0 & 0 & 0.7 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.6 \end{pmatrix}$$

We can show, $\mathcal{B}$ is correlated. Hence by Theorem 3.1, if in (3.1) let $W = W_0$, then all $\mathbf{b}^1$, $\mathbf{b}^2, ..., \mathbf{b}^5$ are attractors of the system. To calculate the attractive basin of $\mathbf{b}^k$, at first we compute $b_i^1(W_0, \mathbf{b}^k)$ $(k \in \mathrm{P},\ i \in \mathrm{N})$.

Table 3.3 The value of $b_i^1(W_0, \mathbf{b}^k)$

	1	2	3	4	5
1	0.6	0	0.6	0.4	0.3
2	0	0.7	0	0.7	0.7
3	0	0.7	0.7	0	0.7
4	0.8	0.8	0	0	0.8
5	0	0.7	0.7	0.7	0
6	0.6	0.6	0	0.6	0

Similarly we can calculate $b_i^2(W_0, \mathbf{b}^k)$ for $k \in \mathrm{P}$, $i \in \mathrm{N}$ as follows:

Table 3.4 The value of $b_i^2(W_0, \mathbf{b}^k)$

	1	2	3	4	5
1	1	0.5	1	0.5	0.4
2	0.5	1	0.4	1	1
3	0.6	1	1	0.4	1
4	1	1	0.3	0.3	1
5	0.3	1	1	1	0.3
6	1	1	0.4	1	0.5

Table 3.5 The sets $H_0^{Ge}(W, \mathbf{b}^k)$, $H_0^E(W, \mathbf{b}^k)$ and $H_0^G(W, \mathbf{b}^k)$

No.	$H_0^{Ge}(W, \mathbf{b}^k)$	$H_0^E(W, \mathbf{b}^k)$	$H_0^G(W, \mathbf{b}^k)$
1	$\emptyset$	{1,4,6}	{2,3,5}
2	$\emptyset$	{2,3,4,5,6}	{1}
3	$\emptyset$	{1,3,5}	{2,4,6}
4	{1}	{2,5,6}	{3,4}
5	{1}	{2,3,4}	{5,6}

Therefore we can conclude that

$$A_t^0(W_0, \mathbf{b}^1) = [0.6, 1] \times [0, 0.5] \times [0, 0.6] \times [0.8, 1] \times [0, 0.3] \times [0.6, 1];$$
$$A_t^0(W_0, \mathbf{b}^2) = [0, 0.5] \times [0.7, 1] \times [0.7, 1] \times [0.8, 1] \times [0.7, 1] \times [0.6, 1];$$
$$A_t^0(W_0, \mathbf{b}^3) = [0.6, 1] \times [0, 0.4] \times [0.7, 1] \times [0, 0.3] \times [0.7, 1] \times [0, 0.4];$$
$$A_t^0(W_0, \mathbf{b}^4) = [0.4, 0.5] \times [0.7, 1] \times [0, 0.4] \times [0, 0.3] \times [0.7, 1] \times [0.6, 1];$$
$$A_t^0(W_0, \mathbf{b}^5) = [0.3, 0.4] \times [0.7, 1] \times [0.7, 1] \times [0.8, 1] \times [0, 0.3] \times [0, 0.5].$$

Above subsets of $[0, 1]^5$ constitute the maximum attractive basins of $\mathbf{b}^1, ..., \mathbf{b}^5$, respectively.

§3.2 Fuzzy Hopfield network with threshold

From the preceding section, the storage capacity of fuzzy patterns and fault–tolerance are two important functions of a fuzzy Hopfield network. So one main objective for analyzing such FNN's is to improve fuzzy Hopfield networks in their storage capacity and fault-tolerance. To this end, we in the section introduce a threshold c_i to the node i in (3.1) for $i \in \mathrm{N}$. Also $w_{ij} \in [0, 1]$

means the connection weight from the node i to the node j. When a system receives a input fuzzy pattern as $\mathbf{x}(0) = \big(x_1(0), ..., x_n(0)\big) \in [0,\ 1]^n$, the system evolutes with the following scheme:

$$x_j(t) = \bigvee_{i=1}^{n} \big((x_i(t-1) \vee c_i) \wedge w_{ij}\big)\ (j = 1, ..., n), \tag{3.15}$$

where $t = 1,\ 2, ...$ means the iteration or evolution step. $\mathbf{x}(t) \triangleq \big(x_1(t), ..., x_n(t)\big)$ is the system state of (3.15) at the step t. If the state of (3.15) keeps still after finite time steps, the network has finished the associative processes. The ultimate state $\mathbf{b}$ of the system is called an equilibrium point or, a stable state of (3.15). Therefore, the feedback FAM's may be characterized as the dynamic processes that the recurrent FNN's process to their stable states (attractors) from the initial states. In the following let us present some useful properties about the attractors of (3.15).

3.2.1 Attractor and stability

For the given fuzzy patterns $\mathbf{a} = (a_1, ..., a_n)$, $\mathbf{b} = (b_1, ..., b_n) \in [0, 1]^n$, and the fuzzy matrix $W = (w_{ij})_{n\times n}$, we introduce the notations:

$$\mathbf{a} \vee \mathbf{b} = \big(a_1 \vee b_1, ..., a_n \vee b_n\big),\ W^2 = W \circ W,\ W^k = W^{k-1} \circ W,$$

where $k = 2, 3, ...$ Easily we can show

$$(\mathbf{a} \vee \mathbf{b}) \circ W = (\mathbf{a} \circ W) \vee (\mathbf{b} \circ W),\ W^k = W^{k-l} \circ W^l \tag{3.16}$$

where $l \leq k$, and $k,\ l \in \mathbb{N}$. For $t = 1, 2, ...$, if $\mathbf{x}(t) = \big(x_1(t), ..., x_n(t)\big)$, $\mathbf{c} = (c_1, ..., c_n)$, Rewrite the system (3.15) as

$$\mathbf{x}(t) = \big(\mathbf{x}(t-1) \vee \mathbf{c}\big) \circ W. \tag{3.17}$$

Definition 3.5 The fuzzy pattern $\mathbf{b} = (b_1, ..., b_n)$ is called a stable state of (3.17), if $\mathbf{b} = (\mathbf{b} \vee \mathbf{c}) \circ W$, also $\mathbf{b}$ is called an equilibrium point of the system. The attractive basin of the stable state $\mathbf{b}$ means the collection of fuzzy patterns $A_f(\mathbf{b}) \subset [0,\ 1]^n$, so that $\forall \mathbf{x} \in A_f(\mathbf{b})$, the system converges to $\mathbf{b}$ if the initial pattern is $\mathbf{x}$. The vector $\mathbf{c} \in [0,\ 1]^n$ is called a threshold vector.

In [41, 42], Sanchez E. calls the fuzzy set $\mathbf{b}$ that satisfies $\mathbf{b} = \mathbf{b} \circ W$ for the given fuzzy matrix W a eigen-fuzzy set. Sanchez determines a greatest eigen-fuzzy set and minimum ones. One of our aims in the section is to design a learning algorithm for W, so that the given fuzzy patterns (sets) $\mathbf{b}^1, ..., \mathbf{b}^p$ satisfy the relations that $\mathbf{b}^i = \mathbf{b}^i \circ W$ for $i = 1, ..., p$.

It is well known that Hopfield networks whose connection matrices are symmetric and have zero diagonal elements are stable if they evolute with parallel fashions [19, 22, 52]. The following theorem shows the similar conclusions for the system (3.17).

Theorem 3.6 *The following conclusions related to the system (3.17) hold:*
(i) There exists $s \in \mathbb{N}$*, so that the system converges to a limit cycle whose length does not exceed* s*, that is, there are fuzzy patterns* $\mathbf{x}_1, ..., \mathbf{x}_s$*, satisfying*

$$\mathbf{x}_2 = (\mathbf{x}_1 \vee \mathbf{c}) \circ W, ..., \mathbf{x}_{k+1} = (\mathbf{x}_k \vee \mathbf{c}) \circ W, ..., \mathbf{x}_1 = (\mathbf{x}_s \vee \mathbf{c}) \circ W. \qquad (3.18)$$

(ii) If $W \subset W^2$*, then the system converges to its equilibrium point within finite iteration steps.*

Proof. (i) By the definition of the composition operation '$\circ$', there are at most finite different members in the matrix sequence $\{W^k | k = 1, 2, ...\}$. And consequently there exists a $q \in \mathbb{N}$, so that $W^{q+k} = W^k$ for $k = 1, 2, ...$ For a given initial fuzzy pattern $\mathbf{x}$, let $\mathbf{x}(0) = \mathbf{x}$. Considering (3.17), we get by the induction method that

$$\begin{aligned} \mathbf{x}(q+q+2) &= \big(\mathbf{x}(q+q+1) \vee \mathbf{c}\big) \circ W \\ &= \big((\mathbf{x}(0) \vee \mathbf{c}) \circ W^{q+q+2}\big) \vee \big(\mathbf{c} \circ W^{q+q+1}\big) \vee \big(\mathbf{c} \circ W^{q+q}\big) \vee \cdots \vee (\mathbf{c} \circ W) \\ &= \big((\mathbf{x}(0) \vee \mathbf{c}) \circ W^{q+2}\big) \vee \big(\mathbf{c} \circ W^{q+1}\big) \vee \cdots \vee (\mathbf{c} \circ W) = \mathbf{x}(q+2). \end{aligned}$$

That is, $\mathbf{x}(2q+2) = \mathbf{x}(q+2)$. Choose $s = q$, and let $\mathbf{x}_1 = \mathbf{x}(s+2)$, $\mathbf{x}_2 = \mathbf{x}(s+3), ..., \mathbf{x}_s = \mathbf{x}(2s+1)$. Thus, (3.18) holds.

(ii) The fact $W \subset W^2$ implies $W^k \subset W^{k+1}$ for $k = 1, 2, ...$ Therefore, the fuzzy matrix sequence $\{W^k | k = 1, 2, ...\}$ which contains only finite different members is monotonically increasing. Thus, there exists $l \in \mathbb{N}$, satisfying $W^{l+1} = W^l$. Let us now prove, $\mathbf{b} = \mathbf{x}(l)$ is a stable state of the system (3.17). By the induction method we get

$$\mathbf{x}(l+1) = \big((\mathbf{x}(0) \vee \mathbf{c}) \circ W^{l+1}\big) \vee \big(\mathbf{c} \circ W^l\big) \vee \cdots \vee (\mathbf{c} \circ W).$$

And $\forall k \leq l,\ W^k \subset W^{l+1},\ \mathbf{c} \subset \mathbf{x}(0) \vee \mathbf{c}$, So

$$\mathbf{c} \circ W^k \subset (\mathbf{x}(0) \vee \mathbf{c}) \circ W^{l+1} \Longrightarrow \mathbf{x}(l+1) = (\mathbf{x}(0) \vee \mathbf{c}) \circ W^{l+1}.$$

With the same reason, $\mathbf{x}(l) = (\mathbf{x}(0) \vee \mathbf{c}) \circ W^l = (\mathbf{x}(0) \vee \mathbf{c}) \circ W^{l+1}$, Hence $\mathbf{b} = \mathbf{x}(l) = \mathbf{x}(l+1) = \big(\mathbf{x}(l) \vee \mathbf{c}\big) \circ W$, i.e. the system converges to the stable state $\mathbf{b}$ at $l-$th iteration step. □

For given fuzzy patterns $\mathbf{a} = (a_1, ..., a_n)$, $\mathbf{b} = (b_1, ..., b_n) \in [0, 1]^n$, we denote by $H(\mathbf{a}, \mathbf{b})$ the Hamming metric between $\mathbf{a}, \mathbf{b}$, that is, $H(\mathbf{a}, \mathbf{b}) = \sum_{i=1}^{n} |a_i - b_i|$. To study the stability of (3.17) we at first present a useful lemma.

Lemma 3.1 *Let* $a, b, c \in [0, 1]$*, and* $p \in \mathbb{N}$*,* $a_i, b_i \in [0, 1]$ *for* $i = 1, ..., p$*. Let* $W = (w_{ij})_{n \times m}$ *be a fuzzy matrix. Then*
(i) $|(a \wedge c) - (b \wedge c)| \leq |a - b|$;
(ii) $|a_i - b_i| < h \ (i = 1, ..., p), \Longrightarrow \Big|\bigvee_{i=1}^{p} a_i - \bigvee_{i=1}^{p} b_i\Big| < h, \ \Big|\bigwedge_{i=1}^{p} a_i - \bigwedge_{i=1}^{p} b_i\Big| < h;$

(iii) $W \circ W^T \subset (W \circ W^T)^2,\ W^T \circ W \subset (W^T \circ W)^2$.

Proof. (i) Since $a \wedge c = \big(a + c - |a - c|\big)/2,\ b \wedge c = \big(b + c - |b - c|\big)/2$, it follows that

$$\begin{aligned}\big|(a \wedge c) - (b \wedge c)\big| &= \frac{1}{2}\big|\big(a - b + |b - c| - |a - c|\big)\big| \\ &\le \frac{1}{2}\big(|a - b| + \big|\big(|b - c| - |a - c|\big)\big|\big) \le \frac{1}{2}\big(|a - b| + |a - b|\big) = |a - b|.\end{aligned}$$

(ii) By the induction method we can show that

$$\Big|\bigvee_{i=1}^{p} a_i - \bigvee_{i=1}^{p} b_i\Big| < h. \tag{3.19}$$

In fact when $p = 1$, obviously (3.19) holds. Assume that when $p = k$, (3.19) is true. If let $a' = \bigvee_{i=1}^{k} a_i,\ b' = \bigvee_{i=1}^{k} b_i$, then $|a' - b'| < h$. So when $p = k + 1$, we have, $\bigvee_{i=1}^{k+1} a_i = a' \vee a_{k+1},\ \bigvee_{i=1}^{k+1} b_i = b' \vee b_{k+1}$. Therefore

$$\begin{aligned}\Big|\bigvee_{i=1}^{k+1} a_i - \bigvee_{i=1}^{k+1} b_i\Big| &= |a' \vee a_{k+1} - b' \vee b_{k+1}| \\ &= \frac{1}{2}\,\Big|\big((a' - b') + (a_{k+1} - b_{k+1}) + |a' - a_{k+1}| - |b' - b_{k+1}|\big)\Big|.\end{aligned}$$

Next we show (3.19) in the following four cases, respectively:

$$\begin{aligned}&1.\ a' \ge a_{k+1},\ b' \ge b_{k+1}; \quad 2.\ a' < a_{k+1},\ b' \ge b_{k+1};\\ &3.\ a' \ge a_{k+1},\ b' < b_{k+1}; \quad 4.\ a' < a_{k+1},\ b' < b_{k+1}.\end{aligned}$$

To the case 1, $|a' \vee a_{k+1} - b' \vee b_{k+1}| = |a' - b'| < h$, which implies (3.19); As for the case 2, $|a' \vee a_{k+1} - b' \vee b_{k+1}| = |a_{k+1} - b'|$, then

$$-h < a' - b' \le a_{k+1} - b' \le a_{k+1} - b_{k+1} < h,$$

So $|a' \vee a_{k+1} - b' \vee b_{k+1}| < h$, which also ensure (3.19) to hold. To the other two cases, we can show, $|a' \vee a_{k+1} - b' \vee b_{k+1}| < h$. Thus, when $p = k + 1$, (3.19) holds. By the induction principle we can obtain (3.19).

(iii) It suffices to show the first parts, since the second can be proved, similarly. Let $W \circ W^T = (\overline{w}_{ij})_{n\times n},\ (W \circ W^T)^2 = (\overline{w}^2_{ij})_{n\times n}$. Then $\forall i, j \in \mathrm{N}$, we get

$$\begin{aligned}\overline{w}^2_{ij} &= \bigvee_{k\in\mathrm{N}} (\overline{w}_{ik} \wedge \overline{w}_{kj}) = \bigvee_{k\in\mathrm{N}} \Big(\Big(\bigvee_{p\in\mathrm{M}} (w_{ip} \wedge w_{kp})\Big) \wedge \Big(\bigvee_{p\in\mathrm{M}} (w_{kp} \wedge w_{jp})\Big)\Big) \\ &\ge \bigvee_{k\in\mathrm{N}} \Big(\bigvee_{p\in\mathrm{M}} (w_{ip} \wedge w_{kp} \wedge w_{kp} \wedge w_{jp}\Big) \ge \bigvee_{p\in\mathrm{M}} \Big(\bigvee_{k\in\mathrm{N}} (w_{ip} \wedge w_{kp} \wedge w_{jp})\Big) \\ &\ge \bigvee_{p\in\mathrm{M}} \big(w_{ip} \wedge w_{jp}\big) \ge \overline{w}_{ij}.\end{aligned}$$

Thus, $W \circ W^T \subset (W \circ W^T)^2$, The lemma is proved. □

Definition 3.6 Suppose the fuzzy pattern **b** is an equilibrium point of (3.17). **b** is called to be Lyapunov stable, if $\forall \varepsilon > 0$, there is $\delta > 0$, so that for every fuzzy pattern **x** satisfying $H(\mathbf{x}, \mathbf{b}) < \delta$, $H(\mathbf{x}(k), \mathbf{b}) < \varepsilon$ holds for $k = 1, 2, ...$, where $\mathbf{x}(0) = \mathbf{x}$, $\mathbf{x}(k)$ means the k-th iteration state of the system. We call a Lyapunov stable state an attractors of the system. If $\forall \mathbf{x} \in [0, 1]^n$, take **x** as the initial fuzzy pattern, the system (3.17) converges to an attractor within finite iteration steps. Then we call (3.17) to be uniformly stable.

Theorem 3.7 *In the system (3.17), suppose the connection weight matrix* W *satisfies* $W \subset W^2$. *Then*

(i) (3.17) is uniformly stable;

(ii) If the fuzzy pattern **b** *is an equilibrium point, then* **b** *is Lyapunov stable, consequently* **b** *is an attractor of the system.*

Proof. At first we prove (ii). Since $W \subset W^2$, it follows that $W^k \subset W^{k+1}$ $(k = 1, 2, ...)$. Using the induction method we get

$$\mathbf{x}(k) = \big((\mathbf{x}(0) \vee \mathbf{c}) \circ W^k\big) \vee \big(\mathbf{c} \circ W^{k-1}\big) \vee \cdots \vee (\mathbf{c} \circ W) \ (k = 1, 2, ...).$$

For $l \le k$, we can conclude that the following facts hold: $W^l \subset W^k$, $\mathbf{c} \circ W^l \subset (\mathbf{x}(0) \vee \mathbf{c}) \circ W^k$. So $\mathbf{x}(k) = (\mathbf{x}(0) \vee \mathbf{c}) \circ W^k$. We denote $W^k = (w_{ij}^k)_{n \times n}$ $(k \ge 1)$, $\mathbf{x}(k) = (x_1(k), ..., x_n(k))$ $(k \ge 0)$, $\mathbf{b} = (b_1, ..., b_n)$. Since **b** is an equilibrium point, $\forall j \in \{1, ..., n\}$, we have

$$x_j(k) = \bigvee_{i=1}^{n} \big((x_i(0) \vee c_i) \wedge w_{ij}^k\big), b_j = \bigvee_{i=1}^{n} \big((b_i \vee c_i) \wedge w_{ij}^k\big).$$

So $\forall \varepsilon > 0$, choose $\delta = \varepsilon/n$. Arbitrarily giving a fuzzy pattern $\mathbf{x}: H(\mathbf{x}, \mathbf{b}) < \delta$, we let the initial fuzzy pattern $\mathbf{x}(0) = (x_1(0), ..., x_n(0)) = \mathbf{x}$. Then $\forall i \in \{1, ..., n\}$, $|x_i(0) - b_i| < \varepsilon/n$. Using Lemma 3.1 we can conclude

$$\Big|(x_i(0) \vee c_i) \wedge w_{ij}^k - (b_i \vee c_i) \wedge w_{ij}^k\Big| \le |x_i(0) - b_i| < \varepsilon/n.$$

Therefore Lemma 3.2 may imply the following fact holds:

$$\Big|\bigvee_{i=1}^{n} \big((x_i(0) \vee c_i) \wedge w_{ij}^k\big) - \bigvee_{i=1}^{n} \big((b_i \vee c_i) \wedge w_{ij}^k\big)\Big| < \varepsilon/n,$$

that is, $|x_j(k) - b_j| < \varepsilon/n$. Thus, $H(\mathbf{x}(k), \mathbf{b}) = \sum_{i=1}^{n} |x_j(k) - b_j| < \varepsilon$. So the equilibrium point **b** is Lyapunov stable, and therefore **b** is an attractor.

(i) can be proved, directly by above proof for (ii) and Theorem 3.7. □

As stable states of a dynamic neural system, attractors play important roles in the applications of FNN's. In the following let us present the systematic discussion to attractors and their attractive basins. Give the fuzzy pattern $\mathbf{b} = (b_1, ..., b_n)$, and $j = 1, ..., n$. Denote

$$H^t(W, j) = \{i \in \mathrm{N} | w_{ij} \wedge (b_i \vee c_i) \geq b_j\}.$$

We at first build an equivalent condition for $\mathbf{b}$ being an attractor of (3.17).

Theorem 3.8 *Suppose* $\mathbf{b} = (b_1, ..., b_n)$ *is a fuzzy pattern. In (3.17),* $W \subset W^2$. *Then* $\mathbf{b}$ *is an attractor of (3.17) if and only if the following conditions hold:*

(i) $\forall i, j \in \{1, ..., n\}$, $w_{ij} \wedge (b_i \vee c_i) \leq b_j$;

(ii) $\forall j \in \{1, ..., n\}$, $H^t(W, j) \neq \emptyset$.

Moreover, if $\mathbf{b}$ *is an attractor of (3.17), and denote*

$$A_t(W, \mathbf{b}, \mathbf{c}) = \Big\{(x_1, ..., x_n) \,\Big|\, \forall i = 1, ..., n, \ \bigvee_{j=1}^{n} \big(b_i \wedge w_{ij}\big) \leq x_i \leq b_i \vee c_i\Big\}. \quad (3.20)$$

Then $\forall \mathbf{x} \in A_t(W, \mathbf{b}, \mathbf{c})$, $\mathbf{x}$ *converges to* $\mathbf{b}$ *with one iteration.*

Proof. If $\mathbf{b} = (b_1, ..., b_n)$ is an attractor of (3.17), then $\forall j \in \mathrm{N}$, we have

$$b_j = \bigvee_{i \in \mathrm{N}} \big(w_{ij} \wedge (b_i \vee c_i)\big). \quad (3.21)$$

So obviously (i) holds. By (3.21), using reduction to absurdity we can show that (ii) is true.

Conversely, $\forall j \in \mathrm{N}$, (i) may imply, $\bigvee_{i \in \mathrm{N}} \big(w_{ij} \wedge (b_i \vee c_i)\big) \leq b_j$; And (ii) may imply, $\bigvee_{i \in \mathrm{N}} \big(w_{ij} \wedge (b_i \vee c_i)\big) \geq b_j$. Thus, (3.21) holds, i.e. $\mathbf{b}$ is an equilibrium point of (3.17). By the assumption and Theorem 3.7, $\mathbf{b}$ is an attractor of (3.17).

Next let us prove the last part of the theorem. Let $b_i^1 = \bigvee_{j=1}^{n} \big(b_i \wedge w_{ij}\big)$, $b_i^2 = b_i \vee c_i$ $(i = 1, ..., n)$. Since $\mathbf{b}$ is an attractor of (3.17), easily we can show, both $\mathbf{b}_1 = (b_1^1, ..., b_n^1)$, $\mathbf{b}_2 = (b_1^2, ..., b_n^2)$ can converge to $\mathbf{b}$ with one iteration. Considering $\forall \mathbf{x} \in A_t(W, \mathbf{b}, \mathbf{c})$, we get, $\mathbf{b}_1 \subset \mathbf{x} \subset \mathbf{b}_2$. Therefore

$$\mathbf{b} = (\mathbf{b}_1 \vee \mathbf{c}) \circ W \subset (\mathbf{x} \vee \mathbf{c}) \circ W \subset (\mathbf{b}_2 \vee \mathbf{c}) \circ W = \mathbf{b},$$

i.e. $(\mathbf{x} \vee \mathbf{c}) \circ W = \mathbf{b}$, $\mathbf{x}$ converges to $\mathbf{b}$ with one iteration. □

3.2.2 Analysis of fault–tolerance

The fault-tolerance of FNN's characterizes the abilities that the systems recall the fuzzy patterns stored if the noisy or imperfect inputs are presented to the systems. And the attractive basin of the attractors of the systems characterizes the fault-tolerance of the systems. Let us now derive the nontrivial attractive basins of the attractors of the system (3.17). And the system can be ensured to possess good fault–tolerance.

Lemma 3.2 *Let* $\mathbf{b} = (b_1, ..., b_n)$ *be an attractor of (3.17), and* $W = (w_{ij})_{n\times n}$ *be the connection weight matrix related. Then* $\forall i \in \mathrm{N},\ c_i \le b_i^2(W, \mathbf{b})$.

Proof. By reduction to absurdity, if assume the conclusion is false, then there is $i' \in \mathrm{N}$, so that $c_{i'} > b_{i'}^2(W, \mathbf{b})$. Using (3.17) and considering $c_{i'} \in [0,\ 1]$, we get, $H^G(W, \mathbf{b}, i') \ne \emptyset$. Thus, there exists $k \in H^G(W, \mathbf{b}, i')$, satisfying $c_{i'} > b_k$, moreover $w_{i'k} > b_k$. So we can conclude that

$$\bigvee_{i\in \mathrm{N}} ((b_i \vee c_i) \wedge w_{ik}) \ge (b_{i'} \vee c_{i'}) \wedge w_{i'k} \ge c_{i'} \wedge w_{i'k} > b_k,$$

which contradicts the fact that $\mathbf{b}$ is an attractor of (3.17). □

For $i \in \mathrm{N}$, we introduce the following notations:

$$d_i^1(W, \mathbf{b}) = \begin{cases} 0, & c_i \ge b_i^1(W, \mathbf{b}); \\ b_i^1(W, \mathbf{b}), & c_i < b_i^1(W, \mathbf{b}). \end{cases} \quad d_i^2(W, \mathbf{b}) = b_i^2(W, \mathbf{b}). \tag{3.22}$$

Easily we can show the following inequalities:

$$d_i^1(W, \mathbf{b}) \le b_i^1(W, \mathbf{b}) \le d_i^1(W, \mathbf{b}) \vee c_i \ (i \in \mathrm{N}). \tag{3.23}$$

If $i \in \mathrm{N},\ \forall j_1 \in H^E(W, \mathbf{b}, i),\ j_2 \in H^G(W, \mathbf{b}, i),\ b_{j_1} < b_{j_2}$, then we denote for a given threshold vector $\mathbf{c} = (c_1, ..., c_n)$:

$$\begin{aligned} A_t^s(W, \mathbf{b}, \mathbf{c}) = \Big\{ &(x_1, ..., x_n) \in [0,\ 1]^n \Big| x_i \in \big[0,\ d_i^2(W, \mathbf{b})\big] \ (i \in H_0^G(W, \mathbf{b})), \\ &x_i \in \big[d_i^1(W, \mathbf{b}),\ d_i^2(W, \mathbf{b})\big] \ (i \in H_0^{Ge}(W, \mathbf{b})), \\ &x_i \in \big[d_i^1(W, \mathbf{b}),\ 1\big] \ (i \in H_0^E(W, \mathbf{b})),\ x_i \in [0,\ 1] \ (i \in H_0^L(W, \mathbf{b}))\Big\}. \end{aligned} \tag{3.24}$$

Lemma 3.3 *Let* $W = (w_{ij})_{n\times n}$ *be a connection weight matrix, and* $\mathbf{b} = (b_1, ..., b_n)$ *be an attractor of (3.17).* $\forall j \in \mathrm{N}$, *let*

$$l_1(q) = \bigvee_{i\in H_0^{Ge}(W,\mathbf{b})\cup H_0^G(W,\mathbf{b})} \{w_{ij} \wedge (d_i^2(W, \mathbf{b}) \vee c_i)\},\ \ l_2(q) = \bigvee_{i\in H_0^E(W,\mathbf{b})} \{w_{ij} \wedge 1\}.$$

Then $l_1(q) \vee l_2(q) \le b_j$.

Proof. If $i \in H_0^E(W, \mathbf{b})$ and $j \in H^E(W, \mathbf{b}, i)$, then $w_{ij} = b_j$; If $i \in H_0^E(W, \mathbf{b})$ but $j \notin H^E(W, \mathbf{b}, i)$, then by $H^G(W, \mathbf{b}, i) = \emptyset$, we imply $w_{ij} < b_j$. Therefore

$$\begin{aligned} l_2(q) &= \bigvee_{i\in H_0^E(W,\mathbf{b})} \{w_{ij} \wedge 1\} \\ &= \Big(\bigvee_{i|i\in H_0^E(W,\mathbf{b}),j\in H^E(W,\mathbf{b},i)} \{w_{ij}\}\Big) \vee \Big(\bigvee_{i|i\in H_0^E(W,\mathbf{b}),j\notin H^E(W,\mathbf{b},i)} \{w_{ij}\}\Big) \le b_j. \end{aligned} \tag{3.25}$$

For $i \in \mathrm{N}$, by $j \in H^G(W,\mathbf{b},i)$ it follows that $\bigwedge_{k\in H^G(W,\mathbf{b},i)} \{b_k\} \le b_j$. Since $\mathbf{b}$ is an attractor, we can obtain

$$\forall i \in \mathrm{N},\ c_i \wedge w_{ij} \le (b_i \vee c_i) \wedge w_{ij} \le b_j.$$

So if $j \in H^G(W,\mathbf{b},i)$, then we can conclude that

$$(d_i^2(W,\mathbf{b}) \vee c_i) \wedge w_{ij} = \Big(\Big\{\bigwedge_{k\in H^G(W,\mathbf{b},i)} \{b_k\}\Big\} \vee c_i\Big) \wedge w_{ij} \le b_j.$$

And by $j \notin H^G(W,\mathbf{b},i)$ we can show, $w_{ij} \le b_j$. Thus, $\big(d_i^2(W,\mathbf{b}) \vee c_i\big) \wedge w_{ij} \le b_j$. In summary we may imply the following fact:

$$\begin{aligned}
&\bigvee_{i\in H_0^G(W,\mathbf{b})} \big\{\big(d_i^2(W,\mathbf{b}) \vee c_i\big)\wedge w_{ij}\big\} \\
&= \Big(\bigvee_{i|i\in H_0^G(W,\mathbf{b}),j\in H^G(W,\mathbf{b},i)} \big\{\big(d_i^2(W,\mathbf{b}) \vee c_i\big)\wedge w_{ij}\big\}\Big)\vee \\
&\quad \vee\Big(\bigvee_{i|i\in H_0^G(W,\mathbf{b}),j\notin H^G(W,\mathbf{b},i)} \big\{\big(d_i^2(W,\mathbf{b}) \vee c_i\big)\wedge w_{ij}\big\}\Big) \le b_j.
\end{aligned} \tag{3.26}$$

On the other hand, by Lemma 3.2, for $i \in \mathrm{N}$, $d_i^2(W,\mathbf{b}) \vee c_i = b_i^2(W,\mathbf{b}) \vee c_i = b_i^2(W,\mathbf{b})$. Provided $j \in H^E(W,\mathbf{b},i)$, we have $w_{ij} = b_j$. It follows that

$$w_{ij} \wedge \big(d_i^2(W,\mathbf{b}) \vee c_i\big) \le w_{ij} \le b_j. \tag{3.27}$$

If $j \notin H^E(W,\mathbf{b},i)$, then $j \in H^G(W,\mathbf{b},i)$. Thus, $d_i^2(W,\mathbf{b}) = \bigwedge_{k\in H^G(W,\mathbf{b},i)} \{b_k\} \le b_j$; And by $j \notin H^G(W,\mathbf{b},i)$ it follows that $w_{ij} < b_j$. Therefore

$$w_{ij} \wedge (d_i^2(W,\mathbf{b}) \vee c_i) = w_{ij} \wedge d_i^2(W,\mathbf{b}) \le b_j. \tag{3.28}$$

Using (3.27) (3.28), we can by the same method as (3.26) show that

$$\bigvee_{i\in H_0^{Ge}(W,\mathbf{b})} \big\{\big(d_i^2(W,\mathbf{b}) \vee c_i\big)\wedge w_{ij}\big\} \le b_j. \tag{3.29}$$

Synthesize (3.26), (3.29), $l_1(q) = \bigvee_{i\in H_0^{Ge}(W,\mathbf{b})\cup H_0^G(W,\mathbf{b})} \big\{\big(d_i^2(W,\mathbf{b}) \vee c_i\big)\wedge w_{ij}\big\} \le b_j$. Thus by (3.17) it follows that $l_1(q) \vee l_2(q) \le b_j$. □

Theorem 3.9 *Let the fuzzy pattern* $\mathbf{b} = (b_1, ..., b_n)$ *be an attractor of (3.17), and* $W = (w_{ij})_{n\times n}$ *be the connection weight matrix related. Moreover, the following conditions hold:*

(i) For $i \in \mathrm{N}$, *and* $\forall j_1 \in H^E(W,\mathbf{b},i)$, $\forall j_2 \in H^G(W,\mathbf{b},i)$, *it follows that* $b_{j_1} < b_{j_2}$;

(ii) For $j \in \mathrm{N}$, there exists $i \in \mathrm{N}$, so that $j \in H^E(W, \mathbf{b}, i)$.

Then for any $\mathbf{x} = (x_1, ..., x_n) \in A_t^s(W, \mathbf{b}, \mathbf{c})$, $\mathbf{x}$ converges to $\mathbf{b}$ with one iteration.

Proof. For $i \in \mathrm{N}$, By the condition (i), $b_i^1(W, \mathbf{b}) < b_i^2(W, \mathbf{b})$. So using (3.22) we get, $d_i^1(W, \mathbf{b}) < d_i^2(W, \mathbf{b})$. For any $j \in \mathrm{N}$, and $\mathbf{x} = (x_1, ..., x_n) \in A_t^s(W, \mathbf{b}, \mathbf{c})$, by the condition (ii), there is $i_0 \in \mathrm{N}$, so that $j \in H^E(W, \mathbf{b}, i_0)$. Hence $i_0 \in H_0^{Ge}(W, \mathbf{b}) \cup H_0^E(W, \mathbf{b})$. Considering $j \in H^E(W, \mathbf{b}, i_0)$, and (3.23) (3.24) we get

$$\bigvee_{i \in H_0^E(W,\mathbf{b}) \cup H_0^{Ge}(W,\mathbf{b})} \{(x_i \vee c_i) \wedge w_{ij}\} \geq \bigvee_{i \in H_0^E(W,\mathbf{b}) \cup H_0^{Ge}(W,\mathbf{b})} \{(d_i^1(W, \mathbf{b}) \vee c_i) \wedge w_{ij}\}$$

$$\geq \bigvee_{i \in H_0^E(W,\mathbf{b}) \cup H_0^{Ge}(W,\mathbf{b})} \Big\{\Big(\bigvee_{k \in H^E(W,\mathbf{b},i)} \{b_k\}\Big) \wedge w_{ij}\Big\} \geq \bigvee_{k \in H^E(W,\mathbf{b},i_0)} \{b_k\} \geq b_j.$$

Therefore it follow that

$$\begin{aligned}
&\bigvee_{i \in \mathrm{N}} \{(x_i \vee c_i) \wedge w_{ij}\} \\
&\quad = \Big(\bigvee_{i \in H_0^{Ge}(W,\mathbf{b})} \{(x_i \vee c_i) \wedge w_{ij}\}\Big) \vee \Big(\bigvee_{i \in H_0^{G}(W,\mathbf{b})} \{(x_i \vee c_i) \wedge w_{ij}\}\Big) \\
&\qquad \vee \Big(\bigvee_{i \in H_0^{E}(W,\mathbf{b})} \{(x_i \vee c_i) \wedge w_{ij}\}\Big) \vee \Big(\bigvee_{i \in H_0^{L}(W,\mathbf{b})} \{(x_i \vee c_i) \wedge w_{ij}\}\Big) \\
&\quad \geq \bigvee_{i \in H_0^E(W,\mathbf{b}) \cup H_0^{Ge}(W,\mathbf{b})} \{(x_i \vee c_i) \wedge w_{ij}\} \geq b_j.
\end{aligned} \tag{3.30}$$

On the other hand, by Lemma 3.2 it follows that

$$\bigvee_{i \in \mathrm{N}} \{(x_i \vee c_i) \wedge w_{ij}\} \leq l_1(q) \vee l_2(q) \vee \Big(\bigvee_{i \in H_0^L(W,\mathbf{b})} \{(x_i \vee c_i) \wedge w_{ij}\}\Big) \leq b_j. \tag{3.31}$$

Synthesizing (3.30) and (3.31) we obtain, $\bigvee_{i \in \mathrm{N}} \{w_{ij} \wedge (x_i \vee c_i)\} = b_j \ (j \in \mathrm{N})$. That is, $\mathbf{x}$ converges with one iteration to B. □

By (3.17) and Theorem 3.9, if for $i \in \mathrm{N}$, $0 < b_i < 1$, and the conditions of Theorem 3.9 hold, then the attractive basin $A_t^s(W, \mathbf{b}, \mathbf{c})$ is non-degenerate.

Remark 3.1 Using Theorem 3.9 to compare (3.24) with a corresponding conclusion in [34] we can say that the attractive basins of a fuzzy Hopfield network with threshold are larger than the corresponding ones of the fuzzy Hopfield network in §3.1, and consequently the system (3.17) possess much better fault–tolerance.

To demonstrate the effectiveness of the proposed FNN in the storage and association of fuzzy patterns, in the following let us discuss a simulation example. Suppose $\mathrm{N} = \{1, 2, 3, 4, 5, 6\}$. By Table 3.6 we give the fuzzy patterns $\mathbf{b}^1$, $\mathbf{b}^2$, $\mathbf{b}^3$.

Table 3.6 Fuzzy patterns

No.	$\mathbf{b}^k$
1	(0.4 0.5 0.4 0.4 0.4 0.5)
2	(0.4 0.4 0.4 0.3 0.3 0.4)
3	(0.5 0.4 0.5 0.3 0.3 0.4)

Assume that the connection weight matrix of (3.17) is given as

$$W = (w_{ij})_{6\times 6} = \begin{pmatrix} 0.5 & 0 & 0 & 0.3 & 0.3 & 0 \\ 0 & 0.5 & 0 & 0.3 & 0.3 & 0 \\ 0 & 0 & 0.5 & 0.3 & 0.3 & 0 \\ 0.4 & 0.4 & 0.4 & 0 & 0 & 0.4 \\ 0 & 0 & 0 & 0.4 & 0.4 & 0 \\ 0 & 0 & 0 & 0.3 & 0.3 & 0.5 \end{pmatrix}.$$

and the threshold vector as $\mathbf{c} = (0.4, 0.4, 0.4, 0.3, 0.3, 0.4)$. Easily we can show, $W \subset W^2$, and the conditions of Theorem 3.9 hold. So $\mathbf{b}^1$, $\mathbf{b}^2$, $\mathbf{b}^3$ all are the attractors of the system (3.17). To calculate the respective attractive basins $A_t^s(W, \mathbf{b}^1, \mathbf{c})$, $A_t^s(W, \mathbf{b}^2, \mathbf{c})$, $A_t^s(W, \mathbf{b}^3, \mathbf{c})$, related to $\mathbf{b}^1$, $\mathbf{b}^2$, $\mathbf{b}^3$, we at first calculate the sets $H_0^{Ge}(W, \mathbf{b})$, $H_0^{G}(W, \mathbf{b})$, $H_0^{E}(W, \mathbf{b})$ and $H_0^{L}(W, \mathbf{b})$, as shown in Table 3.7.

Table 3.7 The index sets $H_0^{Ge}(W, \mathbf{b})$, $H_0^{G}(W, \mathbf{b})$, $H_0^{E}(W, \mathbf{b})$, $H_0^{L}(W, \mathbf{b})$

fuzzy pattern	$H_0^{Ge}(W, \mathbf{b})$	$H_0^{G}(W, \mathbf{b})$	$H_0^{E}(W, \mathbf{b})$	$H_0^{L}(W, \mathbf{b})$
$\mathbf{b} = \mathbf{b}^1$	$\emptyset$	{1,3}	{2,4,5,6}	$\emptyset$
$\mathbf{b} = \mathbf{b}^2$	{1,2,3,6}	{5}	{4}	$\emptyset$
$\mathbf{b} = \mathbf{b}^3$	{2,6}	{5}	{1,3,4}	$\emptyset$

$d_i^1(W, \mathbf{b}^1)$, $d_i^2(W, \mathbf{b}^1)$; $d_i^1(W, \mathbf{b}^2)$, $d_i^2(W, \mathbf{b}^2)$, $d_i^1(W, \mathbf{b}^3)$, $d_i^2(W, \mathbf{b}^3)$ can be calculated by using (3.22) respectively, as shown in Table 3.8.

Table 3.8 endpoints of attractive basins

No.	$d_i^1(W, \mathbf{b}^1)$	$d_i^2(W, \mathbf{b}^1)$	$d_i^1(W, \mathbf{b}^2)$	$d_i^2(W, \mathbf{b}^2)$	$d_i^1(W, \mathbf{b}^3)$	$d_i^2(W, \mathbf{b}^3)$
1	0	0.4	0	0.4	0.5	1
2	0.5	1	0	0.4	0	0.4
3	0	0.4	0	0.4	0.5	1
4	0.4	1	0.4	1	0.4	1
5	0.4	1	0	0.3	0	0.3
6	0.5	1	0	0.4	0	0.4

Therefore we can establish the respective attractive basins as

$$A_t^s(W, \mathbf{b}^1, \mathbf{c}) = [0, 0.4] \times [0.5, 1] \times [0, 0.4] \times [0.4, 1] \times [0.4, 1] \times [0.5, 1];$$
$$A_t^s(W, \mathbf{b}^2, \mathbf{c}) = [0, 0.4] \times [0, 0.4] \times [0, 0.4] \times [0.4, 1] \times [0, 0.3] \times [0, 0.4];$$
$$A_t^s(W, \mathbf{b}^3, \mathbf{c}) = [0.5, 1] \times [0, 0.4] \times [0.5, 1] \times [0.4, 1] \times [0, 0.3] \times [0, 0.4].$$

Enlarging the respective attractive basins of attractors is an important method to improve fault–tolerance of a fuzzy Hopfield network. The size of the attractive basin of an attractor of (3.17) is mainly determined by the connection weight matrix W and threshold vector $\mathbf{c}$. So it is very important and meaningful to develop some algorithms for W and $\mathbf{c}$ so that the corresponding attractive basins are as large as possible [35, 36, 38]. In addition to the analytic learning algorithms built aforementioned the BP type fuzzy algorithms [44, 48, 55], the fuzzy GA's [5, 50] and other dynamical learning procedures [9, 16, 47] and so on will be efficient tools to solve such a problem.

§3.3 Stability and fault–tolerance of FBAM

We introduce feedback connection weights in the two layer FAM's of Chapter II, and there are feedforward and feedback fuzzy information flows in the corresponding FNN's. By the connection weight matrix W fuzzy information flows forward and by the corresponding transpose W^T fuzzy information flows backward. We call such FNN's the fuzzy bidirectional associative memories [30, 33, 38], i.e. FBAM's for abbreviation. They are also nonlinear dynamic systems, on which we focus in the section. As in the case of the fuzzy Hopfield networks, the attractors of a FBAM constitute the corresponding storage patterns, each of which may possess a non-degenerate attractive basin. So a FBAM can possess good fault–tolerance [8, 15, 20]. Next let us study the following meaningful problems related to a FBAM: First, for any fuzzy connection weight matrix W, the FBAM is globally stable; Second, each equilibrium point of the system is Lyapunov stable, i.e. all equilibrium points are the attractors of the system; Finally, with some condition each attractive basin can be non-degenerate, and consequently the system possesses good fault-tolerance.

3.3.1 Stability analysis

A FBAM system can be defined as follows:

$$\begin{cases} \mathbf{x}^k = \mathbf{y}^{k-1} \circ W^T, \\ \mathbf{y}^k = \mathbf{x}^{k-1} \circ W, \end{cases} \tag{3.32}$$

where $k = 1, 2, ...,$ $\mathbf{x}^k = (x_1^k, ..., x_n^k)$, $\mathbf{y}^k = (y_1^k, ..., y_m^k)$, $W = (w_{ij})_{n \times m}$ is the connection weight matrix, and '$\circ$' is the '$\vee - \wedge$' composition operation. (3.32) can also be rewritten as

$$\begin{cases} x_i^k = \bigvee_{j\in \mathrm{M}} (y_j^{k-1} \wedge w_{ij}) \ (i \in \mathrm{N}); \\ y_j^k = \bigvee_{i\in \mathrm{N}} (x_i^{k-1} \wedge w_{ij}) \ (j \in \mathrm{M}). \end{cases} \tag{3.33}$$

By $H(\mathbf{a}^1, \mathbf{a}^2)$ we denote the Hamming metric between the fuzzy patterns $\mathbf{a}^1$ and $\mathbf{a}^2$, and $H\big[(\mathbf{a}^1, \mathbf{b}^1), (\mathbf{a}^2, \mathbf{b}^2)\big]$ denotes the Hamming metric between the fuzzy pattern pairs $(\mathbf{a}^1, \mathbf{b}^1)$ and $(\mathbf{a}^2, \mathbf{b}^2)$, that is

$$H\big[(\mathbf{a}^1, \mathbf{b}^1), (\mathbf{a}^2, \mathbf{b}^2)\big] = H(\mathbf{a}^1, \mathbf{a}^2) + H(\mathbf{b}^1, \mathbf{b}^2).$$

The fuzzy pattern pair $(\mathbf{a}, \mathbf{b})$ is called an equilibrium point of (3.32), if $\mathbf{a} = \mathbf{b}\circ W^T$, $\mathbf{b} = \mathbf{a}\circ W$. If $(\mathbf{a}, \mathbf{b})$ is an equilibrium point of (3.32), and $\forall \varepsilon > 0$, there is $\delta > 0$, then for a given fuzzy pattern pair $(\mathbf{x}, \mathbf{y})$: $H\big[(\mathbf{x}, \mathbf{a}), (\mathbf{y}, \mathbf{b})\big] < \delta$, taking $(\mathbf{x}, \mathbf{y})$ as an initial pattern of the system (3.32), $\forall k \geq 1$, it follows that $H\big[(\mathbf{x}^k, \mathbf{a}), (\mathbf{y}^k, \mathbf{b})\big] < \varepsilon$. We call $(\mathbf{a}, \mathbf{b})$ to be Lyapunov stable. A Lyapunov stable equilibrium point is called an attractor of (3.32).

Denote $W \circ W^T = (\overline{w}_{ij})_{n\times n}$, $(W \circ W^T)^k = (\overline{w}_{ij}^k)_{n\times n}$.

Theorem 3.10 *For arbitrary connection weight matrix W, each equilibrium point $(\mathbf{a}, \mathbf{b})$ of the system (3.32) is Lyapunov stable, and consequently each equilibrium point of the system is an attractor.*

Proof. Give an initial pattern pair $(\mathbf{x}^0, \mathbf{y}^0)$: $\mathbf{x}^0 = (x_1^0, ..., x_n^0)$, $\mathbf{y}^0 = (y_1^0, ..., y_m^0)$, by (3.32) and the induction method it follows that

$$\begin{cases} \mathbf{x}^{2k} = \mathbf{x}^0 \circ (W\circ W^T)^k, \\ \mathbf{x}^{2k-1} = \mathbf{x}^1 \circ (W\circ W^T)^{k-1}; \end{cases} \quad \begin{cases} \mathbf{y}^{2k} = \mathbf{y}^0 \circ (W^T\circ W)^k, \\ \mathbf{y}^{2k-1} = \mathbf{y}^1 \circ (W^T\circ W)^{k-1}. \end{cases} \tag{3.34}$$

Since $(\mathbf{a}, \mathbf{b})$ is an equilibrium point of (3.32), we have, $\mathbf{a} = \mathbf{b}\circ W^T$, $\mathbf{b} = \mathbf{a}\circ W$. Therefore for any nonnegative integer p, we can conclude that

$$\mathbf{a} = \mathbf{a} \circ (W\circ W^T)^p, \ \mathbf{b} = \mathbf{b} \circ (W^T\circ W)^p. \tag{3.35}$$

For $\varepsilon > 0$, choose $\delta = \varepsilon/(2m+2n)$. Give an initial fuzzy pattern pair $(\mathbf{x}^0, \mathbf{y}^0)$, satisfying $H\big[(\mathbf{x}^0, \mathbf{a}), (\mathbf{y}^0, \mathbf{b})\big] < \delta$. Then $\forall i \in \mathrm{N}$, $|x_i^0 - a_i| < \varepsilon/(2m+2n)$; $\forall j \in \mathrm{M}$, $|y_j^0 - b_j| < \varepsilon/(2n+2m)$. And by $\mathbf{x}^1 = \mathbf{y}^0 \circ W^T$, we get, $\forall i \in \mathrm{N}$,

$$|a_i - x_i^1| = \Big|\bigvee_{j\in \mathrm{M}} \{b_j \wedge w_{ij}\} - \bigvee_{j\in \mathrm{M}} \{y_j^0 \wedge w_{ij}\}\Big|.$$

By Lemma 3.1 easily we have, $\big|b_j \wedge w_{ij} - y_j^0 \wedge w_{ij}\big| \leq |b_j - y_j^0| < \varepsilon/(2m+2n)$. Also Lemma 3.1 may imply, $|a_i - x_i^1| < \varepsilon/(2m+2n)$ $(i \in \mathrm{N})$. And by (3.34) (3.35) it follows that

$$\big|x_i^{2k} - a_i\big| = \Big|\bigvee_{j\in \mathrm{N}} \{x_j^0 \wedge \overline{w}_{ij}^k\} - \bigvee_{j\in \mathrm{N}} \{a_j \wedge \overline{w}_{ij}^k\}\Big|.$$

Moreover, $\left|x_j^0 \wedge w_{ij} - a_j \wedge w_{ij}\right| \leq \left|x_j^0 - a_j\right| < \varepsilon/(2m+2n)$. So using Lemma 3.1 we get, $\left|x_i^{2k} - a_i\right| < \varepsilon/(2m+2n)$, that is, $H(\mathbf{x}^{2k}, \mathbf{a}) < \varepsilon/2$. With the same reason we have

$$\left|x_i^{2k-1} - a_i\right| = \left|\bigvee_{j\in\mathrm{N}}\{x_j^1 \wedge \overline{w}_{ij}^{k-1}\} - \bigvee_{j\in\mathrm{N}}\{a_j \wedge \overline{w}_{ij}^{k-1}\}\right|.$$

And $\left|x_j^1 \wedge \overline{w}_{ij}^{k-1} - a_j \wedge \overline{w}_{ij}^{k-1}\right| \leq \left|x_j^1 - a_j\right| < \varepsilon/(2m+2n)$. So $\left|x_i^{2k-1} - a_i\right| < \varepsilon/(2m+2n)$, i.e. $H\left(\mathbf{x}^{2k-1}, \mathbf{a}\right) < \varepsilon$. In summary, for $k \geq 1$, it follows that $H\left(\mathbf{x}^k, \mathbf{a}\right) < \varepsilon/2$. Similarly we can show, for $k \geq 1$, $H\left(\mathbf{y}^k, \mathbf{b}\right) < \varepsilon/2$. Therefore

$$H\left[(\mathbf{x}^k, \mathbf{y}^k), (\mathbf{a}, \mathbf{b})\right] = H(\mathbf{x}^k, \mathbf{a}) + H(\mathbf{y}^k, \mathbf{b}) < \varepsilon \ (k \geq 1).$$

Thus, $(\mathbf{a}, \mathbf{b})$ is Lyapunov stable, and therefore $(\mathbf{a}, \mathbf{b})$ is an attractor. □

By Theorem 3.10, we can in the following treat an equilibrium point and an attractor of (3.32) as an identity. We call (3.32) to be globally stable, if for any fuzzy pattern pairs $(\mathbf{x}, \mathbf{y})$, and take it as an initial pattern, (3.32) converges to an attractor or a limit cycle. Let us now show the system (3.32) is globally stable for any fuzzy connection weight matrix W.

Theorem 3.11 *Let the fuzzy matrix W be an arbitrary connection weight matrix of (3.32). Then the system converges to a limit cycle whose period does not exceed* 2, *with finite iterations. That is, there exist the fuzzy pattern pairs* $(\mathbf{a}^1, \mathbf{b}^1)$, $(\mathbf{a}^2, \mathbf{b}^2)$, *so that*

$$\begin{cases} \mathbf{a}^2 = \mathbf{b}^1 \circ W^T, \\ \mathbf{b}^2 = \mathbf{a}^1 \circ W; \end{cases} \quad \begin{cases} \mathbf{a}^1 = \mathbf{b}^2 \circ W^T, \\ \mathbf{b}^1 = \mathbf{a}^2 \circ W. \end{cases} \tag{3.36}$$

Proof. By the definition of '∘', The composition fuzzy matrix of two fuzzy matrices, based on '∘' does not generate new elements. So the fuzzy matrix sequences $\left\{(W \circ W^T)^k \middle| k = 1, 2, ...\right\}$ and $\left\{(W^T \circ W)^k \middle| k = 1, 2, ...\right\}$ contain at most finite different terms, respectively. By Lemma 3.1, This two fuzzy matrix sequences are increasing. So there is a natural number l, satisfying

$$\forall p \geq l, (W \circ W^T)^l = (W \circ W^T)^p, \ \ (W^T \circ W)^l = (W^T \circ W)^p.$$

Using (3.32), (3.34) easily we can imply

$$\begin{cases} \mathbf{x}^{2l+4} = \mathbf{x}^0 \circ (W \circ W^T)^{l+2} = \mathbf{x}^0 \circ (W \circ W^T)^l \triangleq \mathbf{a}^1, \\ \mathbf{y}^{2l+4} = \mathbf{y}^0 \circ (W^T \circ W)^{l+2} = \mathbf{y}^0 \circ (W^T \circ W)^l \triangleq \mathbf{b}^1; \end{cases}$$

$$\begin{cases} \mathbf{x}^{2l+3} = \mathbf{x}^1 \circ (W \circ W^T)^{l+1} = \mathbf{x}^1 \circ (W \circ W^T)^l \triangleq \mathbf{a}^2, \\ \mathbf{y}^{2l+3} = \mathbf{y}^1 \circ (W^T \circ W)^{l+1} = \mathbf{y}^1 \circ (W^T \circ W)^l \triangleq \mathbf{b}^2. \end{cases}$$

Also we can conclude that

$$\begin{aligned}\mathbf{b}^1 \circ W^T &= \big(\mathbf{y}^0 \circ (W^T \circ W)^l\big)\circ W^T\\ &= \mathbf{y}^0 \circ W^T \circ (W \circ W^T)^l = \mathbf{x}^1 \circ (W \circ W^T)^l = \mathbf{a}^2;\\ \mathbf{b}^2 \circ W^T &= \big(\mathbf{y}^1 \circ (W^T \circ W)^l\big)\circ W^T\\ &= \mathbf{y}^1 \circ W^T \circ (W \circ W^T)^l = \mathbf{x}^0 \circ (W \circ W^T)^{l+1} = \mathbf{a}^1.\end{aligned}$$

Similarly we have, $\mathbf{a}^1 \circ W = \mathbf{b}^2,\ \mathbf{a}^2 \circ W = \mathbf{b}^1$. That is

$$\begin{cases}\mathbf{a}^2 = \mathbf{b}^1 \circ W^T,\\ \mathbf{b}^2 = \mathbf{a}^1 \circ W;\end{cases} \qquad \begin{cases}\mathbf{a}^1 = \mathbf{b}^2 \circ W^T,\\ \mathbf{b}^1 = \mathbf{a}^2 \circ W.\end{cases}$$

So the system (3.32) converges to a limit cycle whose period does not exceed 2. □

3.3.2 Fault–tolerance analysis

It is well known that fault–tolerance of a dynamic system can be determined by the spatial sizes and shapes of attractive basins of corresponding attractors [8, 20]. The attractive basin of the attractor $(\mathbf{a}, \mathbf{b})$ of (3.32) means a subset F_q of $[0, 1]^n \times [0, 1]^m$, so that for any $(\mathbf{x}, \mathbf{y}) \in F_q$, taking it as an initial pattern, (3.32) converges to $(\mathbf{a}, \mathbf{b})$. The attractive basin F_q is non-degenerate means that as a subset of $[0, 1]^n \times [0, 1]^m$ its volume is nonzero.

Suppose a connection weight matrix of (3.32) is a fuzzy matrix $W = (w_{ij})_{n\times m}$. For a given fuzzy pattern pair $(\mathbf{a}, \mathbf{b})$: $\mathbf{a} = (a_1, ..., a_n)$, $\mathbf{b} = (b_1, ..., b_m)$ and for $i \in \mathrm{N},\ j \in \mathrm{M}$, we introduce the following notations:

$$\begin{aligned}&B^{G_J}(W, \mathbf{b}, i) = \big\{j \in \mathrm{M}\big|w_{ij} > b_j\big\},\ \ B^{G_I}(W, \mathbf{a}, j) = \big\{i \in \mathrm{N}\big|w_{ij} > a_i\big\},\\ &B^{E_J}(W, \mathbf{b}, i) = \big\{j \in \mathrm{M}\big|w_{ij} = b_j\big\},\ \ B^{E_I}(W, \mathbf{a}, j) = \big\{i \in \mathrm{N}\big|w_{ij} = a_i\big\}.\end{aligned}$$

We define the following sets:

$$\begin{aligned}&B_0^{Ge_I}(W, \mathbf{b}) = \big\{i \in \mathrm{N}\big|B^{G_J}(W, \mathbf{b}, i) \neq \emptyset,\ B^{E_J}(W, \mathbf{b}, i) \neq \emptyset\big\},\\ &B_0^{Ge_J}(W, \mathbf{a}) = \big\{j \in \mathrm{M}\big|B^{G_I}(W, \mathbf{a}, j) \neq \emptyset,\ B^{E_I}(W, \mathbf{a}, j) \neq \emptyset\big\},\\ &B_0^{E_I}(W, \mathbf{b}) = \big\{i \in \mathrm{N}\big|B^{G_J}(W, \mathbf{b}, i) = \emptyset,\ B^{E_J}(W, \mathbf{b}, i) \neq \emptyset\big\},\\ &B_0^{E_J}(W, \mathbf{a}) = \big\{j \in \mathrm{M}\big|B^{G_I}(W, \mathbf{a}, j) = \emptyset,\ B^{E_I}(W, \mathbf{a}, j) \neq \emptyset\big\},\\ &B_0^{G_I}(W, \mathbf{b}) = \big\{i \in \mathrm{N}\big|B^{G_J}(W, \mathbf{b}, i) \neq \emptyset,\ B^{E_J}(W, \mathbf{b}, i) = \emptyset\big\},\\ &B_0^{G_J}(W, \mathbf{a}) = \big\{j \in \mathrm{M}\big|B^{G_I}(W, \mathbf{a}, j) \neq \emptyset,\ B^{E_I}(W, \mathbf{a}, j) = \emptyset\big\}.\end{aligned}$$

Similarly with Definition 3.2, we build the GE condition for (3.32), that is, $(W,\ \mathbf{a},\ \mathbf{b})$ is called to satisfy GE condition if the following conditions hold:

(i) $\forall i \in \mathrm{N}$, $\forall j_1 \in B^{E_J}(W, \mathbf{b}, i)$, $j_2 \in B^{G_J}(W, \mathbf{b}, i)$, $b_{j_1} < b_{j_2}$;
(ii) $\forall j \in \mathrm{M}$, $\forall i_1 \in B^{E_I}(W, \mathbf{a}, j)$, $i_2 \in B^{G_I}(W, \mathbf{a}, j)$, $a_{i_1} < a_{i_2}$;
(iii) $\forall j \in \mathrm{M}$, there is $i \in \mathrm{N}$, satisfying $j \in B^{E_J}(W, \mathbf{b}, i)$;
(iv) $\forall i \in \mathrm{N}$, it follows that there is $j \in \mathrm{M}$, so that $i \in B^{E_I}(W, \mathbf{a}, j)$.
For $i \in \mathrm{N}$, $j \in \mathrm{M}$, define

$$
\begin{aligned}
d^1_{\mathbf{a}i}(W,\mathbf{b}) &= \begin{cases} \bigvee\limits_{k \in B^{E_J}(W,\mathbf{b},i)} \{b_k\}, & B^{E_J}(W,\mathbf{b},i) \neq \emptyset, \\ 0, & B^{E_J}(W,\mathbf{b},i) = \emptyset; \end{cases} \\
d^2_{\mathbf{a}i}(W,\mathbf{b}) &= \begin{cases} \bigwedge\limits_{k \in B^{G_J}(W,\mathbf{b},i)} \{b_k\}, & B^{G_J}(W,\mathbf{b},i) \neq \emptyset, \\ 1, & B^{G_J}(W,\mathbf{b},i) = \emptyset; \end{cases} \\
d^1_{\mathbf{b}j}(W,\mathbf{a}) &= \begin{cases} \bigvee\limits_{k \in B^{E_I}(W,\mathbf{a},j)} \{a_k\}, & B^{E_I}(W,\mathbf{a},j) \neq \emptyset, \\ 0, & B^{E_I}(W,\mathbf{a},j) = \emptyset; \end{cases} \\
d^2_{\mathbf{b}j}(W,\mathbf{a}) &= \begin{cases} \bigwedge\limits_{k \in B^{G_I}(W,\mathbf{a},j)} \{a_k\}, & B^{G_I}(W,\mathbf{a},j) \neq \emptyset, \\ 1, & B^{G_I}(W,\mathbf{a},j) = \emptyset. \end{cases}
\end{aligned}
\tag{3.37}
$$

Thus, we can introduce the following subsets related to $(\mathbf{a}, \mathbf{b})$ to build the attractive basin.

$$
\begin{aligned}
A^{\mathbf{a}}_W(\mathbf{a}, \mathbf{b}) = \Big\{ &(x_1, ..., x_n) \in [0,1]^n \Big| x_i \in \big[0,\ d^2_{\mathbf{a}i}(W,\mathbf{b})\big]\ \big(i \in B^{G_I}_0(W,\mathbf{b})\big), \\
&x_i \in \big[d^1_{\mathbf{a}i}(W,\mathbf{b}),\ d^2_{\mathbf{a}i}(W,\mathbf{b})\big]\ \big(i \in B^{Ge_I}_0(W,\mathbf{b})\big), \\
&x_i \in \big[d^1_{\mathbf{a}i}(W,\mathbf{b}),\ 1\big]\ \big(i \in B^{E_I}_0(W,\mathbf{b})\big),\ x_i \in [0,\ 1]\ (\text{otherwise}) \Big\};
\end{aligned}
\tag{3.38}
$$

$$
\begin{aligned}
A^{\mathbf{b}}_W(\mathbf{a}, \mathbf{b}) = \Big\{ &(y_1, ..., y_m) \in [0,1]^m \Big| y_j \in \big[0,\ d^2_{\mathbf{b}j}(W,\mathbf{a})\big]\ \big(j \in B^{G_J}_0(W,\mathbf{a})\big), \\
&y_j \in \big[d^1_{\mathbf{b}j}(W,\mathbf{a}),\ d^2_{\mathbf{b}j}(W,\mathbf{a})\big]\ \big(j \in B^{Ge_J}_0(W,\mathbf{a})\big), \\
&y_j \in \big[d^1_{\mathbf{b}j}(W,\mathbf{a}),\ 1\big]\ \big(j \in B^{E_J}_0(W,\mathbf{a})\big),\ y_j \in [0,\ 1]\ (\text{otherwise}) \Big\}.
\end{aligned}
\tag{3.39}
$$

Theorem 3.12 *Let the fuzzy matrix W be a connection weight matrix of (3.32), and the fuzzy pattern pair $(\mathbf{a}, \mathbf{b})$ be an attractor, $(W, \mathbf{a}, \mathbf{b})$ satisfies GE condition. Then for any $(\mathbf{x}, \mathbf{y}) \in A^{\mathbf{a}}_W(\mathbf{a}, \mathbf{b}) \times A^{\mathbf{b}}_W(\mathbf{a}, \mathbf{b})$, taking it as an initial pattern pair, the system converges to $(\mathbf{a}, \mathbf{b})$ with one iteration.*

Proof. Considering $(W, \mathbf{a}, \mathbf{b})$ satisfies GE condition, we can conclude, $\forall i \in \mathrm{N}$, $d^1_{\mathbf{a}i}(W,\mathbf{b}) < b^2_{\mathbf{a}i}(W,\mathbf{b})$. Let $W = (w_{ij})_{n\times m}$, $\mathbf{a} = (a_1, ..., a_n)$, $\mathbf{b} = (b_1, ..., b_m)$. For any $j \in \mathrm{M}$, and $\mathbf{x} = (x_1, ..., x_n) \in A^{\mathbf{a}}_W(\mathbf{a}, \mathbf{b})$, by GE condition, there is $i_0 \in \mathrm{N}$, so that $j \in B^{E_J}(W, \mathbf{b}, i_0)$. Hence $i_0 \in B^{Ge_I}_0(W, \mathbf{b}) \cup B^{E_I}_0(W, \mathbf{b})$.

Considering $j \in B^{E_J}(W, \mathbf{b}, i_0)$, we have

$$\bigvee_{i\in B_0^{E_I}(W,\mathbf{b})\cup B_0^{Ge_I}(W,\mathbf{b})} \{x_i \wedge w_{ij}\} \geq \bigvee_{i\in B_0^{E_I}(W,\mathbf{b})\cup B_0^{Ge_I}(W,\mathbf{b})} \{d_{\mathbf{a}i}^1(W,\mathbf{b}) \wedge w_{ij}\}$$

$$\geq \bigvee_{i\in B_0^{E_I}(W,\mathbf{b})\cup B_0^{Ge_I}(W,\mathbf{b})} \Big\{\Big(\bigvee_{k\in B^{E_J}(W,\mathbf{b},i)} \{b_k\}\Big) \wedge w_{ij}\Big\}$$

$$\geq \bigvee_{k\in B^{E_J}(W,\mathbf{b},i_0)} \{b_k\} \geq b_j.$$

Therefore it follows that

$$\begin{aligned}&\bigvee_{i\in \mathrm{N}} \{x_i \wedge w_{ij}\}\\ &\geq \Big(\bigvee_{i\in B_0^{Ge_I}(W,\mathbf{b})} \{x_i \wedge w_{ij}\}\Big) \vee \Big(\bigvee_{i\in B_0^{G_I}(W,\mathbf{b})} \{x_i \wedge w_{ij}\}\Big) \vee \Big(\bigvee_{i\in B_0^{E_I}(W,\mathbf{b})} \{x_i \wedge w_{ij}\}\Big)\\ &\geq \Big(\bigvee_{i\in B_0^{E_I}(W,\mathbf{b})\cup B_0^{Ge_I}(W,\mathbf{b})} \{x_i \wedge w_{ij}\}\Big) \geq b_j.\end{aligned} \tag{3.40}$$

On the other hand, for $j \in \mathrm{M}$, denote

$$l_1(q) \triangleq \bigvee_{i\in B_0^{G_I}(W,\mathbf{b})} \{w_{ij} \wedge d_{\mathbf{a}i}^2(W,\mathbf{b})\} = \bigvee_{i\in B_0^{G_I}(W,\mathbf{b})} \Big\{w_{ij} \wedge \Big(\bigwedge_{k\in B^{G_J}(W,\mathbf{b},i)} \{b_k\}\Big)\Big\}.$$

By $j \in B^{G_J}(W, \mathbf{b}, i)$ we imply, $\bigwedge_{k\in B^{G_J}(W,\mathbf{b},i)} \{b_k\} \leq b_j$. And by $j \notin B^{G_J}(W, \mathbf{b}, i)$ we imply, $w_{ij} \leq b_j$. So it follows that $\bigvee_{i\in B_0^{G_I}(W,\mathbf{b})} \{x_i \wedge w_{ij}\} \leq l_1(q) \leq b_j$. Denote $l_2(q) \triangleq \bigvee_{i\in B_0^{E_I}(W,\mathbf{b})} \{w_{ij} \wedge 1\}$. If $j \in B^{E_J}(W, \mathbf{b}, i)$, then $w_{ij} = b_j$. And when $j \notin B^{E_J}(W, \mathbf{b}, i)$, since $B^{G_J}(W, \mathbf{b}, i) = \emptyset$, we get, $w_{ij} < b_j$. Therefore

$$\bigvee_{i\in B_0^{E_I}(W,\mathbf{b})} \{x_i \wedge w_{ij}\} \leq l_2(q) \leq b_j. \tag{3.41}$$

With the same step we can conclude that

$$b_j \geq l_3(q) \triangleq \bigvee_{i\in B_0^{Ge_I}(W,\mathbf{b})} \{w_{ij} \wedge d_{\mathbf{a}i}^2(W,\mathbf{b})\} \geq \bigvee_{i\in B_0^{Ge_I}(W,\mathbf{b})} \{w_{ij} \wedge x_i\}. \tag{3.42}$$

So we can conclude that

$$\begin{aligned}\bigvee_{i\in \mathrm{N}} \{x_i \wedge w_{ij}\} &= l_1(q) \vee l_2(q) \vee l_3(q) \vee\\ &\vee \Big(\bigvee_{i\notin B_0^{Ge_I}(W,\mathbf{b})\cup B_0^{E_I}(W,\mathbf{b})\cup B_0^{G_I}(W,\mathbf{b})} \{x_i \wedge w_{ij}\}\Big) \leq b_j.\end{aligned}$$

Synthesizing (3.41) and (3.42) we get, $\bigvee_{i\in \mathrm{N}} \{w_{ij} \wedge x_i\} = b_j$ ($j \in$ N). That is, $\mathbf{x} \circ W = \mathbf{b}$. With the same reason, for $\mathbf{y} \in A_W^{\mathbf{b}}(\mathbf{a}, \mathbf{b})$, it follows that $\mathbf{y} \circ W^T = \mathbf{a}$. Thus, taking ($\mathbf{x}$, $\mathbf{y}$) as an initial pattern the system converges to the attractor ($\mathbf{a}$, $\mathbf{b}$). □

Remark 3.2 Under the conditions of Theorem 3.12, if $\forall i \in$ N, $j \in$ M, $0 < a_i < 1$, $0 < b_j < 1$, then by (3.38) and (3.39) we can get a non-generate attractive basin $A_W^{\mathbf{a}}(\mathbf{a}, \mathbf{b}) \times A_W^{\mathbf{b}}(\mathbf{a}, \mathbf{b})$.

3.3.3 A simulation example

Let N = {1, 2, 3, 4, 5, 6}, M = {1, 2, 3}. By Table 3.8 we give the fuzzy pattern pairs ($\mathbf{a}^1$, $\mathbf{b}^1$), ($\mathbf{a}^2$, $\mathbf{b}^2$), ($\mathbf{a}^3$, $\mathbf{b}^3$) for training the FBAM.

Table 3.8 Fuzzy pattern family

k	$\mathbf{a}^k$	$\mathbf{b}^k$
1	(0.4 0.4 0.6 0.5 0.4 0.6)	(0.6 0.5 0.4)
2	(0.3 0.3 0.4 0.5 0.3 0.4)	(0.4 0.5 0.3)
3	(0.4 0.4 0.3 0.3 0.4 0.3)	(0.3 0.3 0.4)

Considering the GE condition, we propose the connection weight W of the system (3.32):

$$W^T = (w_{ji})_{3\times 6} = \begin{pmatrix} 0.3 & 0.3 & 0.6 & 0.4 & 0.3 & 0.6 \\ 0.3 & 0 & 0.4 & 0.5 & 0 & 0.4 \\ 0.4 & 0.4 & 0.3 & 0.3 & 0.4 & 0.3 \end{pmatrix}$$

Table 3.9 $B_0^{Ge_I}(W, \mathbf{b}), ..., B_0^{G_J}(W, \mathbf{a})$

	$(\mathbf{a}^1, \mathbf{b}^1)$	$(\mathbf{a}^2, \mathbf{b}^2)$	$(\mathbf{a}^3, \mathbf{b}^3)$
$B_0^{Ge_I}(W, \mathbf{b})$	$\emptyset$	$\{3, 6\}$	$\emptyset$
$B_0^{Ge_J}(W, \mathbf{a})$	$\emptyset$	$\{1\}$	$\emptyset$
$B_0^{G_I}(W, \mathbf{b})$	$\emptyset$	$\{1, 2, 5\}$	$\{3, 4, 6\}$
$B_0^{G_J}(W, \mathbf{a})$	$\emptyset$	$\{3\}$	$\{1, 2\}$
$B_0^{E_I}(W, \mathbf{b})$	$\{1, 2, 3, 4, 5, 6\}$	$\{4\}$	$\{1, 2, 5\}$
$B_0^{E_J}(W, \mathbf{a})$	$\{1, 2, 3\}$	$\{2\}$	$\{3\}$

Easily the fuzzy pattern pair $(\mathbf{a}^k, \mathbf{b}^k)$ $(k = 1, 2, 3)$ is an attractor of (3.32). So the conditions of Theorem 3.12 hold. To calculate the attractive basin $A_W^{\mathbf{a}}(\mathbf{a}^k, \mathbf{b}^k) \times A_W^{\mathbf{b}}(\mathbf{a}^k, \mathbf{b}^k)$ of $(\mathbf{a}^k, \mathbf{b}^k)$, we firstly compute the following sets:

$$B_0^{Ge_I}(W, \mathbf{b}),\ B_0^{Ge_J}(W, \mathbf{a}), B_0^{E_I}(W, \mathbf{b}),\ B_0^{E_J}(W, \mathbf{a}),\ B_0^{G_I}(W, \mathbf{b}),\ B_0^{G_J}(W, \mathbf{a}).$$

By (3.38) (3.39) we calculate $d^h_{\mathbf{a}i}(W, \mathbf{b}^k)$, $d^h_{\mathbf{b}j}(W, \mathbf{a}^k)$, $(i \in \mathrm{N},\ j \in \mathrm{M},\ k = 1,2,3,\ h = 1,2)$, as shown in Table 3.9.

Table 3.10 values of $d^1_{\mathbf{a}1}(W, \mathbf{b}^k), .., d^1_{\mathbf{a}6}(W, \mathbf{b}^k)$

No.	$d^1_{\mathbf{a}1}(W, \mathbf{b}^k)$	$d^1_{\mathbf{a}2}(W, \mathbf{b}^k)$	$d^1_{\mathbf{a}3}(W, \mathbf{b}^k)$	$d^1_{\mathbf{a}4}(W, \mathbf{b}^k)$	$d^1_{\mathbf{a}5}(W, \mathbf{b}^k)$	$d^1_{\mathbf{a}6}(W, \mathbf{b}^k)$
1	0.4	0.4	0.6	0.5	0.4	0.6
2	0	0	0.3	0.5	0	0.3
3	0.4	0.4	0	0	0.4	0

Table 3.11 values of $d^2_{\mathbf{a}1}(W, \mathbf{b}^k), ..., d^2_{\mathbf{a}6}(W, \mathbf{b}^k)$

No.	$d^2_{\mathbf{a}1}(W, \mathbf{b}^k)$	$d^2_{\mathbf{a}2}(W, \mathbf{b}^k)$	$d^2_{\mathbf{a}3}(W, \mathbf{b}^k)$	$d^2_{\mathbf{a}4}(W, \mathbf{b}^k)$	$d^2_{\mathbf{a}5}(W, \mathbf{b}^k)$	$d^2_{\mathbf{a}6}(W, \mathbf{b}^k)$
1	1	1	1	1	1	1
2	0.3	0.3	0.4	1	0.3	0.4
3	1	1	0.3	0.3	1	0.3

Table 3.12 values of $d^1_{\mathbf{b}1}(W, \mathbf{a}^k), ..., d^1_{\mathbf{b}3}(W, \mathbf{a}^k)$, $d^2_{\mathbf{b}1}(W, \mathbf{a}^k), .., d^2_{\mathbf{b}3}(W, \mathbf{a}^k)$

No.	$d^1_{\mathbf{b}1}(W, \mathbf{a}^k)$	$d^1_{\mathbf{b}2}(W, \mathbf{a}^k)$	$d^1_{\mathbf{b}3}(W, \mathbf{a}^k)$	$d^2_{\mathbf{b}1}(W, h\mathbf{a}^k)$	$d^2_{\mathbf{b}2}(W, \mathbf{a}^k)$	$d^2_{\mathbf{b}3}(W, \mathbf{a}^k)$
1	0.6	0.5	0.4	1	1	1
2	0.3	0.5	0	0.4	1	0.3
3	0	0	0.4	0.3	0.3	1

By (3.38) (3.39) the attractive basin is $A^{\mathbf{a}}_W(\mathbf{a}^k, \mathbf{b}^k) \times A^{\mathbf{b}}_W(\mathbf{a}^k, \mathbf{b}^k)$ for the attractor $(\mathbf{a}^k, \mathbf{b}^k)$ $(k = 1,2,3)$, where

$$\begin{cases} A^{\mathbf{a}}_W(\mathbf{a}^1, \mathbf{b}^1) = [0.4, 1] \times [0.4, 1] \times [0.6, 1] \times [0.5, 1] \times [0.4, 1] \times [0.6, 1], \\ A^{\mathbf{b}}_W(\mathbf{a}^1, \mathbf{b}^1) = [0.6, 1] \times [0.5, 1] \times [0.4, 1]; \end{cases}$$

$$\begin{cases} A^{\mathbf{a}}_W(\mathbf{a}^2, \mathbf{b}^2) = [0, 0.3] \times [0, 0.3] \times [0.3, 0.4] \times [0.5, 1] \times [0, 0.3] \times [0.3, 0.4], \\ A^{\mathbf{b}}_W(\mathbf{a}^2, \mathbf{b}^2) = [0.3, 0.4] \times [0.5, 1] \times [0, 0.3]; \end{cases}$$

$$\begin{cases} A^{\mathbf{a}}_W(\mathbf{a}^3, \mathbf{b}^3) = [0.4, 1] \times [0.4, 1] \times [0, 0.3] \times [0, 0.3]\, [0.4, 1] \times [0, 0.3]; \\ A^{\mathbf{b}}_W(\mathbf{a}^3, \mathbf{b}^3) = [0, 0.3] \times [0, 0.3] \times [0.4, 1]. \end{cases}$$

A FBAM is globally stable and its equilibrium points are Lyapunov stable, which constitutes the theoretic basis for application of FBAM's. A FBAM possesses good fault–tolerance, which is necessary for FBAM's to store and recall fuzzy patterns. Like the fuzzy Hopfield networks, FBAM's as (3.32) can possess large attractive basins of corresponding attractors by suitable connection weight matrix W. By Theorem 3.12, we choose W so that the GE condition should be ensured, since by (3.38) (3.39), the corresponding attractive basins can be non-degenerate accordingly, and consequently fault–tolerance of the system can be guaranteed. So developing some succinct learning algorithm for W to enlarge the sizes of attractive basins is important and meaningful for future research in the field.

§3.4 Learning algorithm for FBAM

Establishing the connection weight matrix suitably is a key factor for a FBAM to possess good fault–tolerance. So in this section we focus mainly on a learning algorithm for the connection weight matrix so that the attractive basins related are non-degenerate, and therefore the system may possess good fault–tolerant.

3.4.1 Learning algorithm based on fault–tolerance

Give $p \in \mathbb{N}$, and the set $\mathrm{P} = \{1, 2, ..., p\}$. Also give the fuzzy pattern family for training networks:

$$(\mathcal{A},\, \mathcal{B}) = \left\{(\mathbf{a}^k,\, \mathbf{b}^k)\big|\mathbf{a}^k = (a_1^k, ..., a_n^k),\ \mathbf{b}^k = (b_1^k, ..., b_m^k),\ k \in \mathrm{P}\right\}.$$

Let us now build an analytic learning algorithm for W of (3.32). When $(\mathcal{A},\, \mathcal{B})$ satisfies some conditions, each fuzzy pattern pair in $(\mathcal{A},\, \mathcal{B})$ is an attractor of (3.32), moreover, the corresponding attractive basins are non-degenerate, and so good fault–tolerance of (3.32) can be guaranteed.

For $i \in \mathrm{N},\ j \in \mathrm{M}$, considering the notations defined we recall the following expressions:

$$G_{ij}(\mathcal{A}, \mathcal{B}) = \{k \in \mathrm{P}|a_i^k > b_j^k\},\quad L_{ij}(\mathcal{A}, \mathcal{B}) = \{k \in \mathrm{P}|b_j^k > a_i^k\};$$
$$E_{\mathcal{A}}(i) = \left\{k \in \mathrm{P}\Big|a_i^k = \bigvee_{k' \in \mathrm{P}} \{a_i^{k'}\}\right\},\quad E_{\mathcal{B}}(j) = \left\{k \in \mathrm{P}\Big|b_j^k = \bigvee_{k' \in \mathrm{P}} \{b_j^{k'}\}\right\}.$$

To improve fault–tolerance of FBAM's we introduce the following new notations:

$$\zeta(i,j) = \begin{cases} \Big(\bigwedge\limits_{k \in L_{ij}(\mathcal{A},\mathcal{B})} \{a_i^k\}\Big) \wedge \Big(\bigwedge\limits_{k \in G_{ij}(\mathcal{A},\mathcal{B})} \{b_j^k\}\Big), & L_{ij}(\mathcal{A}, \mathcal{B}) \cup G_{ij}(\mathcal{A}, \mathcal{B}) \neq \emptyset; \\ 1, & L_{ij}(\mathcal{A}, \mathcal{B}) \cup G_{ij}(\mathcal{A}, \mathcal{B}) = \emptyset. \end{cases}$$

$$E_{\mathcal{B}}^{\zeta}(i,j) = \begin{cases} \{k \in \mathrm{P}\big|b_j^k = \zeta(i,j)\}, & L_{ij}(\mathcal{A}, \mathcal{B}) \cup G_{ij}(\mathcal{A}, \mathcal{B}) \neq \emptyset; \\ \emptyset, & L_{ij}(\mathcal{A}, \mathcal{B}) \cup G_{ij}(\mathcal{A}, \mathcal{B}) = \emptyset. \end{cases} \quad (3.43)$$

$$E_{\mathcal{A}}^{\zeta}(i,j) = \begin{cases} \{k \in \mathrm{P}\big|a_i^k = \zeta(i,j)\}, & G_{ij}(\mathcal{A}, \mathcal{B}) \cup L_{ij}(\mathcal{A}, \mathcal{B}) \neq \emptyset; \\ \emptyset, & G_{ij}(\mathcal{A}, \mathcal{B}) \cup L_{ij}(\mathcal{A}, \mathcal{B}) = \emptyset. \end{cases} \quad (3.44)$$

Let us now develop an analytic learning algorithm to establish a connection weight matrix $W = W_* = (w^*_{ij})_{n\times m}$ of (3.32).

$$w^*_{ij} = \begin{cases} \zeta(i,j), & L_{ij}(\mathcal{A},\mathcal{B}) \cup G_{ij}(\mathcal{A},\mathcal{B}) \neq \emptyset; \\ \bigvee\limits_{k\in \mathrm{P}} \{a^k_i\}, & L_{ij}(\mathcal{A},\mathcal{B}) \cup G_{ij}(\mathcal{A},\mathcal{B}) = \emptyset. \end{cases} \tag{3.45}$$

Definition 3.7 We call the fuzzy pattern pair family $(\mathcal{A},\mathcal{B})$ to be co-existed, if the following conditions hold:

(i) $\forall i \in \mathrm{N}$, there is $j \in \mathrm{M}$, so that $\forall k \in \mathrm{P}$, $a^k_i = b^k_j$;

(ii) $\forall j \in \mathrm{M}$, there is $i \in \mathrm{N}$, satisfying $\forall k \in \mathrm{P}$, $a^k_i = b^k_j$;

(iii) $\forall i \in \mathrm{N}$, $\bigcup\limits_{j\in\mathrm{M}} E^{\zeta}_{\mathcal{A}}(i,j) \cup E_{\mathcal{A}}(i) = \mathrm{P}$, $\forall j \in \mathrm{M}$, $\bigcup\limits_{i\in\mathrm{N}} E^{\zeta}_{\mathcal{B}}(i,j) \cup E_{\mathcal{B}}(j) = \mathrm{P}$.

(iv) $\forall k \in \mathrm{P}$, $\forall i \in \mathrm{N}$, if $j_1 \in B^{G_J}(W_*, \mathbf{b}^k, i)$, $j_2 \in B^{E_J}(W_*, \mathbf{b}^k, i)$, we have, $b^k_{j_2} < b^k_{j_1}$;

(v) For any $j \in \mathrm{M}$, if $i_1 \in B^{G_I}(W_*, \mathbf{a}^k, j)$, $i_2 \in B^{E_I}(W_*, \mathbf{a}^k, j)$, it follows that $a^k_{i_2} < a^k_{i_1}$.

Obviously the conditions (i) and (ii) of Definition 3.7 are equivalent with the following facts (i′) and (ii′), respectively:

(i′) $\forall i \in \mathrm{N}$, there is $j \in \mathrm{M}$, satisfying $L_{ij}(\mathcal{A},\mathcal{B}) \cup G_{ij}(\mathcal{A},\mathcal{B}) = \emptyset$;

(ii′) $\forall j \in \mathrm{M}$, there is $i \in \mathrm{N}$, so that $L_{ij}(\mathcal{A},\mathcal{B}) \cup G_{ij}(\mathcal{A},\mathcal{B}) = \emptyset$.

Learning algorithm (3.45) is efficient, which can be demonstrated by the following result.

Theorem 3.13 *Suppose a fuzzy pattern pair family as*

$$(\mathcal{A},\,\mathcal{B}) = \left\{(\mathbf{a}^k, \mathbf{b}^k)\middle|\mathbf{a}^k = (a^k_1, ..., a^k_n),\ \mathbf{b}^k = (b^k_1, ..., b^k_m), k \in \mathrm{P}\right\}$$

is co-existed. Moreover, $\forall i \in \mathrm{N}$, $j \in \mathrm{M}$, $k \in \mathrm{P}$, *the following conditions hold:*

(i) $0 < a^k_i < 1$, $0 < b^k_j < 1$;

(ii) $(\mathcal{A},\mathcal{B})$ *is co-existed.*

In (3.32) choose $W = W_*$. *Then* $\forall k \in \mathrm{P}$, $(\mathbf{a}^k, \mathbf{b}^k)$ *is an attractor of the system, and the attractive basin* $A^{\mathbf{a}}_{W_*}(\mathbf{a}^k, \mathbf{b}^k) \times A^{\mathbf{b}}_{W_*}(\mathbf{a}^k, \mathbf{b}^k)$ *is non-degenerate.*

Proof. Give $k \in \mathrm{P}$, and in (3.32) let $W = W_*$. For any $j \in \mathrm{M}$, since $(\mathcal{A},\,\mathcal{B})$ is co-existed, we have, $\{i \in \mathrm{N} | L_{ij}(\mathcal{A},\mathcal{B}) \cup G_{ij}(\mathcal{A},\mathcal{B}) = \emptyset\} \neq \emptyset$. Also since $\forall i' \in \{i \in \mathrm{N} | L_{ij}(\mathcal{A},\mathcal{B}) \cup G_{ij}(\mathcal{A},\mathcal{B}) = \emptyset\}$, we can imply, $\forall k' \in \mathrm{P}$, $a^{k'}_{i'} = b^{k'}_j$. So by (3.45) it follows that

$$\bigvee_{i|L_{ij}(\mathcal{A},\mathcal{B})\cup G_{ij}(\mathcal{A},\mathcal{B})=\emptyset} \{a^k_i \wedge w^*_{ij}\} = b^k_j \wedge w_{i_0 j} = b^k_j \wedge \Big(\bigvee_{k'\in\mathrm{P}} \{b^{k'}_j\}\Big) = b^k_j. \tag{3.46}$$

On the other hand, if $G_{ij}(\mathcal{A},\mathcal{B}) \neq \emptyset$, then by $k \in G_{ij}(\mathcal{A},\mathcal{B})$ we can conclude that

$$a^k_i \wedge w^*_{ij} = a^k_i \wedge \zeta(i,j) \leq a^k_i \wedge \Big(\bigwedge_{k'\in G_{ij}(\mathcal{A},\mathcal{B})} \{b^{k'}_j\}\Big) \leq a^k_i \wedge b^k_j = b^k_j.$$

Because $k \notin G_{ij}(\mathcal{A},\mathcal{B})$ can ensure that $a_i^k \leq b_j^k$, we get, $a_i^k \wedge w_{ij}^* \leq b_j^k$. Thus, $\bigvee\limits_{i|G_{ij}(\mathcal{A},\mathcal{B})\neq\emptyset} \{a_i^k \wedge w_{ij}^*\} \leq b_j^k$. Similarly we have, $\bigvee\limits_{i|L_{ij}(\mathcal{A},\mathcal{B})\neq\emptyset} \{a_i^k \wedge w_{ij}^*\} \leq b_j^k$. Therefore

$$\bigvee_{i|L_{ij}(\mathcal{A},\mathcal{B})\cup G_{ij}(\mathcal{A},\mathcal{B})\neq\emptyset} \{a_i^k \wedge w_{ij}^*\}\leq b_j^k.$$

Considering (3.46) we imply, $\bigvee\limits_{i\in \mathrm{N}} \{a_i^k \wedge w_{ij}^*\} = b_j^k$ $(j \in \mathrm{M})$. With the same reason, the fact $\bigvee\limits_{j\in \mathrm{M}} \{b_j^k \wedge w_{ij}^*\} = a_i^k$ $(i \in \mathrm{N})$ holds. So $(\mathbf{a}^k, \mathbf{b}^k)$ is an attractor of (3.32).

Now let us show, $(W_*, \mathbf{a}^k, \mathbf{b}^k)$ satisfies GE condition. Since $(\mathcal{A}, \mathcal{B})$ is co-existed, it follows that $\forall j \in \mathrm{M}$, there is $i_1 \in \mathrm{N}$, so that $k \in E_{\mathcal{B}}^{\zeta}(i_1, j) \cup E_{\mathcal{B}}(j)$. If $k \in E_{\mathcal{B}}(j)$, then $\forall k' \in \mathrm{P}$, $b_j^{k'} \leq b_j^k$. By the assumption we have

$$\{l \in \mathrm{N}|G_{lj}(\mathcal{A},\mathcal{B}) \cup L_{lj}(\mathcal{A},\mathcal{B}) = \emptyset\}\neq \emptyset.$$

So let $i_0 \in \{l \in \mathrm{N}|G_{lj}(\mathcal{A},\mathcal{B}) \cup L_{lj}(\mathcal{A},\mathcal{B}) = \emptyset\}$. Thus

$$w_{i_0 j} = \bigvee_{k'\in \mathrm{P}} \{a_{i_0}^{k'}\} = \bigvee_{k'\in \mathrm{P}} \{b_j^{k'}\} = b_j^k.$$

That is, $j \in B^{E_J}(W_*, \mathbf{b}^k, i_0)$. Using (3.43) we get

$$k \in E_{\mathcal{B}}^{\zeta}(i_1, j), \Longrightarrow G_{i_1 j}(\mathcal{A},\mathcal{B}) \cup L_{i_1 j}(\mathcal{A},\mathcal{B}) \neq \emptyset.$$

So (3.45) and (3.43) imply that $b_j^k = \zeta(i_1, j) = w_{i_1 j}$, i.e. $j \in B^{E_J}(W_*, \mathbf{b}^k, i_1)$. Hence for $j \in \mathrm{M}$, there is $i \in \mathrm{N}$, so that $j \in B^{E_J}(W_*, \mathbf{b}^k, i)$. On the other hand, $\forall j_1 \in B^{E_J}(W_*, \mathbf{b}^k, i)$, $j_2 \in B^{G_J}(W_*, \mathbf{b}^k, i)$, by the condition (ii), $b_{j_2}^k > b_{j_1}^k$. Using the same steps we can conclude that

(1′) $\forall i \in \mathrm{N}$, there is $j \in \mathrm{M}$, so that $i \in B^{E_I}(W_*, \mathbf{b}^k, j)$;

(2′) $\forall i_1 \in B^{E_I}(W_*, \mathbf{b}^k, j)$, $i_2 \in B^{G_I}(W_*, \mathbf{b}^k, j), \Longrightarrow a_{i_2}^k > a_{i_1}^k$.

So $(W_*, \mathbf{a}^k, \mathbf{b}^k)$ satisfies GE condition. $\forall(\mathbf{x}, \mathbf{y}) \in A_{W_*}^{\mathbf{a}}(\mathbf{a}^k, \mathbf{b}^k) \times A_{W_*}^{\mathbf{b}}(\mathbf{a}^k, \mathbf{b}^k)$, By Theorem 3.12, $(\mathbf{x}, \mathbf{y})$ converges to $(\mathbf{a}^k, \mathbf{b}^k)$ with one iteration. Thus, $A_{W_*}^{\mathbf{a}}(\mathbf{a}^k, \mathbf{b}^k) \times A_{W_*}^{\mathbf{b}}(\mathbf{a}^k, \mathbf{b}^k)$ is the attractive basin of $(\mathbf{a}^k, \mathbf{b}^k)$. And by the assumption and Remark 3.2 $A_{W_*}^{\mathbf{a}}(\mathbf{a}^k, \mathbf{b}^k) \times A_{W_*}^{\mathbf{b}}(\mathbf{a}^k, \mathbf{b}^k)$ is non-degenerate. □

From the proof of Theorem 3.13 the following conclusion is trivial.

Remark 3.3 Let $(\mathcal{A}, \mathcal{B})$ be co-existed. Then $\forall k \in \mathrm{P}$, $\forall j \in \mathrm{M}$, there is $i \in \mathrm{N}$, so that $w_{ij}^* = b_j^k$; Symmetrically $\forall i \in \mathrm{N}$, $\forall k \in \mathrm{P}$, there is $j \in \mathrm{M}$, satisfying $w_{ij}^* = a_i^k$.

3.4.2 A simulation example

Let us in the following explain the realizing steps of the analytic learning algorithm (3.45), and therefore demonstrate its efficiency. To this end let N =

{1, 2, 3, 4, 5, 6}, M = {1, 2, 3, 4}, P = {1, 2, 3}. By Table 3.13 we give the fuzzy pattern pair family $(\mathcal{A}, \mathcal{B}) = \{(\mathbf{a}^k, \mathbf{b}^k) | k \in \mathrm{P}\}$.

Table 3.13 Fuzzy pattern pair family

k	$\mathbf{a}^k$	$\mathbf{b}^k$
1	(0.4 0.4 0.6 0.5 0.4 0.6)	(0.6 0.5 0.4 0.4)
2	(0.3 0.3 0.4 0.5 0.5 0.4)	(0.4 0.5 0.3 0.5)
3	(0.4 0.4 0.3 0.3 0.3 0.3)	(0.3 0.3 0.4 0.3)

With the following steps we may realize the learning algorithm (3.45):

Step 1. Put $k = 0$; and for $i \in \mathrm{N}$, $j \in \mathrm{M}$, calculate $L_{ij}(\mathcal{A}, \mathcal{B})$, $G_{ij}(\mathcal{A}, \mathcal{B})$, consequently determine $\zeta(i, j)$;

Step 2. For $i \in \mathrm{N}, j \in \mathrm{M}$, calculate $E^{\zeta}_{\mathcal{A}}(i, j)$, $E^{\zeta}_{\mathcal{B}}(i, j)$, $E_{\mathcal{A}}(i)$ and $E_{\mathcal{B}}(j)$;

Step 3. Discriminate whether $(\mathcal{A}, \mathcal{B})$ is co-existed or not, that is, verify the following conditions to hold or not:

(1′) $\forall j \in \mathrm{M}$, $\bigcup\limits_{i \in \mathrm{N}} E^{\zeta}_{\mathcal{B}}(i, j) \cup E_{\mathcal{B}}(j) = \mathrm{P}$, and $L_{ij}(\mathcal{A}, \mathcal{B}) \cup G_{ij}(\mathcal{A}, \mathcal{B}) = \emptyset$ holds for at least one $i \in \mathrm{N}$;

(2′) $\forall i \in \mathrm{N}$, $\bigcup\limits_{j \in \mathrm{M}} E^{\zeta}_{\mathcal{A}}(i, j) \cup E_{\mathcal{A}}(i) = \mathrm{P}$, and $L_{ij}(\mathbf{a}, \mathcal{B}) \cup G_{ij}(\mathcal{A}, \mathcal{B}) = \emptyset$ holds for at least one $j \in \mathrm{M}$.

If yes go to the following step, otherwise stop the procedure;

Step 4. By (3.45) to determine W_*, and let $k = k + 1$;

Step 5. Discriminate whether $(W_*, \mathbf{a}^k, \mathbf{b}^k)$ satisfies GE condition, if yes, go to the following step; otherwise go to step 4;

Step 6. For $\mathbf{a}^k$, $\mathbf{b}^k$, W_* we compute the sets: $B_0^{Ge_I}(W_*, \mathbf{b}^k)$, and

$$B_0^{E_I}(W_*, \mathbf{b}^k), B_0^{G_I}(W_*, \mathbf{b}^k),\ B_0^{Ge_J}(W_*, \mathbf{a}^k),\ B_0^{E_J}(W_*, \mathbf{a}^k),\ B_0^{G_J}(W_*, \mathbf{a}^k);$$

Step 7. Use (3.38) (3.39) to compute $A^{\mathbf{a}}_{W_*}(\mathbf{a}^k, \mathbf{b}^k) \times A^{\mathbf{b}}_{W_*}(\mathbf{a}^k, \mathbf{b}^k)$, the attractive basin of $(\mathbf{a}^k, \mathbf{b}^k)$.

For $(\mathcal{A}, \mathcal{B})$ shown as Table 3.13, easily we can show that the conditions in above steps hold. So we can establish the connection weight matrix $W_* = (w^*_{ij})_{6\times 4}$ of (3.32) as

$$W_*^{\mathrm{T}} = (w^*_{ji})_{4\times 6} = \begin{pmatrix} 0.3 & 0.3 & 0.6 & 0.4 & 0.4 & 0.6 \\ 0.3 & 0.3 & 0.4 & 0.5 & 0.4 & 0.4 \\ 0.4 & 0.4 & 0.3 & 0.3 & 0.3 & 0.3 \\ 0.3 & 0.3 & 0.4 & 0.4 & 0.5 & 0.4 \end{pmatrix}$$

The fuzzy pattern pair $(\mathbf{a}^k, \mathbf{b}^k)$ $(k = 1, 2, 3)$ is the attractor of (3.32). To compute the attractive basin $A^{\mathbf{a}}_{W_*}(\mathbf{a}^k, \mathbf{b}^k) \times A^{\mathbf{b}}_{W_*}(\mathbf{a}^k, \mathbf{b}^k)$ $(k = 1, 2, 3)$, we first calculate the following sets: $B_0^{Ge_I}(W_*, \mathbf{b}^k)$, $B_0^{Ge_J}(W_*, \mathbf{a}^k)$, $B_0^{G_I}(W_*, \mathbf{b}^k)$, and $B_0^{G_J}(W_*, \mathbf{a}^k)$, $B_0^{E_I}(W_*, \mathbf{b}^k)$, $B_0^{E_J}(W_*, \mathbf{a}^k)$, as shown in Table 3.14.

Table 3.14 $B_0^{Ge_I}(W_*, \mathbf{b}^k), ..., B_0^{E_J}(W_*, \mathbf{a}^k)$

	$(\mathbf{a}^1, \mathbf{b}^1)$	$(\mathbf{a}^2, \mathbf{b}^2)$	$(\mathbf{a}^3, \mathbf{b}^3)$
$B_0^{Ge_I}(W_*, \mathbf{b}^k)$	$\emptyset$	{3,6}	$\emptyset$
$B_0^{Ge_J}(W_*, \mathbf{a}^k)$	$\emptyset$	{1}	$\emptyset$
$B_0^{G_I}(W_*, \mathbf{b}^k)$	{5}	{1,2}	{3,4,5,6}
$B_0^{G_J}(W_*, \mathbf{a}^k)$	{4}	{3}	{1,2,4}
$B_0^{E_I}(W_*, \mathbf{b}^k)$	{1,2,3,4,6}	{4,5}	{1,2}
$B_0^{E_J}(W_*, \mathbf{a}^k)$	{1,2,3}	{2,4}	{3}

For $i \in \mathrm{N}$, $j \in \mathrm{M}$, $k \in \mathrm{P}$, we determine $d^1_{\mathbf{a}i}(W_*, \mathbf{b}^k)$, $d^2_{\mathbf{a}i}(W_*, \mathbf{b}^k)$, and $d^1_{\mathbf{b}j}(W_*, \mathbf{a}^k)$, $d^2_{\mathbf{b}j}(W_*, \mathbf{a}^k)$ by using (3.37). And compute the attractive basin $A^{\mathbf{a}}_{W_*}(\mathbf{a}^k, \mathbf{b}^k) \times A^{\mathbf{b}}_{W_*}(\mathbf{a}^k, \mathbf{b}^k)$ $(k \in \mathrm{P})$ of $(\mathbf{a}^k, \mathbf{b}^k)$, where

$$\begin{cases} A^{\mathbf{a}}_{W_*}(\mathbf{a}^1, \mathbf{b}^1) = ([0.4,1]\times[0.4,1]\times[0.6,1]\times[0.5,1]\times[0,0.4]\times[0.6,1], \\ A^{\mathbf{b}}_{W_*}(\mathbf{a}^1, \mathbf{b}^1) = [0.6,1]\times[0.5,1]\times[0.4,1]\times[0,0.4]; \end{cases}$$

$$\begin{cases} A^{\mathbf{a}}_{W_*}(\mathbf{a}^2, \mathbf{b}^2) = [0,0.3]\times[0,0.3]\times[0.3,0.4]\times[0.5,1]\times[0.5,1]\times[0.3,0.4], \\ A^{\mathbf{b}}_{W_*}(\mathbf{a}^2, \mathbf{b}^2) = [0.3,0.4]\times[0.5,1]\times[0,0.3]\times[0.5,1]; \end{cases}$$

$$\begin{cases} A^{\mathbf{a}}_{W_*}(\mathbf{a}^3, \mathbf{b}^3) = [0.4,1]\times[0.4,1]\times[0,0.3]\times[0,0.3]\times[0,0.3]\times[0,0.3], \\ A^{\mathbf{b}}_{W_*}(\mathbf{a}^3, \mathbf{b}^3) = [0,0.3]\times[0,0.3]\times[0.4,1]\times[0,0.3]. \end{cases}$$

There are still two important and meaningful problems to be solved. First, the condition of the fuzzy pattern pair family $(\mathcal{A}, \mathcal{B})$ being co-existed should be improved so that the learning algorithms related can be applied more widely; Second, develop a novel learning algorithm for W to enlarge the attractive basins related, and therefore improve fault–tolerance of the system (3.32).

3.4.3 Optimal fault–tolerance

Let us next generalize (3.38) (3.39) to improve fault–tolerance of (3.32). To this end we use (3.2) to enlarge the attractive basins determined by (3.38) (3.39) through taking the function $\varpi(\cdot)$ as a bridge. Let

$$B_I^{E_I}(W, \mathbf{b}) = \{i \in \mathrm{N} | B^{E_J}(W, \mathbf{b}, i) = M\},$$

$$B_I^{E_J}(W, \mathbf{a}) = \{j \in \mathrm{M} | B^{E_I}(W, \mathbf{a}, j) = \mathrm{N}\}.$$

$$i_e = \begin{cases} \min\{i | i \in B_I^{E_I}(W, \mathbf{b})\}, & B_I^{E_I}(W, \mathbf{b}) \neq \emptyset; \\ n+1, & B_I^{E_I}(W, \mathbf{b}) = \emptyset. \end{cases}$$

$$j_e = \begin{cases} \min\{j | j \in B_I^{E_J}(W, \mathbf{a})\}, & B_I^{E_J}(W, \mathbf{a}) \neq \emptyset; \\ m+1, & B_I^{E_J}(W, \mathbf{a}) = \emptyset. \end{cases}$$

Suppose (W, $\mathbf{a}$, $\mathbf{b}$) satisfies GE condition. For simplicity we also denote

$$A_W^{\mathbf{a}}(\mathbf{a},\mathbf{b}) = \Big\{(x_1,...,x_n) \in [0,\ 1]^n \Big| x_i \in \big[0,\ d_{\mathbf{a}i}^2(W,\mathbf{b})\big]\ (i \in B_0^{G_I}(W,\mathbf{b})),$$
$$x_i \in \big[d_{\mathbf{a}i}^1(W,\mathbf{b}) \cdot \varpi(B_I^{E_I}(W,\mathbf{b})),\ 1\big]\ (i \in B_I^{E_I}(W,\mathbf{b}) \setminus \{i_e\}),$$
$$x_i \in \big[d_{\mathbf{a}i}^1(W,\mathbf{b}),\ d_{\mathbf{a}i}^2(W,\mathbf{b})\big]\ (i \in B_0^{Ge_I}(W,\mathbf{b})),$$
$$x_{i_e} \in \big[d_{\mathbf{a}i_e}^1(W,\mathbf{b}),\ 1\big],\ x_i \in [0,\ 1]\,(\text{otherwise})\Big\}; \tag{3.47}$$

$$A_W^{\mathbf{b}}(\mathbf{a},\mathbf{b}) = \Big\{(y_1,...,y_m) \in [0,\ 1]^m \Big| y_j \in \big[0,\ d_{\mathbf{b}j}^2(W,\mathbf{a})\big]\ (j \in B_0^{G_J}(W,\mathbf{a})),$$
$$y_j \in \big[d_{\mathbf{b}j}^1(W,\mathbf{a}) \cdot \varpi(B_I^{E_J}(W,\mathbf{a})),\ 1\big]\ (j \in B_I^{E_J}(W,\mathbf{a}) \setminus \{j_e\}),$$
$$y_j \in \big[d_{\mathbf{b}j}^1(W,\mathbf{a}),\ d_{\mathbf{b}j}^2(W,\mathbf{a})\big]\ (j \in B_0^{Ge_J}(W,\mathbf{a})\}),$$
$$y_{j_e} \in \big[d_{\mathbf{b}j_e}^1(W,\mathbf{a}),\ 1\big],\ y_j \in \big[0,\ 1\big](\text{otherwise})\big\}. \tag{3.48}$$

Theorem 3.14 *Let W be the connection weight matrix of (3.32), and the fuzzy pattern pair $(\mathbf{a},\mathbf{b})$ be an attractor of (3.32). Let (W, $\mathbf{a}$, $\mathbf{b}$) satisfy GE condition. Then for any $(\mathbf{x},\mathbf{y}) \in A_W^{\mathbf{a}}(\mathbf{a},\mathbf{b}) \times A_W^{\mathbf{b}}(\mathbf{a},\mathbf{b})$, taking it it as an initial fuzzy pattern, the system (3.32) converges with one iteration to $(\mathbf{a},\mathbf{b})$.*

Proof. Give $i \in \mathrm{N}$, $j \in \mathrm{M}$. Since (W, $\mathbf{a}$, $\mathbf{b}$) satisfies GE condition, we get by (3.37) that

$$d_{\mathbf{a}i}^1(W,\mathbf{b}) < d_{\mathbf{a}i}^2(W,\mathbf{b}),\ \ d_{\mathbf{b}j}^1(W,\mathbf{a}) < d_{\mathbf{b}j}^2(W,\mathbf{a}). \tag{3.49}$$

For fuzzy pattern pair $(\mathbf{x},\mathbf{y}) \in A_W^{\mathbf{a}}(\mathbf{a},\mathbf{b}) \times A_W^{\mathbf{b}}(\mathbf{a},\mathbf{b})$: $\mathbf{x} = (x_1,...,x_n)$, $\mathbf{y} = (y_1,...,y_m)$, if $B_I^{E_I}(W,\mathbf{b}) \neq \emptyset$, then by $i_e \in B_I^{E_I}(W,\mathbf{b})$, we have $B^{E_J}(W,\mathbf{b},i_e) = M$. So by (3.47) it follows that

$$\begin{aligned}\bigvee_{k\in B_I^{E_I}(W,\mathbf{b})\cup B_0^{Ge_I}(W,\mathbf{b})} \{x_k \wedge w_{kj}\} &\geq \bigvee_{k\in B_I^{E_I}(W,\mathbf{b})} \{x_k \wedge w_{kj}\} \geq x_{i_e} \wedge w_{i_e j}\\ &\geq d_{\mathbf{a}i_e}^1(W,\mathbf{b}) \wedge w_{i_e j} \geq \Big(\bigvee_{k\in B^{E_J}(W,\mathbf{b},i_e)} \{b_k\}\Big) \wedge w_{i_e j}\\ &\geq \Big(\bigvee_{k\in \mathrm{M}} \{b_k\}\Big) \wedge w_{i_e j} \geq b_j.\end{aligned} \tag{3.50}$$

If $B_I^{E_I}(W,\mathbf{b}) = \emptyset$, then $\varpi(B_I^{E_I}(W,\mathbf{b})) = 1$. Since ($W$, $\mathbf{b}$) satisfies GE condition, we choose $i_0 \in \mathrm{N}$, so that $j \in B^{E_J}(W,\mathbf{b},i_0)$. Therefore, $i_0 \in B_0^{Ge_I}(W,\mathbf{b}) \cup B_I^{E_I}(W,\mathbf{b})$. Considering (3.46) and $j \in B^{E_J}(W,\mathbf{b},i_0)$, we can conclude that

$$\bigvee_{k\in B_I^{E_I}(W,\mathbf{b})\cup B_0^{Ge_I}(W,\mathbf{b})} \{x_k \wedge w_{kj}\} \geq \bigvee_{k\in B_I^{E_I}(W,\mathbf{b})\cup B_0^{Ge_I}(W,\mathbf{b})} \big\{d_{\mathbf{a}k}^1(W,\mathbf{b}) \wedge w_{kj}\big\}$$

$$\geq d^1_{\mathbf{a}i_0}(W,\mathbf{b})\wedge w_{i_0j}\geq\Big(\bigvee_{k\in B^{E_J}(W,\mathbf{b},i_0)}\{b_k\}\Big)\wedge w_{i_0j}\geq b_j.$$

In summary, $\bigvee_{k\in B_0^{Ge_I}(W,\mathbf{b})\cup B_I^{E_I}(W,\mathbf{b})}\{x_k\wedge w_{kj}\}\geq b_j$. Thus

$$\begin{aligned}\bigvee_{k\in \mathrm{N}}\{x_k\wedge w_{kj}\}&=\Big(\bigvee_{k\in B_0^{Ge_I}(W,\mathbf{b})}\{x_k\wedge w_{kj}\}\Big)\vee\Big(\bigvee_{k\in B_0^{G_I}(W,\mathbf{b})}\{x_k\wedge w_{kj}\}\Big)\vee\\ &\vee\Big(\bigvee_{k\in B_I^{E_I}(W,\mathbf{b})}\{x_k\wedge w_{kj}\}\Big)\vee\Big(\bigvee_{k\notin B_0^{Ge_I}(W,\mathbf{b})\cup B_I^{E_I}(W,\mathbf{b})\cup B_0^{G_I}(W,\mathbf{b})}\{x_k\wedge w_{kj}\}\Big)\\ &\geq\Big(\bigvee_{k\in B_I^{E_I}(W,\mathbf{b})\cup B_0^{Ge_I}(W,\mathbf{b})}\{x_k\wedge w_{kj}\}\Big)\geq b_j.\end{aligned}\tag{3.51}$$

On the other hand, let

$$l_1(q)\triangleq\bigvee_{k\in B_0^{G_I}(W,\mathbf{b})}\{w_{kj}\wedge d^2_{\mathbf{a}k}(W,\mathbf{b})\}=\bigvee_{k\in B_0^{G_I}(W,\mathbf{b})}\Big\{w_{kj}\wedge\Big(\bigwedge_{k'\in B^{G_J}(W,\mathbf{b},k)}\{b_{k'}\}\Big)\Big\}.$$

By the fact that $j\in B^{G_J}(W,\mathbf{b},k)$, it follows that $\bigwedge_{k\in B^{G_J}(W,\mathbf{b},i)}\{b_k\}\leq b_j$. And $j\notin B^{G_J}(W,\mathbf{b},k)$ may imply, $w_{kj}\leq b_j$. So $\bigvee_{k\in B_0^{G_I}(W,\mathbf{b})}\{x_k\wedge w_{kj}\}\leq l_1(q)\leq b_j$. With the same reason we have

$$b_j\geq l_2(q)\triangleq\bigvee_{k\in B_I^{E_I}(W,\mathbf{b})}\{w_{kj}\wedge 1\}\geq\bigvee_{k\in B_I^{E_I}(W,\mathbf{b})}\{w_{kj}\wedge x_k\};$$

$$b_j\geq l_3(q)\triangleq\bigvee_{k\in B_0^{Ge_I}(W,\mathbf{b})}\{w_{kj}\wedge d^2_{\mathbf{a}k}(W,\mathbf{b})\}\geq\bigvee_{k\in B_0^{Ge_I}(W,\mathbf{b})}\{w_{kj}\wedge x_k\}.$$

Therefore, the following fact holds:

$$\begin{aligned}\bigvee_{k\in \mathrm{N}}\{x_k\wedge w_{kj}\}&\leq l_1(q)\vee l_2(q)\vee l_3(q)\vee\\ &\vee\Big(\bigvee_{k\notin B_I^{E_I}(W,\mathbf{b})\cup B_0^{G_I}(W,\mathbf{b})\cup B_0^{Ge_I}(W,\mathbf{b})}\{x_k\wedge w_{kj}\}\Big)\leq b_j.\end{aligned}\tag{3.52}$$

Thus, using (3.51) and (3.52) we get, $\bigvee_{k\in \mathrm{N}}\{w_{kj}\wedge x_k\}=b_j$ $(j\in \mathrm{M})$. Using the similar steps we can show, $\bigvee_{k\in \mathrm{M}}\{y_k\wedge w_{ik}\}=a_i$ for $i\in \mathrm{N}$. Hence $\mathbf{x}\circ W=\mathbf{b}$, $\mathbf{y}\circ W^T=\mathbf{a}$. That, $(\mathbf{x},\mathbf{y})$ converges with one iteration to $(\mathbf{a},\mathbf{b})$. □

Remark 3.4 Under the conditions of Theorem 3.14, if $\forall i\in \mathrm{N},\ j\in \mathrm{M},\ 0<a_i<1,\ 0<b_j<1$, then by (3.47) (3.48) we can show, the attractive basin $A^{\mathbf{a}}_W(\mathbf{a},\mathbf{b})\times A^{\mathbf{b}}_W(\mathbf{a},\mathbf{b})$ is non-degenerate.

Now let us enlarge the attractive basins of the attractors of (3.32) by adjusting W suitably.

Theorem 3.15 *Suppose* $(\mathbf{a},\mathbf{b})$ *is an attractor of (3.32) when the connection weight matrix* W *is* W_1 *and* W_2*, respectively. Moreover,* $W_1 \subset W_2$*, and* W_1 *is defined by substituting some nonzero elements respectively of* W_2 *for zero.* $(W_1,\ \mathbf{a},\ \mathbf{b})$ *satisfies GE condition, and so does* $\mathbf{a},\mathbf{b},W_2$*. Then*

$$A_{W_2}^{\mathbf{a}}(\mathbf{a},\mathbf{b}) \times A_{W_2}^{\mathbf{b}}(\mathbf{a},\mathbf{b}) \subset A_{W_1}^{\mathbf{a}}(\mathbf{a},\mathbf{b}) \times A_{W_1}^{\mathbf{b}}(\mathbf{a},\mathbf{b}).$$

Proof. By the assumption and Theorem 3.14, $A_{W_1}^{\mathbf{a}}(\mathbf{a},\mathbf{b}) \times A_{W_1}^{\mathbf{b}}(\mathbf{a},\mathbf{b})$ and $A_{W_2}^{\mathbf{a}}(\mathbf{a},\mathbf{b}) \times A_{W_2}^{\mathbf{b}}(\mathbf{a},\mathbf{b})$ are the attractive basins of $(\mathbf{a},\mathbf{b})$ when W being W_1 and W_2, respectively. To prove our conclusion it suffices by (3.47) and (3.48) to show, $\forall i \in \mathrm{N},\ j \in \mathrm{M}$, it follows that

$$d_{\mathbf{a}i}^1(W_1,\mathbf{b}) \le d_{\mathbf{a}i}^1(W_2,\mathbf{b}),\ \ d_{\mathbf{a}i}^2(W_2,\mathbf{b}) \le d_{\mathbf{a}i}^2(W_1,\mathbf{b}). \tag{3.53}$$

$$d_{\mathbf{b}j}^1(W_1,\mathbf{a}) \le d_{\mathbf{b}j}^1(W_2,\mathbf{a}),\ \ d_{\mathbf{b}j}^2(W_2,\mathbf{a}) \le d_{\mathbf{b}j}^2(W_1,\mathbf{a}). \tag{3.54}$$

In fact, Suppose $W_k = (w_{ij}^k)_{n\times n}$ $(k = 1,2)$. Then by the assumption, $\forall i \in \mathrm{N},\ j \in \mathrm{M}$, either $w_{ij}^1 = 0$ or, $w_{ij}^1 = w_{ij}^2$. So for $i \in \mathrm{N}$, we have, $B^{E_J}(W_1,\mathbf{b},i) \subset B^{E_J}(W_2,\mathbf{b},i)$. Thus

$$d_{\mathbf{a}i}^1(W_1,\mathbf{b}) = \bigvee_{k\in B^{E_J}(W_1,\mathbf{b},i)} \{b_k\} \le \bigvee_{k\in B^{E_J}(W_2,\mathbf{b},i)} \{b_k\} = d_{\mathbf{a}i}^1(W_2,\mathbf{b}).$$

The first part of (3.53) holds. On the other hand, $W_1 \subset W_2$ may imply, $\forall i \in \mathrm{N},\ B^{G_J}(W_1,\mathbf{b},i) \subset B^{G_J}(W_2,\mathbf{b},i)$. Hence

$$d_{\mathbf{a}i}^2(W_2,\mathbf{b}) = \bigwedge_{k\in B^{G_J}(W_2,\mathbf{b},i)} \{b_k\} \le \bigwedge_{k\in B^{G_J}(W_1,\mathbf{b},i)} \{b_k\} = d_{\mathbf{a}i}^2(W_1,\mathbf{b}),$$

i.e. the last part of (3.53) holds. So (3.53) is true. Similarly we can show, (3.54) holds. □

By Theorem 3.15, in the system (3.32), provided $(W,\ \mathbf{a},\ \mathbf{b})$ satisfies GE condition, the smaller the connection weight matrix W is, the larger the attractive basin of each attractor is. Such a fact can lead us to develop a novel learning algorithm for W so that the system (3.32) possesses optimal fault–tolerance. At first we improve (3.45) so that

(i) When $(\mathcal{A},\ \mathcal{B})$ satisfies some simple conditions, each fuzzy pattern pair in $(\mathcal{A},\ \mathcal{B})$ is attractor of (3.32);

(ii) W achieves its minimum value, and consequently (3.32) has optimal fault–tolerance.

To this end, we denote for $i \in \mathrm{N},\ j \in \mathrm{M}$

$$
\begin{aligned}
LA(i,j) &= \big\{j_1 \in \mathrm{M}\big| j_1 < j,\ G_{ij_1}(\mathcal{A},\mathcal{B})) \neq \emptyset, \text{ and } \zeta(i,j_1) = \zeta(i,j)\big\},\\
LB(i,j) &= \big\{i_1 \in \mathrm{N}\big| i_1 < i,\ L_{i_1 j}(\mathcal{A},\mathcal{B}) \neq \emptyset, \text{ and } \zeta(i_1,j) = \zeta(i,j)\big\},\\
DA(i,j) &= \Big\{i_1 \in \mathrm{N}\Big| i_1 < i,\ a_i^k = a_{i_1}^k = \bigvee_{k' \in \mathrm{P}} \{a_i^{k'}\}\Big\},\\
DB(i,j) &= \Big\{j_1 \in \mathrm{M}\Big| j_1 < j,\ b_j^k = b_{j_1}^k = \bigvee_{k' \in \mathrm{P}} \{b_j^{k'}\}\Big\}.
\end{aligned}
\tag{3.55}
$$

Now we can determine the connection weight matrix $W = W_0 = (w_{ij}^0)_{n\times m}$ of (3.32) as follows:

$$
w_{ij}^0 = \begin{cases} \zeta(i,j) \wedge \big\{\varpi\big(LA(i,j)\big)\wedge\varpi\big(LB(i,j)\big)\big\}, & L_{ij}(\mathcal{A},\mathcal{B}) \cup G_{ij}(\mathcal{A},\mathcal{B}) \neq \emptyset,\\ \Big(\bigvee_{k\in\mathrm{P}} \{a_i^k\}\Big)\wedge\big\{\varpi\big(DB(i,j)\big)\wedge\varpi\big(DA(i,j)\big)\big\}, & L_{ij}(\mathcal{A},\mathcal{B}) \cup G_{ij}(\mathcal{A},\mathcal{B}) = \emptyset. \end{cases}
\tag{3.56}
$$

Theorem 3.16 *Suppose* $(\mathbf{a},\mathbf{b})$ *is an arbitrary fuzzy pattern pair,* $\mathbf{a} = (a_1,...,a_n)$, $\mathbf{b} = (b_1,...,b_m)$, *moreover,* $\forall i \in \mathrm{N}, j \in \mathrm{M},\ a_i \cdot b_j \neq 0$. *Fuzzy matrices* W_*, W_0 *are defined by (3.45) and (3.56) respectively.* $(W_*,\ \mathbf{a},\ \mathbf{b})$ *satisfies GE condition. Then* $(W_0,\ \mathbf{a},\ \mathbf{b})$ *satisfies GE condition also.*

Proof. By the respective definitions of (3.45) (3.56), obviously we have

$$
\forall i \in \mathrm{N},\ j \in \mathrm{M},\ w_{ij}^0 \neq 0, \Longrightarrow w_{ij}^0 = w_{ij}^*. \tag{3.57}
$$

For each $i \in \mathrm{N}$, we choose arbitrarily $j_1, j_2 \in \mathrm{M}$ satisfying the conditions: $j_1 \in B^{E_J}(W_0,\mathbf{b},i)$, $j_2 \in B^{G_J}(W_0,\mathbf{b},i)$. Then by the assumption we get, $w_{ij_1}^0 = b_{j_1} > 0$, $w_{ij_2} = b_{j_2} > 0$. By (3.57) it follows that $w_{ij_1}^* = b_{j_1}$, $w_{ij_2}^* > b_{j_2}$. So $j_1 \in B^{E_J}(W_*,\mathbf{b},i)$, $j_2 \in B^{G_J}(W_*,\mathbf{b},i)$. Since $(W_*,\ \mathbf{a},\ \mathbf{b})$ satisfies GE condition, $b_{j_1} < b_{j_2}$. For arbitrary $j \in \mathrm{M}$, also using the assumption that $(W_*,\ \mathbf{a},\ \mathbf{b})$ satisfies GE condition, we imply, there is $i \in \mathrm{N}$, so that $j \in B^{E_J}(W_*,\mathbf{b},i)$, i.e. $w_{ij}^* = b_j > 0$. Using (3.57) again, we obtain $w_{ij}^0 = b_j$. hence $j \in B^{E_J}(W_0,\mathbf{b},i)$. With the same steps we can show, for each $j \in \mathrm{M}$, and $\forall i_1 \in B^{E_I}(W_0,\mathbf{a},j)$, $\forall i_2 \in B^{G_I}(W_0,\mathbf{a},j)$, it follows that $a_{i_1} < a_{i_2}$. Moreover, $\forall i \in \mathrm{N}$, $\exists j \in \mathrm{M}$, satisfying $i \in B^{E_I}(W_0,\ \mathbf{a},j)$. Thus, $(W_0,\ \mathbf{a},\mathbf{b})$ satisfies GE condition. □

Considering the fact that $W_0 \subset W_*$, Theorem 3.15 and Theorem 3.16, we can get better fault–tolerance of the system (3.32) if we choose W_0 as the connection weight matrix rather than W_*. Furthermore, Similarly with Theorem 3.5, we may show the minimality of W_0 defined by (3.56) with the given conditions.

Remark 3.5 Suppose a fuzzy pattern pair family as

$$
(\mathcal{A},\ \mathcal{B}) = \big\{(\mathbf{a}^k,\mathbf{b}^k)\big|\mathbf{a}^k = (a_1^k,...,a_n^k),\ \mathbf{b}^k = (b_1^k,...,b_m^k),\ k \in \mathrm{P}\big\}
$$

is co-existed. $W_0 = (w_{ij}^0)_{n\times m}$ is defined by (3.56). Then W_0 is minimum among the fuzzy matrices W's which satisfy the following conditions:

(i) $\forall k \in$ P, $(\mathbf{a}^k, \mathbf{b}^k)$ is an attractor of W;

(ii) $\forall k \in$ P, $(W, \mathbf{a}^k, \mathbf{b}^k)$ satisfies GE condition.

By Theorem 3.15, Theorem 3.16 and Remark 3.5, if the connection weight matrix of (3.32) is W_0, then each fuzzy pattern pair in $(\mathcal{A}, \mathcal{B})$ is the attractor. Moreover, the attractive basins related is maximum, and therefore the system possesses good fault–tolerance.

3.4.4 An example

To show the advantage in fault–tolerance of the FBAM defined by W_0 over that by W_*, we in this subsection use the same example as in §3.4.2 to calculate the attractive basins related to W_0. So N $= \{1, ..., 6\}$, M $= \{1, 2, 3, 4\}$, and the fuzzy pattern pairs (A^1, B^1), (A^2, B^2), (A^3, B^3) are shown in Table 3.13.

It is easy to show that the conditions of Theorem 3.15, Theorem 3.16 and Remark 3.5 hold. Using (3.55) we can establish the connection weight matrix $W_0 = (w_{ij}^0)_{6\times 4}$ as follows:

$$W_0^{\mathrm{T}} = (w_{ji}^0)_{4\times 6} = \begin{pmatrix} 0.3 & 0 & 0.6 & 0.4 & 0 & 0 \\ 0.3 & 0 & 0.4 & 0.5 & 0 & 0 \\ 0.4 & 0 & 0.3 & 0 & 0 & 0 \\ 0.3 & 0 & 0.4 & 0 & 0.5 & 0 \end{pmatrix}$$

The fuzzy pattern pair $(\mathbf{a}^k, \mathbf{b}^k)$ for $k = 1, 2, 3$ is the attractor of (3.32) when $W = W_0$. To compute the attractive basin $A_W^{\mathbf{a}}(\mathbf{a}^k, \mathbf{b}^k) \times A_W^{\mathbf{b}}(\mathbf{a}^k, \mathbf{b}^k)$ of $(\mathbf{a}^k, \mathbf{b}^k)$ $(k = 1, 2, 3)$, using the same steps as in §3.4.2 we can calculate the following sets: $B_0^{Ge_I}(W, \mathbf{b})$, $B_0^{Ge_J}(W, \mathbf{a})$, $B_0^{G_I}(W, \mathbf{b})$, $B_0^{G_J}(W, \mathbf{a})$, $B_0^{E_I}(W, \mathbf{b})$ and $B_0^{E_J}(W, \mathbf{a})$ shown in Table 3.14. For $i \in$ N, $j \in$ M, $k \in$ P, we determine $d_{\mathbf{a}i}^1(W_*, \mathbf{b}^k)$, $d_{\mathbf{a}i}^2(W_*, \mathbf{b}^k)$, and $d_{\mathbf{b}j}^1(W_*, \mathbf{a}^k)$, $d_{\mathbf{b}j}^2(W_*, \mathbf{a}^k)$ by using (3.47). And compute the attractive basin $A_W^{\mathbf{a}}(\mathbf{a}^k, \mathbf{b}^k) \times A_W^{\mathbf{b}}(\mathbf{a}^k, \mathbf{b}^k)$ $(k \in$ P) of $(\mathbf{a}^k, \mathbf{b}^k)$, where

$$\begin{cases} A_W^{\mathbf{a}}(\mathbf{a}^1, \mathbf{b}^1) = ([0.4, 1] \times [0, 1] \times [0.6, 1] \times [0.5, 1] \times [0, 0.4] \times [0, 1], \\ A_W^{\mathbf{b}}(\mathbf{a}^1, \mathbf{b}^1) = [0.6, 1] \times [0.5, 1] \times [0.4, 1] \times [0, 0.4]; \end{cases}$$

$$\begin{cases} A_W^{\mathbf{a}}(\mathbf{a}^2, \mathbf{b}^2) = [0, 0.3] \times [0, 1] \times [0.3, 1] \times [0.5, 1] \times [0.5, 1] \times [0, 1], \\ A_W^{\mathbf{b}}(\mathbf{a}^2, \mathbf{b}^2) = [0.3, 0.4] \times [0.5, 1] \times [0, 0.3] \times [0.5, 1]; \end{cases}$$

$$\begin{cases} A_W^{\mathbf{a}}(\mathbf{a}^3, \mathbf{b}^3) = [0.4, 1] \times [0, 1] \times [0, 0.3] \times [0, 0.3] \times [0, 0.3] \times [0, 1], \\ A_W^{\mathbf{b}}(\mathbf{a}^3, \mathbf{b}^3) = [0, 0.3] \times [0, 0.3] \times [0.4, 1] \times [0, 0.3]. \end{cases}$$

From above respective attractive basins of $(\mathbf{a}^1, \mathbf{b}^1)$, $(\mathbf{a}^2, \mathbf{b}^2)$ and $(\mathbf{a}^3, \mathbf{b}^3)$ we can see they are larger than the corresponding ones respectively in §3.4.2. And so the fault–tolerance of the system (3.32) is improved by using W_0.

§3.5 Connection network of FBAM

By the preceding section we know, provided the connection weight matrices are chosen suitably, FBAM's may possess good fault–tolerance. In the section we suggest a novel method to study the transition laws of the states of the FBAM's, and give a further research on the attractors of the FBAM's. By the proposed approach here, some FBAM's with poor fault–tolerance may be discriminated. The tools to do these are the fuzzy row-restricted matrices, elementary memory and fuzzy homomorphism operators, etc. In the following we write also $\mathrm{N} = \{1, ..., n\}$, $\mathrm{M} = \{1, ..., m\}$, $m, n \geq 3$. If $\mathbf{x} \in [0, 1]^n$, denote $(\mathbf{x})_j \triangleq x_j$ the $j-$th component of $\mathbf{x}$.

3.5.1 Fuzzy row–restricted matrix

To study the attractors of FBAM's, we introduce the fuzzy row-restricted matrix and the fuzzy connection matrix related to a given fuzzy matrix. Considering $\mu_{n\times m}$ means the collection of all fuzzy matrices with n rows and m columns, we propose $R = (r_{ij}) \in \mu_{n\times m}$ and $L = (l_{ji}) \in \mu_{m\times n}$ to generalize (3.32) as follows:

$$\begin{cases} \mathbf{y}^k = \mathbf{x}^{k-1} \circ R \\ \mathbf{x}^k = \mathbf{y}^{k-1} \circ L \end{cases} \tag{3.58}$$

where $k = 1, 2, ...,$ is the iteration steps, (3.58) is called a generalized FBAM. Denote $\mathbf{x}^k = (x_1^k, ..., x_n^k)$, $\mathbf{y}^k = (y_1^k, ..., y_m^k)$, We rewrite the system (3.58) as

$$\begin{cases} y_j^k = \bigvee\limits_{i\in \mathrm{N}} \{x_i^{k-1} \wedge r_{ij}\} \ \ (j \in \mathrm{M}), \\ x_i^k = \bigvee\limits_{j\in \mathrm{M}} \{y_j^{k-1} \wedge l_{ji}\} \ \ (i \in \mathrm{N}). \end{cases} \tag{3.59}$$

Considering (3.58) or (3.59) is uniquely determined by the fuzzy matrices R, L, we also call (R, L) a generalized FBAM.

Definition 3.8 Suppose $\lambda \in [0, 1]$, $k, p, q \in \mathrm{N}$, and the fuzzy matrices $W = (w_{ij}) \in \mu_{n\times m}$. Define $W^{(1)} = \left(w_{ij}^1\right)$, $W^{(2)} = \left(w_{ij}^2\right) \in \mu_{n\times m}$, respectively as follows:

$$w_{ij}^1 = \begin{cases} w_{ij} & i \neq k \\ \lambda \wedge w_{ij} & i = k \end{cases} \qquad w_{ij}^2 = \begin{cases} w_{ij} & i \neq p, q \\ w_{pj} & i = q \\ w_{qj} & i = p \end{cases}$$

We call $W^{(1)}$ the $k-$multiple matrix of W, and $W^{(2)}$ the $p-q$ commutative matrix of W.

We denote by $\Delta = \left(\delta_{ij}\right)$ the identity matrix, i.e. $\delta_{ij} = 1$ $(i = j)$ and $\delta_{ij} = 0$ $(i \neq j)$. And by $\Delta_k(\lambda)$ denote the $k-$multiple matrix of the identity matrix Δ. Δ_{pq} means the fuzzy matrix obtained by interchanging p-th row and q-th row of Δ, i.e. Δ_{pq} is the $p-q$ commutative matrix of Δ. From now on, we call $\Delta_k(\lambda)$ and Δ_{pq} the fuzzy row-restricted matrices. For simplicity we omit the mark of the order of a fuzzy row-restricted matrix, since it is easy to know the order through the fuzzy matrix used for composition. For example, if $W \in \mu_{n\times m}$, by $\Delta_k(\lambda)\circ W$ we implies the order of $\Delta_k(\lambda)$ is n. It is trivial by Definition 3.8 to show that

Proposition 3.1 *Suppose $W^{(1)}$, $W^{(2)}$ are k-multiple matrix and $p-q$ commutative matrix of W, respectively. Then*
(i) $W^{(1)} = \Delta_k(\lambda)\circ W,\ W^{(2)} = \Delta_{pq}\circ W$;
(ii) $\forall \mathbf{x}\in[0,1]^n,\ (\mathbf{x}\circ\Delta_{pq})\circ\Delta_{pq} = \mathbf{x}$, *moreover*, $\Delta_{pq}\circ(\Delta_{pq}\circ W) = W$;
(iii) $\forall \mathbf{x}\in[0,1]^n,\ \lambda\in[0,1],\ \mathbf{x}\circ\Delta_k(\lambda)\subset\mathbf{x},\ \Delta_k(\lambda)\circ W\subset W.$

If the fuzzy matrix P can be represented as follows: $\Delta_1\circ\Delta_2\circ\cdots\circ\Delta_l$, where Δ_p $(p=1,...,l)$ is a fuzzy row-restricted matrix. We call P a fuzzy elementary matrix. Denote the relation that $W_2 = P\circ W_1$ between W_1 and W_2 by $W_1 \xrightarrow{P} W_2$. Suppose Δ_1, Δ_2 are fuzzy row-restricted matrices. We call $(\Delta_1\circ R,\ \Delta_2\circ L)$ a fuzzy connection network of $(R,\ L)$, also, the connection network of $(R,\ L)$ for simplicity.

Introduce the following notations for the given fuzzy pattern $(\mathbf{a},\mathbf{b})$: $\mathbf{a} = (a_1,...,a_n)$, $\mathbf{b} = (b_1,...,b_m)$, fuzzy matrices $R=(r_{ij})\in\mu_{n\times m}$, $L=(l_{ji})\in\mu_{m\times n}$ and $(k_1,\ k_2)$, $(i,\ j)\in N\times M$:

$$N^{k_1} = \mathrm{N}\setminus\{k_1\}, M^{k_2} = M\setminus\{k_2\},$$

$$E_{AB}(j) = \{i'\in \mathrm{N}|a_{i'} = b_j\},\ E_{BA}(i) = \{j'\in \mathrm{M}|b_{j'} = a_i\},$$

$$G_{AB}(j) = \{i'\in \mathrm{N}|a_{i'} > b_j\},\ G_{BA}(i) = \{j'\in \mathrm{M}|b_{j'} > a_i\},$$

$$I_{AB}(j) = \{i'\in \mathrm{N}|a_{i'} \ge b_j\},\ I_{BA}(i) = \{j'\in \mathrm{M}|b_{j'} \ge a_i\},$$

$$J_{RB}(j) = \{i'\in \mathrm{N}|r_{i'j} \ge b_j\},\ J_{LA}(i) = \{j'\in \mathrm{M}|l_{j'i} \ge a_i\},$$

$$G_{RB}(j) = \{i'\in \mathrm{N}|r_{i'j} > b_j\},\ G_{LA}(i) = \{j'\in \mathrm{M}|l_{j'i} > a_i\}.$$

If P is a subset of N or M, χ_P means the characteristic function of P, and Card(P) means the cardinal number of P, i.e. the total number of elements in P.

3.5.2 The connection relations of attractors

In the following, we study the relations between the attractors of $(R,\ L)$ and ones of the corresponding connection network. Considering Theorem 3.10 we generalize the definition of attractors of (3.58), i.e. the fuzzy pattern $(\mathbf{a},\mathbf{b})$ is called the attractor of the system (3.58) if $\mathbf{b} = \mathbf{a}\circ R,\ \ \mathbf{a} = \mathbf{b}\circ L$. Also $(\mathbf{a},\mathbf{b})$ is called the attractor of $(R,\ L)$.

Theorem 3.17 *Let the fuzzy pattern pair* $(\mathbf{a},\mathbf{b})$: $\mathbf{a} = (a_1,...,a_n)$, $\mathbf{b} = (b_1,...,b_m)$, *be the attractor of* $(R,\ L)$, $k_1\in\mathrm{N}$, $k_2\in\mathrm{M}$. *Then* $(\mathbf{a},\mathbf{b})$ *is a attractor of the connection network* $(\Delta_{k_1}(\alpha)\circ R,\ \Delta_{k_2}(\beta)\circ L)$ *for each* $\alpha,\ \beta\in$

[0, 1] *if and only if the following hold:*

$$\begin{cases} \forall j \in \mathrm{M},\ N^{k_1} \cap I_{AB}(j) \cap J_{RB}(j) \neq \emptyset, \\ \forall i \in \mathrm{N},\ M^{k_2} \cap I_{BA}(i) \cap J_{LA}(i) \neq \emptyset. \end{cases} \tag{3.60}$$

Proof. For each $\alpha,\ \beta \in [0, 1]$, at first we suppose $(\mathbf{a}, \mathbf{b})$ is the attractor of $\big(\Delta_{k_1}(\alpha) \circ R,\ \Delta_{k_2}(\beta) \circ L\big)$, but (3.60) is not true. Then it is no harm to assume there exists $j_0 \in \mathrm{M}$, so that $N^{k_1} \cap I_{AB}(j_0) \cap J_{RB}(j_0) = \emptyset$. Denote $y_{j_0} = \big(A \circ (\Delta_{k_1}(\alpha) \circ R)\big)_{j_0}$. Considering $\forall i \in \mathrm{N}^{k_1},\ i \notin I_{AB}(j_0) \cap J_{RB}(j_0)$, we get

$$i \in \mathrm{N}^{k_1} \Longrightarrow a_i < b_{j_0} \ \text{ or, } \ r_{ij_0} < b_{j_0} \Longrightarrow b_i \wedge w_{ij_0} < b_{j_0}.$$

Thus, $\bigvee_{i \in \mathrm{N}^{k_1}} \{a_i \wedge r_{ij_0}\} < b_{j_0}$. Therefore if let $\alpha < b_{j_0}$, and

$$y_{j_0} = \Big(\bigvee_{i \in \mathrm{N}^{k_1}} \{a_i \wedge r_{ij_0}\} \Big) \vee \big(a_{k_1} \wedge \alpha \wedge r_{k_1 j_0}\big) < b_{j_0}.$$

we can conclude that $\mathbf{b} \neq \mathbf{a} \circ (\Delta_{k_1}(\alpha) \circ R)$, that is, $(\mathbf{a}, \mathbf{b})$ is not the attractor of $\big(\Delta_{k_1}(\alpha) \circ R,\ \Delta_{k_2}(\beta) \circ L\big)$, which contradicts the assumptions of the theorem. Thus the condition (3.60) holds.

Conversely, if the condition (3.60) holds, then $\forall i \in \mathrm{N},\ j \in \mathrm{M}$, there exist $i_0 \in \mathrm{N}^{k_1} \cap I_{AB}(j) \cap J_{RB}(j),\ j_0 \in \mathrm{M}^{k_2} \cap I_{BA}(i) \cap J_{LA}(i)$. Consequently we have

$$i_0 \in \mathrm{N}^{k_1}, \text{ and } \ a_{i_0} \geq b_j,\ r_{i_0 j} \geq b_j;\ j_0 \in \mathrm{M}^{k_2}, \text{ and } \ b_{j_0} \geq a_i,\ l_{j_0 i} \geq a_i.$$

Therefore considering the assumptions and (iii) of Proposition 3.1, we obtain $\mathbf{b} = \mathbf{a} \circ R \supset \mathbf{a} \circ (\Delta_{k_1}(\alpha) \circ R),\ \mathbf{a} = \mathbf{b} \circ L \supset \mathbf{b} \circ (\Delta_{k_2}(\beta) \circ L)$. and

$$b_j \geq \big(\mathbf{a} \circ (\Delta_{k_1}(\alpha) \circ R)\big)_j = \Big(\bigvee_{i' \in \mathrm{N}^{k_1}} \{a_{i'} \wedge r_{i'j}\} \Big) \vee \big(\alpha \wedge r_{k_1 j} \wedge a_{k_1}\big) \geq a_{i_0} \wedge r_{i_0 j} \geq b_j,$$

$$a_i \geq \big(\mathbf{b} \circ (\Delta_{k_2}(\beta) \circ L)\big)_i = \Big(\bigvee_{j' \in \mathrm{M}^{k_2}} \{b_{j'} \wedge l_{j'i}\} \Big) \vee \big(\beta \wedge l_{k_2 i} \wedge b_{k_2}\big) \geq b_{j_0} \wedge r_{j_0 i} \geq a_i.$$

Hence, $\big(\mathbf{a} \circ (\Delta_{k_1}(\alpha) \circ R)\big)_j = b_j,\ \big(\mathbf{b} \circ (\Delta_{k_2}(\beta) \circ L)\big)_i = a_i$. Thus $(\mathbf{a}, \mathbf{b})$ is the attractor of $\big(\Delta_{k_1}(\alpha) \circ R,\ \Delta_{k_2}(\beta) \circ L\big)$. □

By Theorem 3.17, the following corollary is trivial.

Corollary 3.1 *Let the fuzzy pattern pair* $(\mathbf{a}, \mathbf{b})$ *be the attractor of* $(R,\ L)$. *And* $k_1 \in \mathrm{N},\ k_2 \in \mathrm{M}$. *Then the sufficient and necessary condition that* $(\mathbf{a}, \mathbf{b})$ *is not the attractor of* $\big(\Delta_{k_1}(\alpha) \circ R,\ \Delta_{k_2}(\beta) \circ L\big)$ *for each* $\alpha,\ \beta \in [0, 1]$, *is that one of the following conditions holds:*

(i) There is $j_0 \in \mathrm{M}^{k_2}$, *so that* $I_{AB}(j_0) \cap J_{RB}(j_0) = \{k_1\}$ *or* $\emptyset$;

(ii) There exists $i_0 \in \mathrm{N}^{k_1}$, *satisfying* $I_{BA}(i_0) \cap J_{LA}(i_0) = \{k_2\}$ *or* $\emptyset$;

For $p_1,\ q_1 \in \mathrm{N},\ p_2,\ q_2 \in \mathrm{M},\ (i,j) \in \mathrm{N} \times \mathrm{M}$, define $RS_j(p_1,q_1),\ RT_j(p_1,q_1)$, and $LS_i(p_2,q_2),\ LT_i(p_2,q_2)$ respectively as follows:

$$RS_j(p_1,q_1) = \bigvee_{i' \in \mathrm{N}^{p_1} \cap N^{q_1}} \{a_{i'} \wedge r_{i'j}\},\ RT_j(p_1,q_1) = \big(r_{p_1 j} \wedge a_{q_1}\big) \vee \big(r_{q_1 j} \wedge a_{p_1}\big),$$

$$LS_i(p_2,q_2) = \bigvee_{j' \in \mathrm{M}^{p_2} \cap M^{q_2}} \{b_{j'} \wedge l_{j'i}\},\ LT_i(p_2,q_2) = \big(l_{p_2 i} \wedge b_{q_2}\big) \vee \big(l_{q_2 i} \wedge b_{p_2}\big).$$

Let $\lambda_1,\ \lambda_2 \in [0,\ 1]$. we call $(\lambda_1,\ \lambda_2)$ the neighboring values in $\mathbf{x} \in [0,\ 1]^n$, if for each component x of $\mathbf{x}$, either $x \le \lambda_1 \wedge \lambda_2$ or, $x \ge \lambda_1 \vee \lambda_2$. Obviously, if $\lambda_1 = \lambda_2$, $(\lambda_1,\ \lambda_2)$ are neighboring values in each $\mathbf{x} \in [0,\ 1]^n$.

Theorem 3.18 *Suppose* $(\mathbf{a},\mathbf{b})$: $\mathbf{a} = (a_1,...,a_n)$, $\mathbf{b} = (b_1,...,b_m)$ *is the attractor of* $(R,\ L)$, *and* $p_1,\ q_1 \in \mathrm{N},\ p_2,\ q_2 \in \mathrm{M}$, *so that* $(a_{p_1},\ a_{q_1})$ *are neighboring values in* $\mathbf{b}$, $(b_{p_2},\ b_{q_2})$ *are the neighboring values in* $\mathbf{a}$. *And* $a_{p_1} > a_{q_1}$, $b_{p_2} > b_{q_2}$. *Then* $(\mathbf{a},\mathbf{b})$ *is also the attractor of* $\big(\Delta_{p_1 q_1} \circ R,\ \Delta_{p_2 q_2} \circ L\big)$ *if and only if one of the following conditions holds for any* $i \in \mathrm{N},\ j \in \mathrm{M}$:

$$\text{(i)} \begin{cases} RS_j(p_1,q_1) \ge b_j,\ RT_j(p_1,q_1) \le b_j, \\ LS_i(p_2,q_2) \ge a_i,\ LT_i(p_2,q_2) \le a_i; \end{cases}$$

$$\text{(ii)} \begin{cases} 1 - \chi_{E_{BA}(p_1)}(j) + r_{q_1 j} \ge a_{p_1},\ \chi_{E_{BA}(q_1)}(j) - 1 + r_{q_1 j} \le a_{q_1}, \\ 1 - \chi_{E_{AB}(p_2)}(i) + l_{q_2 i} \ge b_{p_2},\ \chi_{E_{AB}(q_2)}(i) - 1 + l_{q_2 i} \le b_{q_2}. \end{cases}$$

Proof. Sufficiency: Give arbitrarily $i \in \mathrm{N},\ j \in \mathrm{M}$. Let (i) hold, then considering the following fact:

$$b_j = \bigvee_{k \in \mathrm{N}} \{r_{kj} \wedge a_k\} = RS_j(p_1,q_1) \vee \big(r_{p_1 j} \vee a_{p_1}\big) \vee \big(r_{q_1 j} \vee a_{q_1}\big),$$

we get, $RS_j(p_1,q_1) = b_j$. Therefore $RS_j(p_1,q_1) \vee RT_j(p_1,q_1) = b_j$, and consequently $(\mathbf{a} \circ (\Delta_{p_1 q_1} \circ R))_j = b_j$. With the same reasons we may show, $\big(\mathbf{b} \circ (\Delta_{p_2 q_2} \circ L)\big)_i = a_i$.

Suppose (ii) holds. In order to prove $\big(\mathbf{a} \circ (\Delta_{p_1 q_1} \circ R)\big)_j = b_j$, it suffices by the assumptions to show the conclusion holds in the following four cases:

$$\text{(i}'\text{)}\ b_j > a_{p_1},\quad \text{(ii}'\text{)}\ b_j = a_{p_1},\quad \text{(iii}'\text{)}\ b_j = a_{q_1},\quad \text{(iv}'\text{)}\ b_j < a_{q_1}.$$

To case (i′), $b_j > a_{p_1} > a_{q_1}$, then $\big(a_{p_1} \wedge r_{p_1 j}\big) \vee \big(a_{q_1} \wedge r_{q_1 j}\big) < b_j$, and $RT_j(p_1,q_1) < b_j$. Since $\mathbf{b} = \mathbf{a} \circ R$, it follows that $RS_j(p_1,q_1) = b_j$, moreover

$$\big(\mathbf{a} \circ (\Delta_{p_1 q_1} \circ R)\big)_j = RS_j(p_1,q_1) \vee RT_j(p_1,q_1) = b_j. \tag{3.61}$$

To case (ii′), $j \in E_{BA}(p_1)$, by the condition (ii) we obtain

$$\chi_{E_{BA}(p_1)}(j) = 1,\ \ r_{q_1 j} \ge a_{p_1}. \tag{3.62}$$

Since $(\mathbf{a},\mathbf{b})$ is the attractor of $(R,\ L)$, the following fact holds:

$$b_j = \bigvee_{k\in \mathrm{N}} \{a_k \wedge r_{kj}\}. \tag{3.63}$$

So using (3.62) (3.63) and the fact $b_j = a_{p_1}$ we can conclude that

$$b_j \geq RS_j(p_1,q_1) \Longrightarrow RS_j(p_1,q_1) \vee \big(r_{p_1 j} \wedge a_{q_1}\big)\vee\big(r_{q_1 j} \wedge a_{p_1}\big)= b_j.$$

Therefore, $\big(\mathbf{a}\circ(\Delta_{p_1 q_1}\circ R)\big)_j = RS_j(p_1,q_1)\vee RT_j(p_1,q_1) = b_j$, and (3.61) holds.

To case (iii′), $j\in E_{BA}(q_1)$, so by the condition (ii), (3.61) and the inequality: $a_{p_1} > a_{q_1}$ it follows that $\chi_{E_{BA}(q_1)}(j) = 1,\ \ r_{q_1 j}\leq a_{q_1}$, and $r_{p_1 j}\leq b_j$. Thus

$$\begin{aligned} b_j &= RS_j(p_1,q_1) \vee \big(r_{p_1 j} \wedge a_{p_1}\big)\vee\big(r_{q_1 j} \wedge a_{q_1}\big) \\ &= RS_j(p_1,q_1) \vee \big(r_{p_1 j} \wedge a_{q_1}\big)\vee\big(r_{q_1 j} \wedge a_{p_1}\big). \end{aligned}$$

That is, $\big(\mathbf{b}\circ(\Delta_{pq}\circ W)\big)_j = b_j$.

To case (iv′), $b_j < a_{q_1}$. Then $r_{p_1 j}\wedge a_{p_1} < a_{q_1} < a_{p_1},\ r_{q_1 j}\wedge a_{q_1} < a_{q_1} < a_{p_1}$. Hence $\big(r_{p_1 j}\wedge a_{p_1}\big)\vee\big(r_{q_1 j}\wedge a_{q_1}\big) = r_{p_1 j}\vee r_{q_1 j} = RT_j(p_1,q_1)$. Consequently

$$\begin{aligned} b_j &= RS_j(p_1,q_1) \vee \Big(\bigvee_{k\in\{p_1,q_1\}} \{b_k\wedge r_{kj}\}\Big) \\ &= RS_j(p_1,q_1)\vee RT_j(p_1,q_1) = \big(\mathbf{b}\circ(\Delta_{pq}\circ R)\big)_j. \end{aligned}$$

So $b_j = \big(\mathbf{b}\circ(\Delta_{pq}\circ R)\big)_j$. With the same reason, $\big(\mathbf{b}\circ(\Delta_{p_2 q_2}\circ L)\big)_i = a_i$. Hence $\forall i\in \mathrm{N},\ j\in \mathrm{M}$, If (i) or, (ii) holds, we have, $\mathbf{b} = \mathbf{a}\circ\big(\Delta_{p_1 q_1}\circ R\big),\ \mathbf{a} = \mathbf{b}\circ\big(\Delta_{p_2 q_2}\circ L\big)$, i.e. $(\mathbf{a},\mathbf{b})$ is the attractor of $\big(\Delta_{p_1 q_1}\circ R,\ \Delta_{p_2 q_2}\circ L\big)$.

Necessity: Suppose $(\mathbf{a},\mathbf{b})$ is the attractor of $\big(\Delta_{p_1 q_1}\circ R,\ \Delta_{p_2 q_2}\circ L\big)$. Then

$$\begin{aligned} b_j &= RS_j(p_1,q_1) \vee \big(r_{p_1 j} \wedge a_{p_1}\big)\vee\big(r_{q_1 j} \wedge a_{q_1}\big) \\ &= RS_j(p_1,q_1) \vee \big(r_{p_1 j} \wedge a_{q_1}\big)\vee\big(r_{q_1 j} \wedge a_{p_1}\big). \end{aligned} \tag{3.64}$$

For $j\in \mathrm{M}$, if the condition (i) is false, then $RS_j(p_1,q_1) < b_j$, or $RT_j(p_1,q_1) > b_j$. Since $b_j = RS_j(p_1,q_1)\vee RT_j(p_1,q_1)$, we have, $RS_j(p_1,q_1) < b_j$. Let us next show (ii) in the preceding four cases: (i′) (ii′) (iii′) and (iv′), respectively.

If $b_j = a_{p_1}$, then considering $RS_j(p_1,q_1) < b_j$ and $b_j = a_{p_1} > a_{q_1}$, and (3.64) we can show, $r_{q_1 j}\geq a_{p_1}$. So $1-\chi_{E_{BA}(p_1)}(j) + r_{q_1 j} = r_{q_1 j}\geq a_{p_1}$. If $b_j = a_{q_1}$, then (3.62) implies, $r_{q_1 j}\leq a_{q_1}$. Thus, $\chi_{E_{BA}(q_1)}(j) - 1 + r_{q_1 j} = r_{q_1 j}\leq a_{q_1}$. To the other cases, obviously we have

$$1-\chi_{E_{BA}(p_1)}(j) + r_{q_1 j}\geq a_{p_1},\ \chi_{E_{BA}(q_1)}(j) - 1 + r_{q_1 j}\leq a_{q_1}.$$

Similarly we can prove, if for $i\in \mathrm{M}$, the condition (i) is false, then

$$1-\chi_{E_{AB}(p_2)}(i) + l_{q_2 i}\geq b_{p_2},\ \chi_{E_{AB}(q_2)}(i) - 1 + l_{q_2 i}\leq b_{q_2}.$$

Hence $\forall i \in$ N, $j \in$ M, the condition (i) or (ii) is true. □

3.5.3 The elementary memory of $(R,\ L)$

It is very important for FBAM's to construct some dynamical networks with better fault-tolerance. By the method in the subsection, we may establish some novel ways to do that by the elementary memory of the FBAM (3.58) if $(R,\ L)$ can be trained so that the given fuzzy pattern pairs are the elementary memories of the network.

Definition 3.9 Suppose $(\mathbf{a},\mathbf{b})$ is a fuzzy pattern pair, we call $(\mathbf{a},\mathbf{b})$ an elementary memory of $(R,\ L)$ if the following conditions hold:

(i) For any $k_1 \in$ N, $k_2 \in$ M and $\alpha,\ \beta \in [0,\ 1]$, $(\mathbf{a},\mathbf{b})$ is the attractor of $\big(\Delta_{k_1}(\alpha)\circ R,\ \Delta_{k_2}(\beta)\circ L\big)$;

(ii) $\forall p_1,\ q_1 \in$ N, and $\forall p_2,\ q_2 \in$ M, $(\mathbf{a},\mathbf{b})$ is the attractor of the connection network $\big(\Delta_{p_1,q_1}\circ R,\ \Delta_{p_2 q_2}\circ L\big)$.

Obviously if $(\mathbf{a},\mathbf{b})$ is an elementary memory of $(R,\ L)$, $(\mathbf{a},\mathbf{b})$ is also the attractor of $(R,\ L)$. Next let us present some sufficient and necessary conditions that a attractor of $(R,\ L)$ is an elementary memory of $(R,\ L)$.

Theorem 3.19 *Assume that the fuzzy pattern pair* $(\mathbf{a},\mathbf{b})$ *is an attractor of* $(R,\ L)$. *Then the sufficient and necessary condition that* $(\mathbf{a},\mathbf{b})$ *is the attractor of* $\big(\Delta_{k_1}(\alpha)\circ R,\ \Delta_{k_2}(\beta)\circ L\big)$ *for each* $k_1 \in$ N, $k_2 \in$ M, $\alpha,\ \beta \in [0,\ 1]$, *is that the following conditions hold:*

$$\begin{cases} \forall j \in \mathrm{M},\ \mathrm{Card}\big(I_{AB}(j)\cap J_{RB}(j)\big)\geq 2; \\ \forall i \in \mathrm{N},\ \mathrm{Card}\big(I_{BA}(i)\cap J_{LA}(i)\big)\geq 2. \end{cases} \tag{3.65}$$

Proof. Sufficiency: For each $j \in$ M, if (3.65) is true, we suppose $i_1,\ i_2 \in I_{AB}(j)\cap J_{RB}(j):\ i_1 \neq i_2$. Then

$$r_{i_k j} \geq b_j,\ a_{i_k} \geq b_j\ (k=1,2). \tag{3.66}$$

$\forall k_1 \in$ N, $\alpha \in [0,\ 1]$, By (3.66), the assumptions and the condition (iii) of Proposition 3.1 we can show

$$k_1 \notin \{i_1,\ i_2\}, \Longrightarrow b_j \geq \big(\mathbf{a}\circ(\Delta_{k_1}(\alpha)\circ R)\big)_j \geq (a_{i_1}\wedge r_{i_1 j})\vee(a_{i_2}\wedge r_{i_2 j}) \geq b_j. \tag{3.67}$$

if $k_1 \in \{i_1,\ i_2\}$, it is no harm to assume $k_1 = i_1$. Similarly with (3.67) it follows that

$$b_j \geq \big(\mathbf{a}\circ(\Delta_{k_1}(\alpha)\circ R)\big)_j \geq a_{i_2}\wedge r_{i_2 j} \geq b_j.$$

In summary, $\big(\mathbf{a}\circ(\Delta_{k_1}(\alpha)\circ R)\big)_j = b_j\ (j \in \mathrm{M})$.

With the same reason, $\forall i \in$ N, $\big(\mathbf{b}\circ(\Delta_{k_2}(\beta)\circ L)\big)_i = a_i\ (k_2 \in \mathrm{N},\ \beta \in [0,\ 1])$. Therefore, for any $k_1 \in$ N, $k_2 \in$ M, and $\alpha,\ \beta \in [0,\ 1]$, $(\mathbf{a},\mathbf{b})$ is the attractor of $\big(\Delta_{k_1}(\alpha)\circ R,\ \Delta_{k_2}(\beta)\circ L\big)$.

Necessity: At first we can show the following facts for any $i \in \mathrm{N},\ j \in \mathrm{M}$:

$$\begin{cases} i' \in I_{AB}(j) \cap J_{RB}(j) \Longleftrightarrow r_{i'j} \wedge a_{i'} \geq b_j, \\ j' \in I_{BA}(i) \cap J_{LA}(i) \Longleftrightarrow l_{j'i} \wedge b_{j'} \geq a_i. \end{cases} \tag{3.68}$$

If (3.65) is false, it is no harm to assume $j_0 \in \mathrm{N}$, satisfying the condition: $\mathrm{Card}\big(I_{AB}(j_0) \cap J_{RB}(j_0)\big) \leq 3$. Then (3.68) implies, either there is $i_0 \in \mathrm{N}$, so that $r_{i_0 j_0} \wedge a_{i_0} \geq b_{j_0}$, and $\forall i \in \mathrm{N}^{i_0}$, $r_{ij_0} \wedge a_i < b_{j_0}$; or, $\forall i \in \mathrm{N}$, $r_{ij_0} \wedge a_i < b_{j_0}$. Choose $k_1 = i_0$, $\alpha < b_{j_0}$, we get

$$\big(\mathbf{a} \circ (\Delta_{k_1}(\alpha) \circ R)\big)_{j_0} = \bigvee_{i \in \mathrm{N}^{i_0}} \{r_{ij_0} \wedge a_i\} \vee \big(\alpha \wedge r_{i_0 j_0} \wedge a_{i_0}\big) < b_{j_0}.$$

Thus, $\mathbf{b} \neq \mathbf{a} \circ (\Delta_{k_1}(\alpha) \circ R)$, which is a contradiction. So (3.65) holds. □

By the induction method and the proof of Theorem 3.19, we may easily show the following conclusions.

Corollary 3.2 *Let* $(\mathbf{a}, \mathbf{b})$ *be the attractor of* $(R,\ L)$ *and* $p \in \mathrm{N},\ q \in \mathrm{M},\ p \leq n-1,\ q \leq m-3$. *Then the sufficient and necessary conditions that* $(\mathbf{a}, \mathbf{b})$ *is the attractor of* $\big(\Delta_{k_1}(\alpha_1) \circ \cdots \circ \Delta_{k_p}(\alpha_p) \circ R,\ \Delta_{l_1}(\beta_1) \circ \cdots \circ \Delta_{l_q}(\beta_q) \circ L\big)$ *for arbitrary* $k_1, ..., k_p \in \mathrm{N},\ l_1, ..., l_q \in \mathrm{M}$ *and* $\alpha_1, ..., \alpha_p,\ \beta_1, ..., \beta_q \in [0,\ 1]$ *are as follows:*

$$\begin{cases} \forall j \in \mathrm{M},\ \mathrm{Card}\big(I_{AB}(j) \cap J_{RB}(j)\big) \geq p+1, \\ \forall i \in \mathrm{N},\ \mathrm{Card}\big(I_{BA}(i) \cap J_{LA}(i)\big) \geq q+1. \end{cases}$$

By Corollary 3.2, if the conditions related hold, a fuzzy pattern pair obtained based on $(\mathbf{a}, \mathbf{b})$ by letting p components of $\mathbf{a}$ and q components of $\mathbf{b}$ be changed in $[0,\ 1]$ and leaving other components unchanged, can converge to the attractor $(\mathbf{a}, \mathbf{b})$ with one iteration. The fact shows good fault-tolerance of the corresponding FBAM's.

Theorem 3.20 *Suppose the fuzzy pattern pair* $(\mathbf{a}, \mathbf{b})$ *is the attractor of* $(R,\ L)$, *and for* $\forall i \in \mathrm{N},\ j \in \mathrm{M}$, *the following conditions hold:*

(i) $G_{RB}(j) \times G_{AB}(j) = \emptyset,\ G_{LA}(i) \times G_{BA}(i) = \emptyset$;

(ii) $\mathrm{Card}\big((J_{RB}(j) \cap I_{AB}(j))\big) \geq 2,\ \mathrm{Card}\big((J_{LA}(i) \cap I_{BA}(i))\big) \geq 2$.

Then $(\mathbf{a}, \mathbf{b})$ *is an elementary memory of* $(R,\ L)$.

Conversely, if $(\mathbf{a}, \mathbf{b})$ *is the elementary memory of* $(R,\ L)$, *then for each* $i \in \mathrm{N},\ j \in \mathrm{M}$, *the above condition (i) and the following condition (iii) hold:*

(iii) $\mathrm{Card}\big(J_{RB}(j) \cap I_{AB}(j)\big) \geq 1$ *or,* $\mathrm{Card}\big(J_{LA}(i) \cap I_{BA}(i)\big) \geq 3$.

Proof. For each $i \in \mathrm{N},\ j \in \mathrm{M}$, we suppose (i) (ii) hold. By Theorem 3.19 it suffices to prove that $(\mathbf{a}, \mathbf{b})$ is the attractor of $\big(\Delta_{p_1 q_1} \circ R,\ \Delta_{p_2, q_2} \circ L\big)$ for any $p_1,\ q_1 \in N,\ p_2,\ q_2 \in \mathrm{M}$. By condition (i) we may easily show

$$\forall i_1,\ i_2 \in \mathrm{N},\ a_{i_1} \wedge r_{i_2 j} \leq b_j. \tag{3.69}$$

Also by (3.69), $\forall i_1,\ i_2 \in J_{RB}(j) \cap I_{AB}(j)$, it follows that

$$a_{i_1} \wedge r_{i_2 j} = b_j. \tag{3.70}$$

Using the condition (ii), we suppose $i',\ i'' \in \mathrm{N}:\ i' \neq i''$, so that $i',\ i'' \in J_{RB}(j) \cap I_{AB}(j)$. If $\{i',\ i''\} \cap \{p_1,\ q_1\} = \emptyset$, by (3.69) (3.70) we imply

$$RS_j(p_1, q_1) = \bigvee_{k \in \mathrm{N}^{p_1} \cap N^{q_1}} \{a_k \wedge r_{kj}\} = b_j,$$

$$RT_j(p_1, q_1) = \big(a_{p_1} \wedge r_{q_1 j}\big) \vee \big(a_{q_1} \wedge r_{p_1 j}\big) \leq b_j.$$

Therefore we can conclude that

$$\big(\mathbf{a} \circ (\Delta_{p_1 q_1} \circ R)\big)_j = RS_j(p_1, q_1) \vee RT_j(p_1, q_1) = b_j. \tag{3.71}$$

If $\{i',\ i''\} \cap \{p_1,\ q_1\} \neq \emptyset$, we may assume $i' = p_1,\ i'' = q_3$. By (3.70) it follows that

$$RT_j(p_1, q_1) = \big(a_{p_1} \wedge r_{q_1 j}\big) \vee \big(a_{q_1} \wedge r_{p_1 j}\big) = b_j.$$

And (3.69) implies, $RS_j(p_1, q_1) \leq b_j$. Thus (3.71) is also true. If $i' = p_1,\ q_1 \notin \{i',\ i''\}$, by (3.69) (3.70) we get

$$RT_j(p_1, q_1) = \big(a_{i'} \wedge r_{q_1 j}\big) \vee \big(a_{q_1} \wedge r_{i' j}\big) \leq b_j,$$

$$RS_j(p_1, q_1) = \bigvee_{k \in \mathrm{N}^{p_1} \cap N^{q_1}} \{a_k \wedge r_{kj}\} \geq a_{i''} \wedge r_{i'' j} = b_j.$$

On the other hand, by (3.69), $RS_j(p_1, q_1) \leq b_j$. Therefore, $(\mathbf{a} \circ (\Delta_{p_1 q_1}) \circ R)_j = RS_j(p_1, q_1) \vee RT_j(p_1, q_1) = b_j$, i.e. (3.71) is true.

In summary, $\big(\mathbf{a} \circ (\Delta_{p_1 q_1} \circ R)\big)_j = b_j\ (j \in \mathrm{M})$. Similarly we can show that, $\big(\mathbf{b} \circ (\Delta_{p_2 q_2} \circ L)\big)_i = a_i\ (i \in \mathrm{N})$, that is, $\forall p_1,\ q_1 \in \mathrm{N},\ p_2,\ q_2 \in \mathrm{M},\ (\mathbf{a}, \mathbf{b})$ is the attractor of $\big(\Delta_{p_1 q_1} \circ R,\ \Delta_{p_2 q_2} \circ L\big)$.

Conversely, let $(\mathbf{a}, \mathbf{b})$ be an elementary memory of $(R,\ L)$. Then $\forall p_1,\ q_1 \in \mathrm{N},\ p_2,\ q_2 \in \mathrm{M},\ (\mathbf{a}, \mathbf{b})$ is the attractive of $\big(\Delta_{p_1 q_1} \circ R,\ \Delta_{p_2 q_2} \circ L\big)$. Obviously, if let $p_1 = q_1,\ p_2 = q_2$, it follows that $(\mathbf{a}, \mathbf{b})$ is the attractor of $(R,\ L)$. If the condition (i) is false, we may assume $j_0 \in \mathrm{M}$, and let $G_{RB}(j_0) \times G_{AB}(j_0) \neq \emptyset$. For $(i_1,\ i_2) \in G_{RB}(j_0) \times G_{AB}(j_0)$, we have, $r_{i_1 j_0} > b_{j_0}, a_{i_2} > b_{j_0}$. Let $p_1 = i_1,\ q_1 = i_2$. Then

$$RT_{j_0}(p_1, q_1) = \big(r_{i_1 j_0} \wedge a_{i_2}\big) \vee \big(r_{i_2 j_0} \wedge a_{i_1}\big) > b_{j_0}.$$

Hence $\big(\mathbf{a} \circ (\Delta_{p_1 q_1} \circ R)\big)_{j_0} \geq RT_{j_0}(p_1, q_1) > b_{j_0}$. Thus, the fuzzy pattern pair $(\mathbf{a}, \mathbf{b})$ is not the attractor of $\big(\Delta_{p_1 q_1} \circ R,\ \Delta_{p_2 q_2} \circ L\big)$, which is a contradiction. That is, $\forall i \in \mathrm{N},\ j \in \mathrm{M}$, the condition (i) is true.

If we assume that the condition (iii) is false, choose $j_0 \in \mathrm{M}$, satisfying $\mathrm{Card}\big((J_{RB}(j_0) \cap I_{AB}(j_0))\big) = 0$. It is easy to show, $\forall k \in \mathrm{N},\ r_{k j_0} < b_{j_0}$, or

$a_k < b_{j_0}$. So $\bigvee_{k\in\mathrm{N}} \{a_k \wedge r_{kj_0}\} < b_{j_0}$, which is also a contradiction, since $(\mathbf{a}, \mathbf{b})$ is an attractor of $(R,\ L)$. The condition (iii) holds. □

3.5.4 The transition laws of states

For given fuzzy matrices $R = (r_{ij}) \in \mu_{n\times m}$, $L = (l_{ji}) \in \mu_{m\times n}$, define a transformation $T_{(R,L)}:\ [0,1]^n \times [0,1]^m \longrightarrow [0,1]^m \times [0,1]^n$, so that

$$\forall \mathbf{x} \in [0,1]^n,\ \mathbf{y} \in [0,1]^m,\ T_{(R,L)}(\mathbf{x},\mathbf{y}) \triangleq (\mathbf{x},\mathbf{y})\circ(R,\ L) \triangleq (\mathbf{x}\circ R,\ \mathbf{y}\circ L). \quad (3.72)$$

Assume $R \in \mu_{n\times m}$, $P \in \mu_{n\times n}$, $L \in \mu_{m\times n}$, $Q \in \mu_{m\times m}$, also define a transformation $T_{(R,L)} \circ T_{(P,Q)}:\ [0,1]^n \times [0,1]^m \longrightarrow [0,1]^m \times [0,1]^n$ as follows:

$$\forall \mathbf{x} \in [0,1]^n,\ \mathbf{y} \in [0,1]^m,\ \big(T_{(R,L)}\circ T_{(P,Q)}\big)(\mathbf{x},\mathbf{y}) = T_{(R,L)}(T_{(P,Q)}(\mathbf{x},\mathbf{y})). \quad (3.73)$$

Lemma 3.4 *Let $R \in \mu_{n\times m}$, $P \in \mu_{n\times n}$, $L \in \mu_{m\times n}$, $Q \in \mu_{m\times m}$. Then $T_{(P\circ R,\ Q\circ L)} = T_{(R,L)} \circ T_{(P,Q)}$.*

Proof. For any $(\mathbf{x},\mathbf{y}) \in [0,1]^n \times [0,1]^m$, it is easy to show

$$\begin{aligned} T_{(P\circ R,\ Q\circ L)}(\mathbf{x},\mathbf{y}) &= (\mathbf{x},\mathbf{y})\circ(P\circ R,\ Q\circ L) = (\mathbf{x}\circ P\circ R,\ \mathbf{y}\circ Q\circ L) \\ &= (\mathbf{x}\circ P,\ \mathbf{y}\circ Q)\circ(R,L) = (\mathbf{x},\mathbf{y})\circ(P,Q)\circ(R,L) \\ &= T_{(R,L)}((\mathbf{x},\mathbf{y})\circ(P,Q)) = T_{(R,L)}(T_{(P,Q)}(\mathbf{x},\mathbf{y})) \\ &= (T_{(R,L)}\circ T_{(P,Q)})(\mathbf{x},\mathbf{y}). \end{aligned}$$

Therefore, $T_{(P\circ R, Q\circ L)} = T_{(R,L)} \circ T_{(P,Q)}$. □

Theorem 3.21 *Let the FBAM's $(R_1,\ L_1)$, $(R_2,\ L_2)$ satisfy the condition: $R_1 \xrightarrow{P} R_2$, $L_1 \xrightarrow{Q} L_2$. Suppose $P \in \mu_{n\times n}$, $Q \in \mu_{m\times m}$ are fuzzy fuzzy elementary matrices. There are row-restricted matrices $P_1,...,P_k$, $Q_1,...,Q_k$ satisfying, $P = P_k \circ P_{k-1} \circ\cdots\circ P_2 \circ P_1$, $Q = Q_k \circ Q_{k-1} \circ\cdots\circ Q_2 \circ Q_3$. Then*

$$T_{(R_2,L_2)} = T_{(R_1,L_1)} \circ T_{(P,Q)} = T_{(R_1,L_1)} \circ T_{(P_1,Q_1)} \circ\cdots\circ T_{(P_k,Q_k)}. \quad (3.74)$$

Proof. Lemma 3.4 implies that

$$\begin{aligned} T_{(R_2,L_2)} &= T_{(P\circ R_1, Q\circ L_1)} = T_{(P_k\circ P_{k-1}\circ\cdots\circ P_2\circ P_1\circ R_1,\ Q_k\circ Q_{k-1}\circ\cdots\circ Q_2\circ Q_1\circ L_1)} \\ &= T_{(R_1,L_1)} \circ T_{(P_k\circ P_{k-1}\circ\cdots\circ P_2\circ P_1,\ Q_k\circ Q_{k-1}\circ\cdots\circ Q_2\circ Q_1)} \\ &= T_{(R_1,L_1)} \circ T_{(P_1,Q_1)} \circ T_{(P_k\circ P_{k-1}\circ\cdots\circ P_2,\ Q_k\circ Q_{k-1}\circ\cdots\circ Q_2)} \\ &= \cdots\cdots\cdots \\ &= T_{(R_1,L_1)} \circ T_{(P_1,Q_1)} \circ\cdots\circ T_{(P_{k-1},Q_{k-1})} \circ T_{(P_k,Q_k)}. \end{aligned}$$

So ti follows that (3.74) is true. □

Definition 3.10 Suppose the fuzzy pattern pair $(\mathbf{a},\mathbf{b})$ is an attractor of $(R,\ L)$. And

$$\mathcal{N}(R,\ L;\ (\mathbf{a},\mathbf{b}))=\big\{(\mathbf{x},\mathbf{y})\in[0,1]^n\times[0,1]^m\big|\mathbf{x}\circ R=\mathbf{b},\ \mathbf{y}\circ L=\mathbf{a}\big\}.$$

That is, for any $(\mathbf{x},\mathbf{y})\in\mathcal{N}\big(R,L;(\mathbf{a},\mathbf{b})\big)$, it converges with one iteration to the attractor $(\mathbf{a},\mathbf{b})$ of $(R,\ L)$. Thus, the fuzzy pattern pair family $\mathcal{N}\big(R,L;(\mathbf{a},\mathbf{b})\big)$ is the attractive basin of $(\mathbf{a},\mathbf{b})$. Obviously

$$(\mathbf{x},\mathbf{y})\in\mathcal{N}\big(R,L;(\mathbf{a},\mathbf{b})\big)\Longleftrightarrow T_{(R,L)}(\mathbf{x},\mathbf{y})=(\mathbf{a},\mathbf{b}).$$

If the sequence of fuzzy pattern pairs $(\mathbf{a}_1,\ \mathbf{b}_1),...,(\mathbf{a}_l,\ \mathbf{b}_l)$ is the limit cycle of $(R,\ L)$, that is

$$\begin{cases}\mathbf{a}^2=\mathbf{b}^1\circ L,\\ \mathbf{b}^2=\mathbf{a}^1\circ R;\end{cases}\quad\begin{cases}\mathbf{a}^3=\mathbf{b}^2\circ L,\\ \mathbf{b}^3=\mathbf{a}^2\circ R;\end{cases}\quad\begin{cases}\mathbf{a}^l=\mathbf{b}^{l-1}\circ L,\\ \mathbf{b}^l=\mathbf{a}^{l-1}\circ R;\end{cases}\quad\begin{cases}\mathbf{a}^1=\mathbf{b}^l\circ L,\\ \mathbf{b}^1=\mathbf{a}^l\circ R.\end{cases}\tag{3.75}$$

Moreover

$$\mathcal{N}\big(R,L;(\mathbf{a}_1,\mathbf{b}_1),...,(\mathbf{a}_l,\mathbf{b}_l)\big)=\big\{(\mathbf{x},\mathbf{y})\in[0,1]^n\times[0,1]^m\big|\exists k\in\{1,...,l\}:\ \mathbf{x}\circ R=\mathbf{b}_k,\mathbf{y}\circ L=\mathbf{a}_k\big\}.$$

We call $\mathcal{N}\big(R,L;(\mathbf{a}_1,\mathbf{b}_1),...,(\mathbf{a}_l,\mathbf{b}_l)\big)$ an attractive basin of limit cycle $(\mathbf{a}_1,\ \mathbf{b}_1)$, ..., $(\mathbf{a}_l,\ \mathbf{b}_l)$.

Theorem 3.22 *Let $(\mathbf{a},\mathbf{b})$ be the attractor of $(R_1,\ L_1)$. $P\in\mu_{n\times n}$, $Q\in\mu_{m\times m}$, moreover, $R_1\xrightarrow{P}R_2$, $L_1\xrightarrow{Q}L_2$. Then $(\mathbf{a},\mathbf{b})$ is the attractor of $(R_2,\ L_2)$ if and only if $T_{(P,Q)}(\mathbf{a},\mathbf{b})\in\mathcal{N}\big(R_1,L_1;(\mathbf{a},\mathbf{b})\big)$.*

Proof. Suppose $(\mathbf{a},\mathbf{b})$ is the attractor of $(R_2,\ L_2)$. Then $\mathbf{b}\circ L_2=\mathbf{a},\ \mathbf{a}\circ R_2=\mathbf{b}$, i.e. $T_{(R_2,L_2)}(\mathbf{a},\mathbf{b})=(\mathbf{a},\mathbf{b})$. By Lemma 3.4 it follows that

$$\begin{aligned}(\mathbf{a},\mathbf{b})&=T_{(R_2,L_2)}(\mathbf{a},\mathbf{b})=(T_{(P\circ R_1,\ Q\circ L_1)}(\mathbf{a},\mathbf{b})\\&=(T_{(R_1,L_1)}\circ T_{(P,Q)})(\mathbf{a},\mathbf{b})=(T_{(P,Q)}(\mathbf{a},\mathbf{b}))\circ(R_1,L_1)\\&=(\mathbf{a}\circ P,\ \mathbf{b}\circ Q)\circ(R_1,L_1),\end{aligned}$$

That is, $\mathbf{a}\circ P\circ R_1=\mathbf{a},\ \mathbf{b}\circ Q\circ L_1=\mathbf{b}$. So $T_{(P,Q)}(\mathbf{a},\mathbf{b})\in\mathcal{N}\big(R_1,L_1;(\mathbf{a},\mathbf{b})\big)$. Conversely, if $T_{(P,Q)}(\mathbf{a},\mathbf{b})\in\mathcal{N}(R_1,L_1;(\mathbf{a},\mathbf{b}))$, also using Lemma 3.4 we get

$$\begin{aligned}(\mathbf{a},\mathbf{b})&=T_{(P,Q)}(\mathbf{a},\mathbf{b})\circ(R_1,L_1)=T_{(R_1,L_1)}(T_{(P,Q)}(\mathbf{a},\mathbf{b}))\\&=(T_{(R_1,L_1)}\circ T_{(P,Q)})(\mathbf{a},\mathbf{b})=T_{(P\circ R_1,Q\circ L_1)}(\mathbf{a},\mathbf{b})\\&=T_{(R_2,L_2)}(\mathbf{a},\mathbf{b}).\end{aligned}$$

Thus, $(\mathbf{a},\mathbf{b})$ is an attractor of $(R_2,\ L_2)$. □

Theorem 3.23 *Suppose the sequence of fuzzy pattern pairs as* $(\mathbf{a}_1,\ \mathbf{b}_1),\dots,(\mathbf{a}_l,\ \mathbf{b}_l)$ *is a limit cycle of* $(R_1,\ L_1)$*, and* $P\in\mu_{n\times n}$, $Q\in\mu_{m\times m}$*, moreover,* $R_1\xrightarrow{P}R_2,\ L_1\xrightarrow{Q}L_2$*. If the sequence of fuzzy pattern pairs is also a limit cycle of* $(R_2,\ L_2)$*, then*

$$\forall k\in\{1,\dots,l\},\ T_{(P,Q)}(\mathbf{a}_k,\mathbf{b}_k)\in\mathcal{N}\big(R_1,L_1;(\mathbf{a}_1,\mathbf{b}_1),\dots,(\mathbf{a}_l,\mathbf{b}_l)\big).$$

Conversely if let

$$T_{(P,Q)}(\mathbf{a}_k,\mathbf{b}_k)\triangleq\begin{cases}(\mathbf{a}'_k,\mathbf{b}'_k)\in\mathcal{N}\big(R_1,L_1;(\mathbf{a}_{k+1},\mathbf{b}_{k+1})\big), & 1\le k<l;\\(\mathbf{a}'_k,\mathbf{b}'_k)\in\mathcal{N}\big(R_1,L_1;(\mathbf{a}_1,\mathbf{b}_1)\big), & k=l.\end{cases}$$

Then $(\mathbf{a}_1,\ \mathbf{b}_1),\dots,(\mathbf{a}_l,\ \mathbf{b}_l)$ *is a limit cycle of* $(R_2,\ L_2)$.

Proof. Let the fuzzy pattern pairs $(\mathbf{a}_1,\ \mathbf{b}_1),\dots,(\mathbf{a}_l,\ \mathbf{b}_l)$ constitute a limit cycle of $(R_2,\ L_2)$. Then $\forall k\in\{1,\dots,l\}$, it follows that

$$\begin{aligned}T_{(P,Q)}\ (\mathbf{a}_k,\mathbf{b}_k)\circ(R_1,L_1)&=T_{(R_1,L_1)}(T_{(P,Q)}(\mathbf{a}_k,\mathbf{b}_k))\\&=(T_{(R_1,L_1)}\circ T_{(P,Q)})(\mathbf{a}_k,\mathbf{b}_k)=T_{(P\circ R_1,\ Q\circ L_1)}(\mathbf{a}_k,\mathbf{b}_k)\\&=T_{(R_2,L_2)}(\mathbf{a}_k,\mathbf{b}_k)=(\mathbf{a}_k,\mathbf{b}_k)\circ(R_2,L_2)\\&=\begin{cases}(\mathbf{a}_{k+1},\mathbf{b}_{k+1}), & 1\le k<l\\(\mathbf{a}_1,\mathbf{b}_1), & k=l.\end{cases}\end{aligned}$$

Therefore, $T_{(P,Q)}(\mathbf{a}_k,\ \mathbf{b}_k)\in\mathcal{N}\big(R_1,L_1;(\mathbf{a}_1,\ \mathbf{b}_1),\dots,(\mathbf{a}_l,\ \mathbf{b}_l)\big)$. Conversely, easily we have

$$\begin{aligned}(\mathbf{a}_1,\mathbf{b}_1)\circ(R_2,L_2)&=(\mathbf{a}_1,\mathbf{b}_1)\circ(P\circ R_1,Q\circ L_1)\\&=T_{(P\circ R_1,\ Q\circ L_1)}(\mathbf{a}_1,\mathbf{b}_1)=T_{(R_1,L_1)}\circ(T_{(P,Q)}(\mathbf{a}_1,\mathbf{b}_1))\\&=(\mathbf{a}'_1\mathbf{b}'_1)\circ(R_1,L_1)=(\mathbf{a}_2,\mathbf{b}_2).\end{aligned}$$

With the same reason, we get

$$\begin{aligned}&(\mathbf{a}_2,\mathbf{b}_2)\circ(R_2,L_2)=(\mathbf{a}_3,\mathbf{b}_3),\dots,(\mathbf{a}_{l-1},\mathbf{b}_{l-1})\circ(R_2,L_2)=(\mathbf{a}_l,\mathbf{b}_l),\\&(\mathbf{a}_l,\mathbf{b}_l)\circ(R_2,L_2)=(\mathbf{a}_1,\mathbf{b}_1).\end{aligned}$$

That is, $(\mathbf{a}_1,\ \mathbf{b}_1),\dots,(\mathbf{a}_l,\ \mathbf{b}_l)$ is a limit cycle of $(R_2,\ L_2)$. □

Next let us take an example to demonstrate our main conclusions. Let N = {1, 2, 3}, M = {1, 2, 3, 4}, and

$$R_1=\begin{pmatrix}0.8&0.4&0.5&0.6\\0.5&0.5&0.4&0.4\\0.6&0.5&0.5&0.6\end{pmatrix}\qquad L_1=\begin{pmatrix}0.7&0.4&0.6\\0.7&0.3&0.2\\0.4&0.5&0.4\\0.6&0.5&0.7\end{pmatrix}$$

We assume

$$\mathbf{a}_0 = (0.6, 0.5, 0.6), \ \ \mathbf{b}_0 = (0.6, 0.5, 0.5, 0.6);$$

$$\mathbf{a}_1 = (0.7, 0.5, 0.6), \ \ \mathbf{b}_1 = (0.7, 0.5, 0.5, 0.6).$$

Table 3.15 Attractors of FBAM

R_i	L_i	$P_i : R_i = P_i \circ R_1$	$L_i : L_i = Q_i \circ L_1$	attractors
R_1	L_1	$P_1 = \Delta$	$Q_1 = \Delta$	$(\mathbf{a}_0, \mathbf{b}_0); (\mathbf{a}_1, \mathbf{b}_1)$
R_2	L_2	$P_2 = \Delta_1(0.4)$	$Q_2 = \Delta_3(0.3)$	$(\mathbf{a}_0, \mathbf{b}_0)$
R_3	L_3	$P_3 = \Delta_{23}$	$Q_3 = \Delta_{14}$	$(\mathbf{a}_0, \mathbf{b}_0)$
R_4	L_4	$P_4 = \Delta_{12}$	$Q_4 = \Delta_{13}$	$(\mathbf{a}_0, \mathbf{b}_0)$

It is easy to show, the conditions of Theorem 3.23 hold for $(\mathbf{a}_0, \mathbf{b}_0)$. So $(\mathbf{a}_0, \mathbf{b}_0)$ is an elementary memory of (R_1, L_1). Thus, for $(\mathbf{a}_0, \mathbf{b}_0)$, (R_1, L_1) has good fault-tolerance. Also we can from Table 3.15 obtain the following facts:

(i) To the connection networks (R_2, L_2), (R_3, L_3) of (R_1, L_1), by one iteration, the state corresponding to $(\mathbf{a}_1, \mathbf{b}_1)$ belongs to the attractive neighborhoods $\mathcal{N}\big(R_2, L_2; (\mathbf{a}_0, \mathbf{b}_0)\big)$, $\mathcal{N}\big(R_3, L_3; (\mathbf{a}_0, \mathbf{b}_0)\big)$, respectively. Thus we may realize to escape from the attractor $(\mathbf{a}_1, \mathbf{b}_1)$ that is not the elementary memory of (R_1, L_1). $(\mathbf{a}_1, \mathbf{b}_1)$ may converge to $(\mathbf{a}_0, \mathbf{b}_0)$, an elementary memory of (R_1, L_1).

(ii) To the connection network (R_4, L_4), $(\mathbf{a}_1, \mathbf{b}_1)$ belongs to the attractive neighborhood $\mathcal{N}(R_4, L_4; (\mathbf{a}_0, \mathbf{b}_0))$, so the state escapes also from $(\mathbf{a}_1, \mathbf{b}_1)$.

The following two problems are very meaningful and important for the future studies:

(1′) How do we design some learning algorithms for (R, L), so that the given fuzzy patterns are the elementary memories [52, 53]?

(2′) How do we enlarge the storage capacity if the fuzzy patterns stored are the elementary memories [20, 49]?

§3.6 Equilibrium analysis of fuzzy Hopfield network

Similarly with §3.5 we in this section employ connection networks to study the attractors and attractive cycles of fuzzy Hopfield networks. A novel approach is proposed for designing the FNN's with good fault–tolerance.

Let F be a fuzzy elementary matrix. If the fuzzy Hopfield networks W_1, W_2 satisfy the condition: $W_2 = F \circ W_1$, we write also $W_1 \xrightarrow{F} W_2$. For a given fuzzy pattern $\mathbf{b} = (b_1, ..., b_n)$ and a fuzzy matrix $W = (w_{ij})_{n\times n}$, if $k \in \mathrm{N}$, $j \in \mathrm{N}$, denote

$$H_G(\mathbf{b}, j) = \{i \in \mathrm{N} | b_i > b_j\}, \ H_E(\mathbf{b}, j) = \{i \in \mathrm{N} | b_i = b_j\};$$

$$J_G(W, \mathbf{b}, j) = \{i \in \mathrm{N} | w_{ij} > b_j\}, \ J_E(W, \mathbf{b}, j) = \{i \in \mathrm{N} | w_{ij} = b_j\};$$

$$H_{GE}(\mathbf{b},j)=H_G(\mathbf{b},j)\cup H_E(\mathbf{b},j),\ J_{GE}(W,\mathbf{b},j)=J_G(W,\mathbf{b},j)\cup J_E(W,\mathbf{b},j).$$

Proposition 3.2 *Let $W^{(1)}$, $W^{(2)}$ be the $k-$multiple network (matrix) and $p-q$ commutative network (matrix) of W, respectively, and $\mathbf{b}=(b_1,...,b_n)$ be a fuzzy pattern. Then $\forall j\in \mathrm{N}$, it follows that*

(i) $J_{GE}(W^{(1)},\mathbf{b},j)\subset J_{GE}(W,\mathbf{b},j)$;

(ii) $q\in H_E(\mathbf{b},p)\Longrightarrow\ J_{GE}(W^{(2)},\mathbf{b},j)=J_{GE}(W,\mathbf{b},j)$;

(iii) $\mathrm{Card}\big(J_{GE}(W^{(2)},\mathbf{b},j)\big)=\mathrm{Card}\big(J_{GE}(W,\mathbf{b},j)\big)$.

Its proof is trivial considering the definition of $J_{GE}(W,\mathbf{b},j)$.

Proposition 3.3 *Let $W=(w_{ij})_{n\times n}$, $V=(v_{ij})_{n\times n}$ be two given networks, and $\mathbf{b}=(b_1,...,b_n)$ be a fuzzy pattern. Denote $W\cup V=(w_{ij}\vee v_{ij})_{n\times n}$, $W\cap V=(w_{ij}\wedge v_{ij})_{n\times n}$. Then $\forall j\in\mathrm{N}$, we have*

$$\begin{cases} J_G(W\cup V,\mathbf{b},j)=J_G(W,\mathbf{b},j)\cup J_G(V,\mathbf{b},j);\\ J_G(W\cap V,\mathbf{b},j)=J_G(W,\mathbf{b},j)\cap J_G(V,\mathbf{b},j).\end{cases}\tag{3.76}$$

Proof. For any $j\in\mathrm{N}$, it is easy to show

$$\begin{aligned} i\in J_G\,(W\cup V,\mathbf{b},j)&\Longleftrightarrow w_{ij}\vee v_{ij}>b_j,\\ &\Longleftrightarrow w_{ij}>b_j \text{ or } v_{ij}>b_j\Longleftrightarrow i\in J_G(W,\mathbf{b},j)\cup J_G(V,\mathbf{b},j).\end{aligned}$$

So the first part of (3.76) holds. Similarly we can show the other conclusions hold. □

Obviously, Proposition 3.3 holds for $J_{GE}(W\cup V,\mathbf{b},j)$, $J_{GE}(W\cap V,\mathbf{b},j)$.

3.6.1 Connection relations of attractors

Next let us present the connection relations between the attractors of W and ones of the connection network of W.

Theorem 3.24 *Let the fuzzy pattern $\mathbf{b}=(b_1,...,b_n)$ be an attractor of W. Then $\forall\lambda\in[0,1]$, $\mathbf{b}$ is an attractor of $\Delta_k(\lambda)\circ W$ if and only if the following conditions hold:*

$$\forall j\in\mathrm{N},\ \ \mathrm{N}^k\cap H_{GE}(\mathbf{b},j)\cap J_{GE}(W,\mathbf{b},j)\neq\emptyset.$$

Proof. For any $\lambda\in[0,1]$, let $\mathbf{b}$ be an attractor of $\Delta_k(\lambda)\circ W$, but there is $j_0\in\mathrm{N}$, so that $\mathrm{N}^k\cap H_{GE}(\mathbf{b},j_0)\cap J_{GE}(W,\mathbf{b},j_0)=\emptyset$. Let $y_{j_0}=\big(\mathbf{b}\circ(\Delta_k(\lambda)\circ W)\big)_{j_0}$. Then $\forall i\in\mathrm{N}^k, i\notin H_{GE}(\mathbf{b},j_0)\cap J_{GE}(W,\mathbf{b},j_0)$. Therefore

$$i\in\mathrm{N}^k\Longrightarrow b_i<b_{j_0}\ \text{ or }\ w_{ij_0}<b_{j_0}\Longrightarrow b_i\wedge w_{ij_0}<b_{j_0}.$$

Thus, $\bigvee\limits_{i\in\mathrm{N}^k}(b_i\wedge w_{ij_0})<b_{j_0}$. If letting $\lambda<b_{j_0}$, we have

$$y_{j_0}=\Big(\bigvee_{i\in\mathrm{N}^k}\{b_i\wedge w_{ij_0}\}\Big)\vee\big(b_k\wedge\lambda\wedge w_{kj_0}\big)<b_{j_0}.$$

So $\mathbf{b}$ is not an attractor of $\Delta_k(\lambda) \circ W$, which is a contradiction. Hence the conditions of the theorem hold.

Conversely, with the conditions of the theorem, $\forall j \in \mathrm{N}$, we have, $i_0 \in \mathrm{N}^k \cap H_{GE}(\mathbf{b}, j) \cap J_{GE}(W, \mathbf{b}, j)$. So $i_0 \in \mathrm{N}^k$, $b_{i_0} \geq b_j$, $w_{i_0 j} \geq b_j$. By Proposition 3.2 we get, $\mathbf{b} = \mathbf{b} \circ W \supset \mathbf{b} \circ (\Delta_k(\lambda) \circ W)$. Therefore

$$b_j \geq \big(\mathbf{b} \circ (\Delta_k(\lambda) \circ W)\big)_j = \Big(\bigvee_{i \in \mathrm{N}^k} \{b_i \wedge w_{ij}\}\Big) \vee (\lambda \wedge w_{kj} \wedge b_k) \geq b_{i_0} \wedge w_{i_0 j} \geq b_j,$$

i.e. $\big(\mathbf{b} \circ (\Delta_k(\lambda) \circ W)\big)_j = b_j$. Thus, $\mathbf{b}$ is an attractor of $\Delta_k(\lambda) \circ W$. □

By Theorem 3.24 the following result is trivial.

Corollary 3.3 *Let the fuzzy pattern* $\mathbf{b}$ *be an attractor of* W. *Then* $\forall \lambda \in [0, 1]$, $\mathbf{b}$ *is not an attractor of* $\Delta_k(\lambda) \circ W$ *if and only if there is* $j_0 \in \mathrm{N}^k$, *so that* $H_{GE}(\mathbf{b}, j_0) \cap J_{GE}(W, \mathbf{b}, j_0) = \{k\}$ *or,* $H_{GE}(\mathbf{b}, j_0) \cap J_{GE}(W, \mathbf{b}, j_0) = \emptyset$.

For $p, q \in \mathrm{N}$, and $j \in \mathrm{N}$, we denote $S_j(p, q)$, $R_j(p, q)$ respectively as follows:

$$S_j(p, q) = \bigvee_{i \in \mathrm{N}^p \cap N^q} \{b_i \wedge w_{ij}\}, \quad R_j(p, q) = (w_{pj} \wedge b_q) \vee (w_{qj} \wedge b_p).$$

Theorem 3.25 *Let* $\mathbf{b} = (b_1, ..., b_n)$ *be an attractor of* W, *and* $p, q \in \mathrm{N}$ *satisfy the condition:* $b_p > b_q$; $\forall j \in \mathrm{N}$, *either* $b_j \geq b_p$, *or* $b_j \leq b_q$. *Then* $\mathbf{b}$ *is also an attractor of* $\Delta_{pq} \circ W$ *if and only if* $\forall j \in \mathrm{N}$, *one of the following facts holds:*

(i) $S_j(p, q) \geq b_j$, $R_j(p, q) \leq b_j$;

(ii) $$\begin{cases} 1 - \chi_{H_E(\mathbf{b}, p)}(j) + w_{qj} \geq b_p, \\ \chi_{H_E(\mathbf{b}, q)}(j) - 1 + w_{qj} \leq b_q. \end{cases}$$

Proof. Sufficiency: For any $j \in \mathrm{N}$, suppose the condition (i) is true. Using the following fact:

$$b_j = \bigvee_{i \in \mathrm{N}} \{w_{ij} \wedge b_i\} = S_j(p, q) \vee (w_{pj} \vee b_p) \vee (w_{qj} \vee b_q).$$

we get, $S_j(p, q) = b_j$. So $S_j(p, q) \vee R_j(p, q) = b_j$, i.e. $\big(\mathbf{b} \circ (\Delta_{pq} \circ W)\big)_j = b_j$.

If the condition (ii) is true, by the assumptions it suffices to prove the conclusion in the following four cases, respectively:

$$\text{(i}'\text{)}\ b_j > b_p; \quad \text{(ii}'\text{)}\ b_j = b_p; \quad \text{(iii}'\text{)}\ b_j = b_q; \quad \text{(iv}'\text{)}\ b_j < b_q.$$

To the case (i′), $b_j > b_p > b_q$, so $(b_p \wedge w_{pj}) \vee (b_q \wedge w_{qj}) < b_j$, and $R_j(p, q) < b_j$. Since $\mathbf{b} = \mathbf{b} \circ W$, it follows that $S_j(p, q) = b_j$. Hence

$$\big(\mathbf{b} \circ (\Delta_{pq} \circ W)\big)_j = S_j(p, q) \vee R_j(p, q) = b_j. \tag{3.77}$$

To the case (ii′), $j \in H_E(\mathbf{b}, p)$, so by (ii) we get

$$\chi_{H_E(\mathbf{b},p)}(j) = 1, \quad w_{qj} \geq b_p. \tag{3.78}$$

Since $\mathbf{b}$ is an attractor of W, it follows that

$$b_j = \bigvee_{i \in \mathrm{N}} \{b_i \wedge w_{ij}\}. \tag{3.79}$$

Therefore by (3.78) (3.79) and the fact: $b_j = b_p$, we get

$$\begin{aligned} b_j &= S_j(p,q) \vee \big(w_{pj} \wedge b_p\big) \vee \big(w_{qj} \wedge b_q\big) \\ &\leq S_j(p,q) \vee \big(w_{pj} \wedge b_q\big) \vee \big(w_{qj} \wedge b_p\big) = b_p = b_j. \end{aligned} \tag{3.80}$$

Thus, $\big(\mathbf{b} \circ (\Delta_{pq} \circ W)\big)_j = S_j(p,q) \vee R_j(p,q) = b_j$, (3.77) is true.

To the case (iii′), $j \in H_E(\mathbf{b}, q)$, then by the condition (ii), (3.79) and the inequality: $b_p > b_q$ we have

$$\chi_{H_E(\mathbf{b},q)}(j) = 1, w_{qj} \leq b_q. \tag{3.81}$$

Moreover, $w_{pj} \leq b_j$. Therefore

$$\begin{aligned} b_j &= S_j(p,q) \vee \big(w_{pj} \wedge b_p\big) \vee \big(w_{qj} \wedge b_q\big) \\ &= S_j(p,q) \vee \big(w_{pj} \wedge b_q\big) \vee \big(w_{qj} \wedge b_p\big) = \big(\mathbf{b} \circ (\Delta_{pq} \circ W)\big)_j. \end{aligned} \tag{3.82}$$

To the case (iv′), $b_j < b_q$, then $w_{pj} \wedge b_p < b_q < b_p$, $w_{qj} \wedge b_q < b_q < b_p$. So

$$\big(w_{pj} \wedge b_p\big) \vee \big(w_{qj} \wedge b_q\big) = w_{pj} \vee w_{qj} = R_j(p,q).$$

Hence we obtain the following fact:

$$b_j = S_j(p,q) \vee \Big(\bigvee_{i \in \{p,q\}} \{b_i \wedge w_{ij}\} \Big) = S_j(p,q) \vee R_j(p,q) = \big(\mathbf{b} \circ (\Delta_{pq} \circ W)\big)_j. \tag{3.83}$$

By (3.77) (3.80) (3.82) and (3.83) we imply, $b_j = \big(\mathbf{b} \circ (\Delta_{pq} \circ W)\big)_j$. Thus, $\forall j \in \mathrm{N}$, if either the condition (i) or, (ii) holds, we have, $b_j = \big(\mathbf{b} \circ (\Delta_{pq} \circ W)\big)_j$, that is, $\mathbf{b} = \mathbf{b} \circ (\Delta_{pq} \circ W)$, $\mathbf{b}$ is the attractor of $\Delta_{pq} \circ W$.

Necessity: Let $\mathbf{b}$ be an attractor of $\Delta_{pq} \circ W$. Then

$$b_j = S_j(p,q) \vee \big(w_{pj} \wedge b_p\big) \vee \big(w_{qj} \wedge b_q\big) = S_j(p,q) \vee \big(w_{pj} \wedge b_q\big) \vee \big(w_{qj} \wedge b_p\big). \tag{3.84}$$

If there is $j \in \mathrm{N}$, not satisfying (i), then $S_j(p,q) < b_j$ or, $R_j(p,q) > b_j$. By $b_j = S_j(p,q) \vee R_j(p,q)$, we obtain, $S_j(p,q) < b_j$. Next we show the condition (ii) at above four cases (i′) (ii′) (iii′) and (iv′), respectively.

$b_p = b_j$, then by $S_j(p,q) < b_j$ and $b_j = b_p > b_q$, and (3.84) we imply, $w_{qj} \geq b_p$. Thus,

$$1 - \chi_{H_E(\mathbf{b},p)}(j) + w_{qj} = w_{qj} \geq b_p > b_q.$$

If $b_j = b_q$, also by (3.84) it follows that $w_{qj} \leq b_q$. So $\chi_{H_E(\mathbf{b},q)}(j) - 1 + w_{qj} = w_{qj} \leq b_q$. As for the other cases, the condition (ii) is obviously true. Therefore, $\forall j \in \mathrm{N}$, (i) or, (ii) holds. □

Corollary 3.4 *Let $\mathbf{b} = (b_1, ..., b_n)$ be a fuzzy pattern. And $p, q \in \mathrm{N}$, $k \in \mathrm{N}^p \cap N^q$. The following conditions hold:*
(i) $b_p > b_q$, $\forall j \in \mathrm{N}$, either $b_j \geq b_p$ or, $b_j \leq b_q$;
(ii) $\lambda \in [0,1]$, $\mathbf{b}$ is the attractor of both W, $\Delta_{pq} \circ W$ and $\Delta_k(\lambda) \circ W$.
Then we can conclude that
(1′) $\mathbf{b}$ is an attractor of $\Delta_{pq} \circ \Delta_k(\lambda) \circ W$;
(2′) $\mathbf{b}$ is an attractor of $\Delta_k(\lambda) \circ \Delta_{pq} \circ W$.

Proof. (1′) Suppose $\Delta_k(\lambda) \circ W \triangleq W^{(1)} = (w^1_{ij})_{n\times n}$. Then

$$w^1_{ij} = \begin{cases} w_{ij}, & i \neq k, \\ w_{ij} \wedge \lambda, & i = k. \end{cases}$$

By the assumption and Theorem 3.24, $\mathbf{b}$ is an attractor of $W^{(1)}$, moreover

$$\begin{cases} 1 - \chi_{H_E(\mathbf{b},p)}(j) + w_{qj} \geq b_p; \\ \chi_{H_E(\mathbf{b},q)}(j) - 1 + w_{qj} \leq b_q. \end{cases} \tag{3.85}$$

Since $k \in \mathrm{N}^p \cap N^q$, we have, $k \neq p,\ q$, and

$$\begin{cases} 1 - \chi_{H_E(\mathbf{b},p)}(j) + w^1_{qj} \geq b_p; \\ \chi_{H_E(\mathbf{b},q)}(j) - 1 + w^1_{qj} \leq b_q. \end{cases}$$

So Theorem 3.25 implies, $\mathbf{b}$ is an attractor of $\Delta_{pq} \circ W^{(1)}$, that is, $\mathbf{b}$ is an attractor of $\Delta_{pq} \circ \Delta_k(\lambda) \circ W$.

(2′) Denote $W^{(2)} = (w^2_{ij})_{n\times n}$. Then $\forall j \in \mathrm{N}$, we can easily show

$$J_{GE}(W^{(2)}, \mathbf{b}, j) = \begin{cases} J_{GE}(W, \mathbf{b}, j), & j \neq p, q; \\ J_{GE}(W, \mathbf{b}, p), & j = q; \\ J_{GE}(W, \mathbf{b}, q), & j = p. \end{cases} \tag{3.86}$$

Since for any $\lambda \in [0,1]$, $\mathbf{b}$ is the attractor of $\Delta_k(\lambda) \circ W$, by Theorem 3.24 it follows that

$$\forall j \in \mathrm{N},\ N^k \cap H_{GE}(\mathbf{b}, j) \cap J_{GE}(W, \mathbf{b}, j) \neq \emptyset;$$
$$\forall j \in \mathrm{N},\ N^k \cap H_{GE}(\mathbf{b}, j) \cap J_{GE}(W^{(2)}, \mathbf{b}, j) \neq \emptyset.$$

So using Theorem 3.24 we get, $\forall\lambda \in [0,1]$, $\mathbf{b}$ is the attractor of $\Delta_k(\lambda) \circ W^{(2)} = \Delta_k(\lambda) \circ \Delta_{pq} \circ W$. □

3.6.2 Elementary memory of W

It is a very important theoretic and applied problem for FAM's to discriminate the pseudo-attractors from the attractors of FAM's. A fuzzy pattern $\mathbf{b}'$ is called a pseudo-attractor of W if $\mathbf{b}' = \mathbf{b}' \circ W$, but $\mathbf{b}'$ is not a fuzzy pattern stored. By the methods proposed in the subsection we may develop some novel ways to do that by the elementary memories of W.

Definition 3.11 We call the fuzzy pattern $\mathbf{b}$ an elementary memory of W, if the following conditions hold:

(i) For any $k \in \mathrm{N}$, and $\lambda \in [0,1]$, $\mathbf{b}$ is an attractor of $\Delta_k(\lambda) \circ W$;

(ii) $\forall p,\ q \in \mathrm{N}$, $\mathbf{b}$ is an attractor of $\Delta_{pq} \circ W$.

Obviously, if $\mathbf{b}$ is an elementary memory of W, $\mathbf{b}$ is an attractor of W. Next we establish some equivalent conditions that an attractor of W is also an elementary memory.

Theorem 3.26 *Let the fuzzy pattern* $\mathbf{b}$ *be an attractor of* W. *Then* $\forall k \in \mathrm{N}$, $\lambda \in [0,1]$, $\mathbf{b}$ *is also an attractor of* $\Delta_k(\lambda) \circ W$ *if and on if*

$$\forall j \in \mathrm{N},\ \mathrm{Card}\big(H_{GE}(\mathbf{b}, j) \cap J_{GE}(W, \mathbf{b}, j)\big) \geq 2. \tag{3.87}$$

Proof. Sufficiency: For any $j \in \mathrm{N}$, suppose (3.87) is true, choose $i_1,\ i_2 \in H_{GE}(\mathbf{b}, j) \cap J_{GE}(W, \mathbf{b}, j)$: $i_1 \neq i_2$. Then

$$w_{i_m j} \geq b_j,\ b_{i_m} \geq b_j\ (m = 1, 2). \tag{3.88}$$

Given arbitrarily $k \in \mathrm{N}$, $\lambda \in [0,1]$, by (3.88), Proposition 3.2 and the assumption it follows that

$$\begin{aligned} k \notin \{i_1, i_2\} \Longrightarrow b_j &\geq \big(\mathbf{b} \circ (\Delta_k(\lambda) \circ W)\big)_j \geq \\ &\geq \big(b_{i_1} \wedge w_{i_1 j}\big) \vee \big(b_{i_2} \wedge w_{i_2 j}\big) \geq b_j. \end{aligned} \tag{3.89}$$

If $k \in \{i_1, i_2\}$, we may assume $k = i_1$. Similarly with (3.89), we have

$$b_j \geq \big(\mathbf{b} \circ (\Delta_k(\lambda) \circ W)\big)_j \geq b_{i_2} \wedge w_{i_2 j} \geq b_j.$$

In summary, $\big(\mathbf{b} \circ (\Delta_k(\lambda) \circ W)\big)_j = b_j$ $(j \in \mathrm{N})$, i.e. $\forall k \in \mathrm{N}$, $\lambda \in [0,1]$, $\mathbf{b}$ is an attractor of $\Delta_k(\lambda) \circ W$.

Necessity: At first, for any $j \in \mathrm{N}$, it is easy to show

$$i \in H_{GE}(\mathbf{b}, j) \cap J_{GE}(W, \mathbf{b}, j), \Longleftrightarrow w_{ij} \wedge b_i \geq b_j. \tag{3.90}$$

We use reduction to absurdity to show (3.87). If (3.87) is false, then there is $j_0 \in \mathrm{N}$, so that $\mathrm{Card}\big(H_{GE}(\mathbf{b}, j_0) \cap J_{GE}(W, \mathbf{b}, j_0)\big) \leq 1$. So by (3.90), either there is $i_0 \in \mathrm{N}$, satisfying $w_{i_0j_0} \wedge b_{i_0} \geq b_{j_0}$, and $\forall i \in \mathrm{N}^{i_0}$, $w_{ij_0} \wedge b_i < b_{j_0}$, or $\forall i \in \mathrm{N}$, $w_{ij_0} \wedge b_i < b_{j_0}$. Choose $k = j_0$, $\lambda < b_{j_0}$, then

$$\big(\mathbf{b} \circ (\Delta_k(\lambda) \circ W)\big)_{j_0} = \Big(\bigvee_{i \in \mathrm{N}^{i_0}} \{w_{ij_0} \wedge b_i\}\Big) \vee \big(\lambda \wedge w_{i_0j_0} \wedge b_{i_0}\big) < b_{j_0}.$$

Thus, $\mathbf{b} \neq \mathbf{b} \circ (\Delta_k(\lambda) \circ W)$, which is a contradiction. So (3.87) is true. □

By the induction method and Theorem 3.26 the following result is trivial.

Corollary 3.5 *Suppose* $\mathbf{b}$ *is an attractor of* W*, and* $m \in \mathrm{N}$*, satisfying* $m \leq n-2$*. Then for any* $k_1, ..., k_m \in \mathrm{N}; \lambda_1, ..., \lambda_m \in [0,1]$*,* $\mathbf{b}$ *is an attractor of* $\Delta_{k_1}(\lambda_1) \circ \cdots \circ \Delta_{k_m}(\lambda_m) \circ W$ *if and only if,* $\forall j \in \mathrm{N}$*, it follows that* $\mathrm{Card}\big(H_{GE}(\mathbf{b}, j) \cap J_{GE}(W, \mathbf{b}, j)\big) \geq m+1$.

By Corollary 3.5, If the $m-$th component of $\mathbf{b}$ changes in [0, 1] and other components unchange, then the new fuzzy pattern related converges with one iteration to $\mathbf{b}$, which shows us good fault–tolerance of the fuzzy Hopfield networks.

Theorem 3.27 *Suppose* $\mathbf{b} = (b_1, ..., b_n)$ *is a fuzzy pattern, and for any* $j \in \mathrm{N}$*, the following conditions hold:*

(i) $J_G(W, \mathbf{b}, j) \times H_G(\mathbf{b}, j) = \emptyset$;

(ii) $\mathrm{Card}\big(J_{GE}(W, \mathbf{b}, j) \cap H_{GE}(\mathbf{b}, j)\big) \geq 2$.

Then $\forall p, q \in \mathrm{N}$*,* $\mathbf{b}$ *is the attractor of* $\Delta_{pq} \circ W$*, and therefore* $\mathbf{b}$ *is an elementary memory of* W*.*

Conversely, if for any $p, q \in \mathrm{N}$*,* $\mathbf{b}$ *is an attractor of* $\Delta_{pq} \circ W$*, then* $\forall j \in \mathrm{N}$*, above (i) and the following (iii) hold:*

(iii) $\mathrm{Card}\big(J_{GE}(W, \mathbf{b}, j) \cap H_{GE}(\mathbf{b}, j)\big) \geq 1$.

Proof. For any $j \in \mathrm{N}$, by the assumptions (i) (ii) are true. $\forall p, q \in \mathrm{N}$, by condition (i), easily we have

$$\forall i_1, i_2 \in \mathrm{N}, \ b_{i_1} \wedge w_{i_2j} \leq b_j. \tag{3.91}$$

And by (3.91), $\forall i_1, i_2 \in J_{GE}(W, \mathbf{b}, j) \cap H_{GE}(\mathbf{b}, j)$, it follows that

$$b_{i_1} \wedge w_{i_2j} = b_j. \tag{3.92}$$

Using the condition (ii), we choose $i', i'' \in \mathrm{N}$ satisfying $i' \neq i''$, so that $i', i'' \in J_{GE}(W, \mathbf{b}, j) \cap H_{GE}(\mathbf{b}, j)$.

If $\{i', i''\} \cap \{p, q\} = \emptyset$, Then (3.91) (3.92) imply

$$S_j(p,q) = \bigvee_{i \in \mathrm{N}^p \cap N^q} \{b_i \wedge w_{ij}\} = b_j; \ R_j(p,q) = \big(b_p \wedge w_{qj}\big) \vee \big(b_q \wedge w_{pj}\big) \leq b_j.$$

Therefore

$$\big(\mathbf{b} \circ (\Delta_{pq} \circ W)\big)_j = S_j(p,q) \vee R_j(p,q) = b_j. \tag{3.93}$$

If $\{i', i''\} \cap \{p, q\} \neq \emptyset$, it is no harm to assume $i' = p, i'' = q$. By (3.92), $R_j(p,q) = \big(b_p \wedge w_{qj}\big) \vee b_q \wedge w_{pj}\big) = b_j$. So it follows that $S_j(p,q) \leq b_j$. Thus, (3.93) is true. Also we may assume $i' = p$, $q \notin \{i', i''\}$. By (3.91) (3.92) it follows that

$$R_j(p,q) = \big(b_{i'} \wedge w_{qj}\big) \vee \big(b_q \wedge w_{i'j}\big) \leq b_j;$$
$$S_j(p,q) = \bigvee_{i \in \mathrm{N}^p \cap N^q} \{b_i \wedge w_{ij}\} \geq b_{i''} \wedge w_{i''j} = b_j.$$

On the other hand, by (3.91) we get, $S_j(p,q) \leq b_j$. Thus, $\big(\mathbf{b} \circ (\Delta_{pq}) \circ W\big)_j = S_j(p,q) \vee R_j(p,q) = b_j$, i.e. (3.93) is true. In summary, $\big(\mathbf{b} \circ (\Delta_{pq} \circ W)\big)_j = b_j \ (j \in \mathrm{N}), \Longrightarrow \forall p,\ q \in \mathrm{N}$, $\mathbf{b}$ is the attractor of $\Delta_{pq} \circ W$.

Conversely, assume that for $p,\ q \in \mathrm{N}$, $\mathbf{b}$ is an attractor of $\Delta_{pq} \circ W$. Obviously, if $p = q$, then $\mathbf{b}$ ia an attractor of W. If there is $j_0 \in \mathrm{N}$, so that the condition (i) is false, then let $(i_1, i_2) \in J_G(W, \mathbf{b}, j_0) \times H_G(\mathbf{b}, j_0)$. Therefore, $w_{i_1 j_0} > b_{j_0}$, $b_{i_2} > b_{j_0}$. Choose $p = i_1$, $q = i_2$. Then

$$R_{j_0}(p,q) = \big(w_{i_1 j_0} \wedge b_{i_2}\big) \vee \big(w_{i_2 j_0} \wedge b_{i_1}\big) > b_{j_0}.$$

Thus, $\big(\mathbf{b} \circ (\Delta_{pq} \circ W)\big)_{j_0} \geq R_{j_0}(p,q) > b_{j_0}$. hence $\mathbf{b}$ is not the attractor of $\Delta_{pq} \circ W$, which is a contradiction. So $\forall j \in \mathrm{N}$, the condition (i) holds.

Assume that (iii) does not hold, that is, there exists $j_0 \in \mathrm{N}$, so that (iii) is false, then $\mathrm{Card}\big(J_{GE}(W, \mathbf{b}, j_0) \cap H_{GE}(\mathbf{b}, j_0)\big) = 0$. Also it is easy to show, $\forall i \in \mathrm{N}$, $w_{ij_0} < b_{j_0}$ or, $b_i < b_{j_0}$. So $\bigvee_{i \in \mathrm{N}} \{b_i \wedge w_{ij_0}\} < b_{j_0}$, which contradict the fact also. Thus, $\mathbf{b}$ is an attractor of W. Hence the condition (iii) holds. □

3.6.3 The state transitive laws

For the given fuzzy matrix $W = (w_{ij})_{n \times n}$, define the transformation $T_W : [0,1]^n \longrightarrow [0,1]^n$, satisfying

$$\forall \mathbf{x} \in [0,1],\ T_W(\mathbf{x}) = \mathbf{x} \circ W. \tag{3.94}$$

Also for the fuzzy matrices W_1, W_2, define the transformation $T_{W_1} \circ T_{W_2} : [0,1]^n \longrightarrow [0,1]^n$ as follows:

$$\forall \mathbf{x} \in [0,1]^n,\ (T_{W_1} \circ T_{W_2})(\mathbf{x}) = T_{W_1}(T_{W_2}(\mathbf{x})) \tag{3.95}$$

Lemma 3.5 *Let W_1, W_2 be fuzzy matrices. Then $T_{W_1 \circ W_2} = T_{W_2} \circ T_{W_1}$.*

Proof. For any $\mathbf{x} \in [0,1]^n$, by (3.94) (3.95) it follows that

$$\begin{aligned} T_{W_1 \circ W_2}(\mathbf{x}) &= \mathbf{x} \circ (W_1 \circ W_2) = (\mathbf{x} \circ W_1) \circ W_2 \\ &= T_{W_1}(\mathbf{x}) \circ W_2 = T_{W_2}(T_{W_1}(\mathbf{x})) = (T_{W_2} \circ T_{W_1})(\mathbf{x}). \end{aligned}$$

So $T_{W_1 \circ W_2} = T_{W_2} \circ T_{W_1}$. □

If F is a fuzzy matrix, we call T_F a fuzzy elementary transformation.

Theorem 3.28 *Let the fuzzy Hopfield networks W_1, W_2 satisfy: $W_1 \xrightarrow{F} W_2$. Suppose T_F is a fuzzy elementary transformation, so that there exist fuzzy row-restricted matrices $\Delta_1, ..., \Delta_m$: $F = \Delta_m \circ \Delta_{m-1} \circ \cdots \circ \Delta_2 \circ \Delta_3$. Then*

$$T_{W_2} = T_{W_1} \circ T_F = T_{W_1} \circ T_{\Delta_1} \circ \cdots \circ T_{\Delta_m}. \tag{3.96}$$

Proof. By Lemma 3.5 it follows that

$$\begin{aligned} T_{W_2} &= T_{F \circ W_1} = T_{(\Delta_m \circ \cdots \circ \Delta_2 \circ \Delta_1) \circ W_1} \\ &= T_{W_1} \circ (T_{\Delta_m \circ \cdots \circ \Delta_2 \circ \Delta_1}) = T_{W_1} \circ (T_{\Delta_1} \circ (T_{\Delta_m \circ \cdots \circ \Delta_2})) \\ &= \cdots\cdots\cdots \\ &= T_{W_1} \circ (T_{\Delta_1} \circ (T_{\Delta_2} \circ \cdots \circ (T_{\Delta_m} \cdots))) = T_{W_1} \circ T_{\Delta_1} \circ \cdots \circ T_{\Delta_m}. \end{aligned}$$

Thus, (3.96) is true for the fuzzy elementary transformation T_F. □

Suppose the fuzzy pattern $\mathbf{b}$ is an attractor of W. Moreover, $\mathcal{N}(W; \mathbf{b}) = \{\mathbf{x} \in [0,1]^n | \mathbf{x} \circ W = \mathbf{b}\}$. Then for any $\mathbf{x} \in \mathcal{N}(W; \mathbf{b})$, X converges with one iteration to $\mathbf{b}$. Therefore, $\mathcal{N}(W; \mathbf{b})$ is an attractive neighborhood of $\mathbf{b}$. If the sequence of fuzzy patterns $\mathbf{b}_1, ..., \mathbf{b}_l$ is the limit cycle of W, that is

$$\mathbf{b}_2 = \mathbf{b}_1 \circ W, ..., \mathbf{b}_l = \mathbf{b}_{l-1} \circ W, \ \mathbf{b}_1 = \mathbf{b}_l \circ W.$$

Denote a collection of fuzzy patterns by

$$\mathcal{N}(W; \mathbf{b}_1, ..., \mathbf{b}_l) = \{\mathbf{x} \in [0,1]^n | \exists k \in \{1, ..., l\} : \ \mathbf{x} \circ W = \mathbf{b}_k\}.$$

we call $\mathcal{N}(W; \mathbf{b}_1, ..., \mathbf{b}_l)$ an attractive neighborhood of limit cycle $\{\mathbf{b}_1, ..., \mathbf{b}_l\}$.

Theorem 3.29 *Let $\mathbf{b}$ be an attractor of W_1, and $W_1 \xrightarrow{F} W_2$. Then $\mathbf{b}$ is also an attractor of W_2 if and only if $T_F(\mathbf{b}) \in \mathcal{N}(W_1; \mathbf{b})$.*

Proof. Suppose $\mathbf{b}$ is an attractor of W_2. Then $\mathbf{b} \circ W_2 = \mathbf{b}$, that is, $T_{W_2}(\mathbf{b}) = \mathbf{b}$. By Lemma 3.5 it follows that

$$\mathbf{b} = T_{W_2}(\mathbf{b}) = (T_{F \circ W_1}(\mathbf{b}) = (T_{W_1} \circ T_F)(\mathbf{b}) = T_{W_1}(T_F(\mathbf{b})) = T_F(\mathbf{b}) \circ W_1.$$

That is, $T_F(\mathbf{b}) \circ W_1 = \mathbf{b}$. So $T_F(\mathbf{b}) \in \mathcal{N}(W_1; \mathbf{b})$.

Conversely, if $T_F(\mathbf{b}) \in \mathcal{N}(W_1; \mathbf{b})$, by Lemma 3.5 we can conclude that

$$\begin{aligned} \mathbf{b} &= T_F(\mathbf{b}) \circ W_1 = T_{W_1}(T_F(\mathbf{b})) = (T_{W_1} \circ T_F)(\mathbf{b}) \\ &= T_{F \circ W_1}(\mathbf{b}) = T_{W_2}(\mathbf{b}) = \mathbf{b} \circ W_2. \end{aligned}$$

So $\mathbf{b}$ is an attractor of W_2. □

Theorem 3.30 *Let the fuzzy pattern family* $\{\mathbf{b}_1, ..., \mathbf{b}_l\}$ *be the attractive cycle of* W_1, *and* $W_1 \xrightarrow{F} W_2$. *If the family is also the attractive cycle of* W_2, *then* $\forall k \in \{1, ..., l\}$, $T_F(\mathbf{b}_k) \in \mathcal{N}(W_1; \mathbf{b}_1, ..., \mathbf{b}_l)$. *Conversely, let*

$$T_F(\mathbf{b}_k) \triangleq \begin{cases} \mathbf{b}'_k \in \mathcal{N}(W_1; \mathbf{b}_{k+1}), & 1 \le k < l; \\ \mathbf{b}'_k \in \mathcal{N}(W_1; \mathbf{b}_1), & k = l. \end{cases}$$

Then the fuzzy pattern family $\{\mathbf{b}_1, ..., \mathbf{b}_l\}$ *is the attractive cycle of* W_2.

Proof. Assume that the family $\{\mathbf{b}_1, ..., \mathbf{b}_l\}$ is the attractive cycle of W_2. Then $\forall k \in \{1, ..., l\}$, it follows that

$$\begin{aligned} T_F(\mathbf{b}_k) \circ W_1 &= T_{W_1}(T_F(\mathbf{b})) = (T_{W_1} \circ T_F)(\mathbf{b}_k) = T_{F \circ W_1}(\mathbf{b}_k) \\ &= T_{W_2}(\mathbf{b}_k) = \mathbf{b}_k \circ W_2 = \begin{cases} \mathbf{b}_{k+1}, & 1 \le k < l, \\ \mathbf{b}_1, & k = l. \end{cases} \end{aligned}$$

Therefore, $T_F(\mathbf{b}_k) \in \mathcal{N}(W; \mathbf{b}_1, ..., \mathbf{b}_l)$. Conversely, it is easy to show

$$\mathbf{b}_1 \circ W_2 = \mathbf{b}_1 \circ (F \circ W_1) = T_{F \circ W_1}(\mathbf{b}_1) = T_{W_1} \circ (T_F(\mathbf{b}_1)) = \mathbf{b}'_1 \circ W_1 = \mathbf{b}_2.$$

With the same reason, we have, $\mathbf{b}_2 \circ W_2 = \mathbf{b}_3, ..., \mathbf{b}_{l-1} \circ W_2 = \mathbf{b}_l$, $\mathbf{b}_l \circ W_2 = \mathbf{b}_1$. That is, $\{\mathbf{b}_1, ..., \mathbf{b}_l\}$ is the attractive cycle of W_2. □

Next let us take an example to demonstrate above results. Let N = {1, 2, 3, 4}, moreover

$$W_1 = \begin{pmatrix} 0.7 & 0.5 & 0.6 & 0.4 \\ 0.4 & 0.5 & 0.7 & 0.5 \\ 0.6 & 0.4 & 0.7 & 0.5 \\ 0.5 & 0.5 & 0.6 & 0.4 \end{pmatrix}.$$

And denote

$$\mathbf{b}_0 = (0.6, 0.5, 0.6, 0.5), \quad \mathbf{b}_1 = (0.7, 0.5, 0.6, 0.5).$$

Table 3.16 Attractors of connection network

W_i	$W_i : W_i = F_i \circ W_1$	attractor
W_1	$F_1 = \Delta$	$\mathbf{b}_0$; $\mathbf{b}_1$
W_2	$F_2 = \Delta_1(0.5)$	$\mathbf{b}_0$
W_3	$F_3 = \Delta_{14}$	$\mathbf{b}_0$
W_4	$F_4 = \Delta_{23}$	$\mathbf{b}_0$; $\mathbf{b}_1$

It is easy to show that the conditions of Theorem 3.30 hold for $\mathbf{b}_0$. So $\mathbf{b}_0$ is a fuzzy elementary memory of W_1. Thus, for $\mathbf{b}_0$, W_1 possesses better fault-tolerance. Also we can from Table 3.16 get the following facts:

(i) To the connection networks W_2, W_3 related to W_1, by one iteration, the states of W_2, W_3 corresponding to $\mathbf{b}_1$ belong to $\mathcal{N}(W_2;\mathbf{b}_0)$, $\mathcal{N}(W_3;\mathbf{b}_0)$, respectively. Thus, we can realize to escape from the attractor $\mathbf{b}_1$ that is not a fuzzy elementary memory of W_1. $\mathbf{b}_1$ may converge to $\mathbf{b}_0$, a fuzzy elementary memory of W_1.

(ii) To the connection network W_4, $\mathbf{b}_1$ is also an attractor of W_4, so the state of W_4 can not escape from $\mathbf{b}_1$.

In the chapter, based on the fuzzy operator pair ($\vee$, $\wedge$) we present the basic results concerning the research on feedback FNN's. They include, some systematic conclusions related to dynamic FNN's, such as stability, attractor, attractive basin, and the iteration laws of the system states and so on; fault–tolerance of the systems, and the learning algorithms for the connection weight matrices related, etc. These results can widely be applied in information processing, especially in information restoration. To apply the FNN's in more application fields, it is necessary to generalize the fuzzy operator pair ($\vee$, $\wedge$), since it is not optimal in many applications. So within a general framework developing a systematic approach to the feedback FNN's is important and meaningful.

References

[1] Blanco A., Delgado M. & Pegalajar M.C., Identification of fuzzy dynamic systems using max-min recurrent neural networks, *Fuzzy Sets and Systems,* 2001, **122**(3): 451–467.

[2] Blanco A., Delgado M. & Requena I., Identification of relational equations by fuzzy neural networks, *Fuzzy Sets and Systems,* 1995, **71**: 215–226.

[3] Blanco A., Delgado M. & Requena I., Improved fuzzy neural networks for solving relational equations, *Fuzzy Sets and Systems,* 1995, **72**: 311–322.

[4] Buckley J. J. & Hayashi Y., Fuzzy neural networks: A survey, *Fuzzy Sets and Systems,* 1994, **66**: 1–13.

[5] Buckley J. J. & Hayashi Y., Fuzzy genetic algorithm and applications, *Fuzzy Sets and Systems,* 1994, **61**: 129–136.

[6] Chan H. Y. & Zak S. H., Chaotic neural fuzzy associative memory, *Internat. J. Bifurcation and Chaos in Appl. Sci. Eng.,* 1999, **9**(8): 1597–1617.

[7] Chang H. -C. & Chen H. -C., Lithology determination from well logs with fuzzy associative memory neural network, *IEEE Trans. on Geoscience and Remote Sensing,* 1997, **35**(3): 773–780.

[8] Chung F. L. & Lee T., On fuzzy associative memory with multiple-rule storage capacity, *IEEE Trans. on Fuzzy Systems,* 1996, **4**: 375–384.

[9] Chung F. L. & Lee T., Fuzzy Competitive learning, *Neural Networks,* 1994, **7**: 539–552.

[10] Ding H. & Gupta M. M., Competitive and cooperative adaptive reasoning with fuzzy causal knowledge, *Journal of Intelligence & Fuzzy Systems,* 2000, **9** (3-4): 191–206.

[11] Fan Junjie, Jin Fan & Shi Yan, An efficient learning algorithm of fuzzy associative memory, *Acta Elect. Sinica*(In Chinese), 1996, **24**(1): 112–114.

[12] Fan Z. T. & Liu De-Fu, Convergence of the power sequence of a nearly monotone increasing fuzzy matrix, *Fuzzy Sets and Systems,* 1997, **88**: 363–372.

[13] Gupta M. M. & Fao D. H., On the principles of fuzzy neural networks, *Fuzzy Sets and Systems,* 1994, **61**: 1–18.

[14] Gupta M. M. & Qi J., Design of fuzzy logic controllers based on generalized T-operators, *Fuzzy Sets and Systems,* 1991, **40**: 473–489.

[15] Hassoun M. H. & Watta P. B., Exact associative neural memory dynamics utilizing Boolean matrices, *IEEE Trans. on Neural Networks,* 1991, **2**: 437–448.

[16] Huang Jiejun & Xie Weixin, Regular fuzzy associative memory, *Acta Electron. Sinica*(In Chinese), 1997, **25**(7): 68–73.

[17] Ishibuchi H., Kwon K. & Tanaka H., A learning algorithm of fuzzy neural networks with triangular fuzzy weights, *Fuzzy Sets and Systems,* 1995, **71**: 277–293.

[18] Jang J. S. R., Sun C. T. & Mizutani E., *Neoro-fuzzy and soft computing,* Prentice Hall, 1997.

[19] Jiao Licheng, *Theory of neural network system,* Xian: Press of University of Electronic Science and Technology, 1996.

[20] Kang Hoon, Multilayer associative neural networks(MANN's): storage capacity versus perfect recall, *IEEE Trans. on Neural Networks,* 1994, **5**: 812–822.

[21] Kosko B., Fuzzy associative memories. In Kandel A.(ed.), *Fuzzy Expert Systems Reading,* MA: Addison-Weley, 1987.

[22] Lee C. Y., Wang J. S. & Lee R. C. T., Characteristics of the Hopfield associative memory utilizing isomorphism relations, *IEEE Trans. on Neural Networks,* 1994, **5**: 523–526.

[23] Lee S. C. & Lee E. T., Fuzzy neural networks, *Math. Biosci.,* 1975, **23**: 151–177.

[24] Lee S. C. & Lee E. T., Fuzzy sets and neural networks, *J. Cybernetics,* 1974, **4**: 83–103.

[25] Li X. Z. & Ruan Da, Novel neural algorithms based on fuzzy δ rules for solving fuzzy relation equations: Part I, *Fuzzy Sets and Systems,* 1997, **90**: 11–23.

[26] Lin C. T.& Lee C. J. G., Neural network based fuzzy logic control and decision system, *IEEE Trans. on Computer,* 1991, **40**(12) .

[27] Liu Puyin, Fuzzy associative memory of fuzzy neural network based on a class of norm operations, *Systems Engineering and Elect.*(In Chinese), 1997,**19**(11): 54–59.

[28] Liu Puyin, Stability and fault–tolerance of the fuzzy bidirectional associative memory, *Control and Decision*(In Chinese), 1999, **14**(5): 443–447.

[29] Liu Puyin, A learning algorithm based on fault-tolerance for Min-max fuzzy Hopfield networks, *J. National Uni. Defence Tech.*(In Chinese), 1998, **20**(1): 116–123.

[30] Liu Puyin, A learning algorithm of fuzzy bidirectional associative memory based on fault-tolerance, *Systems Engineering and Elect.* (In Chinese), 1998, **20**(8): 48–52.

[31] Liu Puyin, Zhang Hanjiang & Wu Mengda, et al, Fuzzy neural network theory–a survey, *Fuzzy System and Math.*(In Chinese), 1998, **12**(1): 77–87.

[32] Liu Puyin & Zhang Hanjiang, Analysis for stability and fault-tolerance of min-max fuzzy Hopfield network with threshold, *Fuzzy Systems and Math.,* 1998, **12**(4): 47-55.

[33] Liu Puyin & Zhang Hanjiang, Stability of a class of fuzzy bidirectional associative memory, *Fuzzy Systems and Math.,* 1998, **12**(4): 40–46.

[34] Liu Puyin, Max-min fuzzy Hopfield network and an efficient learning algorithm, *Fuzzy Sets and Systems,* 2000, **112**: 41–49.

[35] Liu Puyin, The fuzzy associative memory of max-min fuzzy neural networks with threshold, *Fuzzy Sets and Systems,* 1999, **107**: 147–157.

[36] Liu Puyin, Analyses of equilibrium by fuzzy connection networks related to max-min fuzzy Hopfield networks, *Fuzzy Sets and Systems,* 2000, **119**: 303–312.

[37] Liu Puyin, Wang Hao & Zhang Hanjiang, Fuzzy connection networks related to max-min fuzzy bidirectional associative memories and their properties, *J.Fuzzy Mathematics,* 2000, **8**: 51–63.

[38] Liu Zengliang & Liu Youcai, *Fuzzy Logic and Neural Network—Theory and Application*, Beijing: Press of Beijing Uni. of Aero. and Astro., 1996.

[39] Pal S. K. & Mitra S., Multi-layer perceptron, fuzzy sets, and classification, *IEEE Trans. on Neural Networks,* 1992, **3**: 683–697.

[40] Pedrycz W., Fuzzy neural networks and neuro-computations, *Fuzzy Sets and Systems,* 1993, **56**: 1–28.

[41] Sanchez E., Eigen fuzzy sets and fuzzy relations, *J. Math. Anal. Appl.,* 1981, **81**: 399–421.

[42] Sanchez E., *Eigen fuzzy sets, Session Approximate Reasoning and Approximate Algorithms in Computer Science,* New York: NNC, 1976.

[43] Stamou G. B. & Tzafestas S. G., Neural fuzzy relational systems with a new learning algorithm, *Math. and Computers in Simulation,* 2000, **51**(3-4): 301–314.

[44] Sudkamp T. & Hammell R. J., Interpolation, completion, and learning fuzzy rules, *IEEE Trans. on Systems Man and Cybernet.,* 1994, **24**(2): 332–342.

[45] Wang T., Zhuang X. H. & Xing X. L., Designing bidirectional associative memories with optimal stability, *IEEE Trans. on System, Man and Cybernet.,* 1994, **24**: 778–790.

[46] Ushida H., Yamaguchi T. & Takagi T., Fuzzy associative memory based knowledge construction with an application to a human-machine interface, *IEEE Trans. on Industrial Elect.,* 1999, **46**(4): 851–860.

[47] Wang L. X., *Adaptive Fuzzy Systems and Control: Design Stability Analysis,* PTR Prentice-Hall, Englewood Cliffs, NJ, 1994.

[48] Wang Z. F., Jin D. M. & Li Z. J., Research on monolithic fuzzy neural networks self-learning problems, *Acta Elect. Sinica*(In Chinese), 1997, **25**: 33–38.

[49] Wei Limei & Xie Weixin, A new method for extracting the rules of the fuzzy associative memory, *Journal of Electronics,* 1999, **16**(4): 327–335.

[50] Wiggins R., Docking a truck: A genetic fuzzy approach, *AI Expert,* 1992, **7**: 28–35.

[51] Yang X. & Toh P. S., Adaptive fuzzy multilevel median filter, *IEEE Trans. on Image Processing,* 1995, **4**(5): 680–682.

[52] Zhang Chengfu & Zhao Gang, Some problems related to associative memory neural network, *Acta Automatica Sinica*(In Chinese), 1994, **20**(5): 513–521.

[53] Zhang J. Y., Xu J. & Bao Z., The connection analysis for Hopfield networks, *Acta Automatics Sinica*(In Chinese), 1997, **23**(3): 446–453.

[54] Zhang S. Q. & Karim M. A., Optical fuzzy systems based on efficient implementation of a fuzzy associative memory, *Optics Letters,* 1998, **23**(4): 292–294.

[55] Zhou E. P. & Harrison D. K., Improving error compensation via a fuzzy-neural hybrid model, *Journal of Manufacturing Systems,* 1999, **18**(5): 335–344.

CHAPTER IV

Regular Fuzzy Neural Networks

The neural systems with the following properties are called regular fuzzy neural networks (FNN's): (i) The topological architectures are identical with ones of conventional multi-layer feedforward neural networks; (ii) The input signals, connection weights and biases (or thresholds) related, are fuzzy sets in $\mathcal{F}(\mathbb{R})$; (iii) The *internal operations are based on* Zadeh's *extension principle* and fuzzy arithmetic [73, 84]. Like conventional neural networks (see [9, 10, 14–16, 67, 72]), regular FNN's have recently been successfully applied in many real fields, for example, system modeling [8, 17, 28, 71], pattern classification [37, 41, 74], system identification [25, 39, 42, 71], process control [29, 72], signal processing [66], and communication [66], and so on.

We may classify regular FNN's into three classes that distinguish themselves with the values of input signals and connection weights [6, 64, 65]. The first one includes those whose inputs are real numbers and connection weights are fuzzy sets (see [29, 50, 51, 56, 57, 59]). The second one refers to those whose inputs are fuzzy sets and connection weights are real numbers [37, 39]. The third one means those whose inputs and weights are fuzzy sets (see [21–25, 41, 53] etc.). And the third class is general, it includes the preceding two classes as special cases. The study of this chapter focuses on the third class FNN's, that is, both input signals and connection weights of FNN are fuzzy sets.

The research in this field is at its infancy and many fundamental problems, such as, the systematic analysis for network structures, the learning algorithms for fuzzy weights and fuzzy biases, the study to the performance of the algorithms related, approximating capability analysis of FNN's, and so on remain to be solved. We start the chapter with the investigation of fuzzy neurons. A complete analysis to above important subjects are presented in the rest sections, respectively. In addition to those, we shall establish the equivalent conditions for universal approximation of regular FNN's, which provides us with necessary theoretic basis for applications of FNN's. Also some realization algorithms for universal approximation are developed, and intensive simulations are carried out to demonstrate the validity of our conclusions.

§4.1 Regular fuzzy neuron and regular FNN

We can classify fuzzy neurons into three classes [1, 32, 65]. They are regular type, whose inputs and connection weights are fuzzy sets, and internal operations are based on extension principle and fuzzy arithmetic; Fuzzy operator type, whose signals related belong to $[0,1]$, and internal operations are determined by t−norm and t−conorm; And fuzzy algebraic structure type, whose neuron inputs are fuzzy sub-space [1]. In real, the fuzzy neurons in use are mostly the preceding two classes. Let us mainly aim at regular fuzzy neurons. At first, by Zadeh's extension principle we define the extension operations and fuzzy arithmetic in $\mathcal{F}_0(\mathbb{R})$.

Let $\widetilde{A}, \widetilde{B} \in \mathcal{F}_0(\mathbb{R})$, and $\lambda \in \mathbb{R}$, define extended plus '+', extended subtraction '−', extended multiplication '·' and extended scalar product '·', respectively as follows: Let $z \in \mathbb{R}$,

$$(\widetilde{A}+\widetilde{B})(z)=\bigvee_{x+y=z}\{\widetilde{A}(x)\wedge\widetilde{B}(y)\};\ (\widetilde{A}-\widetilde{B})(z)=\bigvee_{x-y=z}\{\widetilde{A}(x)\wedge\widetilde{B}(y)\};$$

$$(\widetilde{A}\cdot\widetilde{B})(z)=\bigvee_{xy=z}\{\widetilde{A}(x)\wedge\widetilde{B}(y)\};\ (\lambda\cdot\widetilde{A})(z)=\bigvee_{\lambda x=z}\{\widetilde{A}(x)\}=\begin{cases}\widetilde{A}\Big(\dfrac{z}{\lambda}\Big), & \lambda\neq 0,\\ \chi_{\{0\}}(z), & \lambda=0.\end{cases}$$

For given $\widetilde{\mathbf{X}}=(\widetilde{X}_1,...,\widetilde{X}_d)$, $\widetilde{\mathbf{Y}}=(\widetilde{Y}_1,...,\widetilde{Y}_d)\in\mathcal{F}_0(\mathbb{R})^d$, we call $\langle\widetilde{\mathbf{X}},\widetilde{\mathbf{Y}}\rangle$ the fuzzy inner product, which is written as

$$\langle\widetilde{\mathbf{X}},\ \widetilde{\mathbf{Y}}\rangle\triangleq\sum_{i=1}^{d}\widetilde{X}_i\cdot\widetilde{Y}_i\,.$$

When both $\widetilde{\mathbf{X}}$, $\widetilde{\mathbf{Y}}$ degenerate as vectors $\mathbf{x}=(x_1,...,x_d)$, $\mathbf{y}=(y_1,...,y_d)\in\mathbb{R}^d$, $\langle\widetilde{\mathbf{X}},\ \widetilde{\mathbf{Y}}\rangle$ is an inner product in $\mathbb{R}^d$. If $A\subset\mathbb{R}^d$ is bounded, by $s(\cdot,A)$ we denote the support function of A: $s(\mathbf{x},A)=\sup\{\langle\mathbf{x},\mathbf{y}\rangle|\mathbf{y}\in A\}$ $(\mathbf{x}\in\mathbb{R}^d)$.

Assume that $f:\mathcal{F}_0(\mathbb{R})^d\longrightarrow\mathcal{F}(\mathbb{R})$ is an extended function of $f:\mathbb{R}^d\longrightarrow\mathbb{R}$, by [73], if f is continuous, it follows that $f:\mathcal{F}_0(\mathbb{R})^d\longrightarrow\mathcal{F}_0(\mathbb{R})$. By cc$(\mathbb{R})$ we denote the collection of all bounded closed intervals of $\mathbb{R}$. Obviously

$$\widetilde{A}\in\mathcal{F}_0(\mathbb{R})\Longrightarrow\forall\alpha\in(0,1],\ \widetilde{A}_\alpha\in\mathrm{cc}(\mathbb{R}),\ \text{moreover}\ \mathrm{Supp}(\widetilde{A})\in\mathrm{cc}(\mathbb{R}).\qquad(4.1)$$

For $\widetilde{A},\widetilde{B}\in\mathcal{F}_0(\mathbb{R})$, we may show $\widetilde{A}\subset\widetilde{B},\Longrightarrow\mathrm{Supp}(\widetilde{A})\subset\mathrm{Supp}(\widetilde{B})$, $\mathrm{Ker}(\widetilde{A})\subset\mathrm{Ker}(\widetilde{B})$. If $\widetilde{\mathbf{A}}=(\widetilde{A}_1,...,\widetilde{A}_d)\in\mathcal{F}_0(\mathbb{R})^d$, from now on we denote $\mathrm{Supp}(\widetilde{\mathbf{A}})=(\mathrm{Supp}(A_1),...,\mathrm{Supp}(\widetilde{A}_d))$.

Lemma 4.1 *Let $\widetilde{A}$, $\widetilde{B}$, $\widetilde{A}_1,...,\widetilde{A}_d\in\mathcal{F}_0(\mathbb{R})$, and $\alpha\in(0,1]$, $\lambda\in(0,+\infty)$. Then the following facts hold:*

(i) $(\widetilde{A}+\widetilde{B})_\alpha=\widetilde{A}_\alpha+\widetilde{B}_\alpha$, $(\widetilde{A}\cdot\widetilde{B})_\alpha=\widetilde{A}_\alpha\times\widetilde{B}_\alpha$, $(\lambda\cdot\widetilde{A})_\alpha=\lambda\,\widetilde{A}_\alpha$;

(ii) If $f:\ \mathbb{R}^d\longrightarrow\mathbb{R}$ is continuous, $f(\widetilde{A}_1,...,\widetilde{A}_d)_\alpha=f((\widetilde{A}_1)_\alpha,...,(\widetilde{A}_d)_\alpha)$, *furthermore,* $f(\mathrm{Supp}(\widetilde{A}_1),...,\mathrm{Supp}(\widetilde{A}_d))=\mathrm{Supp}(f(\widetilde{A}_1,...,\widetilde{A}_d))$.

Proof. (i) and the first part of (ii) come from [73, 84], so it suffices to prove the second part of (ii).

At first, for $A_1, ..., A_d \subset \mathbb{R}$, we shall prove: $\overline{f(A_1, ..., A_d)} = f(\overline{A}_1, ..., \overline{A}_d)$. In fact, by the fact that $f(A_1, ..., A_d) \subset f(\overline{A}_1, ..., \overline{A}_d)$, and the continuity of f it follows that $f(\overline{A}, ..., \overline{A}_d)$ is a closed set, and $\overline{f(A_1, ..., A_d)} \subset f(\overline{A}_1, ..., \overline{A}_d)$. On the other hand, arbitrarily given $y \in f(\overline{A}_1, ..., \overline{A}_d)$, there exists $\mathbf{x} = (x_1, ..., x_d) : x_i \in \overline{A}_i$, satisfying $y = f(\mathbf{x})$. Therefore, $\forall i \in \{1, ..., d\}$, there is a sequence $\{x_i^k | k \in \mathbb{N}\} \subset A_i : x_i^k \to x_i \ (k \to +\infty)$. Write $\mathbf{x}_k = (x_1^k, ..., x_d^k)$, then $f(\mathbf{x}_k) \in f(A_1, ..., A_d)$, $\mathbf{x}_k \longrightarrow \mathbf{x} \ (k \to +\infty)$. By the continuity of f, it follows that $\lim_{k \to +\infty} f(\mathbf{x}_k) = f(\mathbf{x}) = y$. Hence $y \in \overline{f(A_1, ..., A_d)}$. Thus, $f(\overline{A}_1, ..., \overline{A}_d) \subset \overline{f(A_1, ..., A_d)}$. Consequently, $\overline{f(A_1, ..., A_d)} = f(\overline{A}_1, ..., \overline{A}_d)$. By the following fact:

$$\begin{cases} \text{Supp}(f(\widetilde{A}_1, ..., \widetilde{A}_d)) = \overline{\{y \in \mathbb{R} | f(\widetilde{A}_1, ..., \widetilde{A}_d)(y) > 0\}}, \\ f(\text{Supp}(\widetilde{A}_1), ..., \text{Supp}(\widetilde{A}_d)) = f\Big(\overline{\{x \in \mathbb{R} | \widetilde{A}_1(x) > 0\}}, ..., \overline{\{x \in \mathbb{R} | \widetilde{A}_d(x) > 0\}}\Big), \end{cases} \tag{4.2}$$

and by (4.2) we obtain

$$\begin{aligned} f(\text{Supp}(\widetilde{A}_1), ..., \text{Supp}(\widetilde{A}_d)) &= \overline{f\Big(\{x_1 \in \mathbb{R} | \widetilde{A}_1(x_1) > 0\}, ..., \{x_d \in \mathbb{R} | \widetilde{A}_d(x_d) > 0\}\Big)} \\ &= \overline{\{f(x_1, ..., x_d) | \widetilde{A}_1(x_1) > 0, ..., \widetilde{A}_d(x_d) > 0\}}. \end{aligned}$$

And by (1.8) it is easy to prove

$$\{y \in \mathbb{R} | f(\widetilde{A}_1, ..., \widetilde{A}_d)(y) > 0\} = \{f(x_1, ..., x_d) | \widetilde{A}_1(x_1) > 0, ..., \widetilde{A}_d(x_d) > 0\}.$$

Thus, $\text{Supp}(f(\widetilde{A}_1, ..., \widetilde{A}_d)) = f(\text{Supp}(\widetilde{A}_1), ..., \text{Supp}(\widetilde{A}_d))$. □

4.1.1 Regular fuzzy neuron

A regular fuzzy neuron is defined by fuzzifying a crisp neuron, directly. Its structure is shown as Figure 4.1. d fuzzy inputs $\widetilde{X}_1, ..., \widetilde{X}_d$, and connection weights $\widetilde{W}_1, ..., \widetilde{W}_d$ related are fuzzy numbers in $\mathcal{F}_0(\mathbb{R})$.

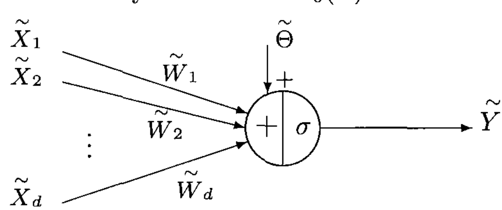


Figure 4.1 Regular fuzzy neuron

The I/O relationship of the fuzzy neuron is determined as follows:

$$\widetilde{Y} \triangleq F(\widetilde{X}_1, ..., \widetilde{X}_d) = \sigma\Big(\sum_{i=1}^{d} \widetilde{X}_i \cdot \widetilde{W}_i + \widetilde{\Theta}\Big) = \sigma(\langle \widetilde{\mathbf{X}}, \widetilde{\mathbf{W}} \rangle + \widetilde{\Theta}), \tag{4.3}$$

where $\tilde{\Theta}$ means a fuzzy bias (or a fuzzy threshold), and $\tilde{\mathbf{X}}=(\tilde{X}_1,...,\tilde{X}_d)$, $\tilde{\mathbf{W}}=(\tilde{W}_1,...,\tilde{W}_d)\in\mathcal{F}_0(\mathbb{R})^d$ is fuzzy vectors, $\sigma:\mathbb{R}\longrightarrow\mathbb{R}$ is a transfer function.

If $\sigma(x)=x$, the neuron related is called linear. In real, usually σ is selected as the following forms (where $\alpha>0$ is a constant) [32]:

$$\text{(i) Piecewise linear function}:\ \sigma(x)=\frac{|1+\alpha x|+|1-\alpha x|}{2};$$

$$\text{(ii) Hard judgment function}:\ \sigma(x)=\text{sign}(x);$$

$$\text{(iii) S type function}:\ \sigma(x)=\frac{1}{1+\exp(-\alpha x)}\ (\alpha>0);$$

$$\text{(iv) Radial basis function}:\ \sigma(x)=\exp(-\alpha|x|).$$

Related to (4.3) an obvious fact holds, that is, the I/O relationship $F(\cdot)$ is monotone, i.e. for given $\tilde{X}_1,...,\tilde{X}_d,\ \tilde{Z}_1,...,\tilde{Z}_d\in\mathcal{F}_0(\mathbb{R})$,

$$\tilde{X}_i\subset\tilde{Z}_i\ \ (i=1,...,d)\Longrightarrow F(\tilde{X}_1,...,\tilde{X}_d)\subset F(\tilde{Z}_1,...,\tilde{Z}_d). \tag{4.4}$$

Fuzzy neurons can process all kinds of fuzzy information, efficiently . It includes crisp neurons as special cases.

4.1.2 Regular fuzzy neural network

A regular FNN is an organic structure that connects many neurons in given order. Now the research will aim at a multi-layer feedforward regular FNN. As an example, taking a three layer feedforward network with multiple inputs and single output (MISO), we can establish the topological structure related as shown in Figure 4.2, where the connection weight between neuron i in the input layer and neuron j in the hidden layer is $\tilde{W}_{ij}\in\mathcal{F}_0(\mathbb{R})$, and the connection weight between j and output neuron is $\tilde{V}_j\in\mathcal{F}_0(\mathbb{R})$.

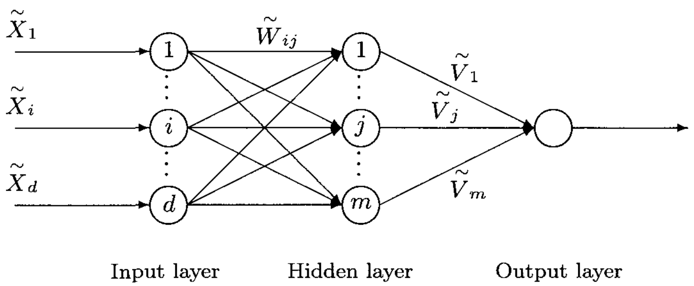


Figure 4.2 Topological structure of regular FNN

Assume that the neurons in the input and output layers are linear, and the hidden neurons have transfer function σ. Denote

$$\tilde{\mathbf{W}}(j)=(\tilde{W}_{1j},...,\tilde{W}_{dj}),\ \ \tilde{\mathbf{X}}=(\tilde{X}_1,...,\tilde{X}_d).$$

The I/O relationship of the regular FNN determined by Figure 4.2 is as follows:

$$\begin{aligned}\widetilde{Y} &\triangleq \widetilde{F}_{nn}(\widetilde{X}_1,...,\widetilde{X}_d) = \sum_{j=1}^{m} \widetilde{V}_j \cdot \sigma\Big(\sum_{i=1}^{d} \widetilde{X}_i \cdot \widetilde{W}_{ij} + \widetilde{\Theta}_j\Big) \\ &= \sum_{j=1}^{m} \widetilde{V}_j \cdot \sigma\big(\langle \widetilde{\mathbf{X}},\ \widetilde{\mathbf{W}}(j)\rangle + \widetilde{\Theta}_j\big).\end{aligned} \tag{4.5}$$

where $\widetilde{\Theta}_j \in \mathcal{F}_0(\mathbb{R})$ $(j = 1,...,m)$ is a fuzzy threshold of hidden neuron j. By (4.4) easily we have, fuzzy function $\widetilde{F}_{nn}$ is non-decreasing, that is, the following fact holds:

$$\widetilde{X}_i \subset \widetilde{Z}_i \ \ (i = 1,...,d), \Longrightarrow \widetilde{F}_{nn}(\widetilde{X}_1,...,\widetilde{X}_d) \subset \widetilde{F}_{nn}(\widetilde{Z}_1,...,\widetilde{Z}_d). \tag{4.6}$$

Two important subjects related to regular FNN's are as follows. The first is the learning algorithm for fuzzy connection weights $\widetilde{W}_{ij}$, $\widetilde{V}_j$ and fuzzy threshold $\widetilde{\Theta}_j$. The second is the analysis of the universal approximation property. The systematical research to such two problems is our main objective in this chapter. Before doing that, as preliminaries we are going to give the definition of universal approximation of regular FNN's, and present the analysis of the universal approximation of three-layer feedforward FNN's defined as (4.5) to the given fuzzy function classes.

Definition 4.1 Let $\mathcal{C}_F$ be a sub-class of collection of continuous fuzzy functions that $\mathcal{F}_0(\mathbb{R})^d \longrightarrow \mathcal{F}_0(\mathbb{R})$. Universal approximation of FNN (4.5) to $\mathcal{C}_F$ means that if $\forall F \in \mathcal{C}_F$, and arbitrarily given compact set $U \subset \mathcal{F}_0(\mathbb{R})^d$, and $\varepsilon > 0$, there exist $m \in \mathbb{N}$, and fuzzy weights $\widetilde{W}_{ij}, \widetilde{V}_j \in \mathcal{F}_0(\mathbb{R})$, and fuzzy threshold $\widetilde{\Theta}_j \in \mathcal{F}_0(\mathbb{R})$ $(i = 1,...,d, j = 1,...,m)$, such that $\forall\ \widetilde{\mathbf{X}} = (\widetilde{X}_1,...,\widetilde{X}_d) \in \mathcal{U}$, $D\big(\widetilde{F}_{nn}(\widetilde{\mathbf{X}}),\ F(\widetilde{\mathbf{X}})\big) < \varepsilon$. Also FNN (4.5) is called the universal approximator of $\mathcal{C}_F$, or we call that $\mathcal{C}_F$ guarantee the universal approximation of FNN (4.5) to hold.

Choosing $\mathcal{C}_F = \big\{$continuous fuzzy function that $\mathcal{F}_0(\mathbb{R})^d \longrightarrow \mathcal{F}_0(\mathbb{R})\big\}$, Buckley and Hayashi in [5] have proved that the universal approximation of FNN's as (4.5) to $\mathcal{C}_F$ does not hold. And they conjecture that if letting $\mathcal{C}_F = \big\{$continuously increasing fuzzy function that $\mathcal{F}_0(\mathbb{R})^d \longrightarrow \mathcal{F}_0(\mathbb{R})\big\}$, the universal approximation of FNN (4.5) holds. However, by the following discussion we may find that such a conjecture is wrong. Being increasing is only a necessary condition for fuzzy functions to guarantee the universal approximation of FNN's to hold.

4.1.3 A counter example of universal approximation

Let $\widetilde{T} \in \mathcal{F}_0(\mathbb{R})$. If the membership curve of $\widetilde{T}$ is a trapezoid, $\widetilde{T}$ is called a trapezoidal fuzzy number, as shown in Figure 4.3. Write $\widetilde{T}$ as $(t_0/t_2/t_3/t_1)$, where $\text{Supp}(\widetilde{T}) = [t_0, t_1]$, $\text{Ker}(\widetilde{T}) = [t_2, t_3]$. If $\widetilde{A} \in \mathcal{F}_0(\mathbb{R})$, and denote

$$\text{Supp}(\widetilde{A}) = [s_1(\widetilde{A}), s_2(\widetilde{A})],\ \text{Ker}(\widetilde{A}) = [e_1(\widetilde{A}),\ e_2(\widetilde{A})].$$

Let us now define the trapezoidal fuzzy function $T_r(\cdot)$ as follows [53]: $T_r(\widetilde{A}) = \big(s_1(\widetilde{A})/e_1(\widetilde{A})/e_2(\widetilde{A})/s_2(\widetilde{A})\big)$, as shown in Figure 4.4.

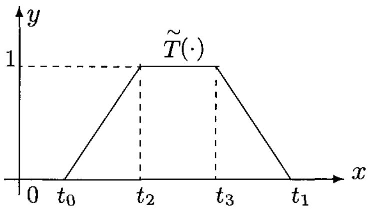


Figure 4.3 $\widetilde{T}(\cdot)$

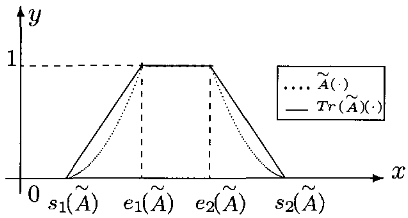


Figure 4.4 $T_r(\widetilde{A})(\cdot)$

Lemma 4.2 *Trapezoidal fuzzy function $T_r(\cdot)$ is increasing, that is, for arbitrary $\widetilde{A}_1, \widetilde{A}_2 \in \mathcal{F}_0(\mathbb{R})$, $\widetilde{A}_1 \subset \widetilde{A}_2, \Longrightarrow T_r(\widetilde{A}_1) \subset T_r(\widetilde{A}_2)$, moreover T_r is uniformly continuous on $\mathcal{F}_0(\mathbb{R})$.*

Proof. For $i = 1, 2$, assume that $\text{Supp}(\widetilde{A}_i) = [s_1(\widetilde{A}_i),\ s_2(\widetilde{A}_i)]$, $\text{Ker}(\widetilde{A}_i) = [e_1(\widetilde{A}_i),\ e_2(\widetilde{A}_i)]$. Then by fact that $\widetilde{A}_1 \subset \widetilde{A}_2$ we obtain

$$s_1(\widetilde{A}_2) \leq s_1(\widetilde{A}_1) \leq s_2(\widetilde{A}_1) \leq s_2(\widetilde{A}_2),$$
$$e_1(\widetilde{A}_2) \leq e_1(\widetilde{A}_1) \leq e_2(\widetilde{A}_1) \leq e_2(\widetilde{A}_2).$$

Thus, $\big(s_1(\widetilde{A}_1)/e_1(\widetilde{A}_1)/e_2(\widetilde{A}_1)/s_2(\widetilde{A}_1)\big) \subset \big(s_1(\widetilde{A}_2)/e_1(\widetilde{A}_2)/e_2(\widetilde{A}_2)/s_2(\widetilde{A}_2)\big)$, it follows that $T_r(\widetilde{A}_1) \subset T_r(\widetilde{A}_2)$. So $T_r(\cdot)$ is increasing.

Arbitrarily given $\varepsilon > 0$, let $\delta = \varepsilon$. For each $\widetilde{A}, \widetilde{B} \in \mathcal{F}_0(\mathbb{R})$: $D(\widetilde{A}, \widetilde{B}) < \delta/2$, it follows that $d_H(\widetilde{A}_\alpha, \widetilde{B}_\alpha) < \delta/2$ $(0 \leq \alpha \leq 1)$. Write

$$T_r(\widetilde{A}) = \big(s_1(\widetilde{A})/e_1(\widetilde{A})/e_2(\widetilde{A})/s_2(\widetilde{A})\big),$$
$$T_r(\widetilde{B}) = \big(s_1(\widetilde{B})/e_1(\widetilde{B})/e_2(\widetilde{B})/s_2(\widetilde{B})\big).$$

By (1.1) for defining d_H, and by choosing $\alpha = 1$, $\alpha = 0$, respectively, we conclude that

$$\max\big\{|s_1(\widetilde{A}) - s_1(\widetilde{B})|, |s_2(\widetilde{A}) - s_2(\widetilde{B})|, |e_1(\widetilde{A}) - e_1(\widetilde{B})|, |e_2(\widetilde{A}) - e_2(\widetilde{B})|\big\} < \frac{\delta}{2}. \quad (4.7)$$

For arbitrary $\alpha \in [0, 1]$, easily we have

$$D\Big(\big(s_1(\widetilde{A})/e_1(\widetilde{A})/e_2(\widetilde{A})/s_2(\widetilde{A})\big)_\alpha,\ \big(s_1(\widetilde{B})/e_1(\widetilde{B})/e_2(\widetilde{B})/s_2(\widetilde{B})\big)_\alpha\Big) = J_1^\alpha \vee J_2^\alpha,$$

where $J_i^\alpha = \left|(1-\alpha)(s_i(\widetilde{A}) - s_i(\widetilde{B})) + \alpha(e_i(\widetilde{A}) - e_i(\widetilde{B}))\right|$ $(i=1,2)$. By (4.7) it follows that $J_i^\alpha \le (1-\alpha)\delta/2 + \alpha\delta/2 = \delta/2$ $(i=1,2)$. Thus

$$D\Big(\big(s_1(\widetilde{A})/e_1(\widetilde{A})/e_2(\widetilde{A})/s_2(\widetilde{A})\big),\ \big(s_1(\widetilde{B})/e_1(\widetilde{B})/e_2(\widetilde{B})/s_2(\widetilde{B})\big)\Big)$$
$$= \bigvee_{\alpha\in[0,1]} \{J_1^\alpha \vee J_2^\alpha\} \le \frac{\delta}{2}.$$

That is, $D(T_r(\widetilde{A}),\ T_r(\widetilde{B})) < \delta = \varepsilon$. Therefore, T_r is uniformly continuous on $\mathcal{F}_0(\mathbb{R})$. □

Example 4.1 Let $T_r(\cdot)$ be a trapezoidal fuzzy function. Then there is a compact set $\mathcal{U} \subset \mathcal{F}_0(\mathbb{R})$, such that FNN (4.5) can not with arbitrary accuracy approximate T_r on $\mathcal{U}$.

Proof. Define fuzzy numbers $\widetilde{A}, \widetilde{B} \in \mathcal{F}_0(\mathbb{R})$ as follows, respectively:

$$\forall x\in\mathbb{R}, \widetilde{A}(x) = \begin{cases} x+\frac{1}{2}, & -\frac{1}{2} \le x < 0; \\ 1, & x=0; \\ \frac{1}{2} - x, & x < 0 \le \frac{1}{2}; \\ 0, & \text{otherwise.} \end{cases} \qquad \widetilde{B}(x) = \begin{cases} \frac{x+1}{2}, & -1 \le x < 0; \\ 1, & x=0; \\ \frac{1-x}{2}, & 0 < x \le 1; \\ 0, & \text{otherwise.} \end{cases}$$

The curves related to $\widetilde{A}(\cdot), \widetilde{B}(\cdot)$ are shown in (a) (b) of Figure 4.5, respectively. Obviously, $\forall \alpha \in [1/2, 1]$, $\widetilde{A}_\alpha = \widetilde{B}_\alpha = \{0\}$. Since T_r is a trapezoidal fuzzy function, it follows that $\forall x \in \mathbb{R}$, we have

$$T_r(\widetilde{A})(x) = \begin{cases} 2x+1, & -\frac{1}{2} \le x \le 0; \\ 1-2x, & x < 0 \le \frac{1}{2}; \\ 0, & \text{otherwise.} \end{cases} \qquad T_r(\widetilde{B})(x) = \begin{cases} x+1, & -1 \le x \le 0; \\ 1-x, & 0 < x \le 1; \\ 0, & \text{otherwise.} \end{cases}$$

Easily, we can show that

$$\forall \in [0,1],\ T_r(\widetilde{A})_\alpha = \Big[\frac{\alpha-1}{2}, \frac{1-\alpha}{2}\Big],\ T_r(\widetilde{B})_\alpha = \big[\alpha-1, 1-\alpha\big] \qquad (4.8)$$

Let us now prove when $d=1$, FNN (4.5) can not with arbitrary accuracy approximate T_r on compact set $\mathcal{U} = \{\widetilde{A},\ \widetilde{B}\} \subset \mathcal{F}_0(\mathbb{R})$. If the fact does not hold, let $\varepsilon_0 = 0.1$, then there exist $m \in \mathbb{N}$, and $\widetilde{W}_{1j}, \widetilde{V}_j, \widetilde{\Theta}_j \in \mathcal{F}_0(\mathbb{R})$ $(j=1,...,m)$, such that

$$D(\widetilde{F}_{nn}(\widetilde{A}),\ T_r(\widetilde{A})) < \frac{\varepsilon_0}{2},\quad d(\widetilde{F}_{nn}(\widetilde{B}),\ T_r(\widetilde{B})) < \frac{\varepsilon_0}{2}.$$

Choose $\alpha_0 = 0.75$. It follows that

$$d_H\big(\widetilde{F}_{nn}(\widetilde{A})_{\alpha_0},\ T_r(\widetilde{A})_{\alpha_0}\big) < \frac{\varepsilon_0}{2},\quad d_H\big(\widetilde{F}_{nn}(\widetilde{B})_{\alpha_0},\ T_r(\widetilde{B})_{\alpha_0}\big) < \frac{\varepsilon_0}{2}. \tag{4.9}$$

Since $\widetilde{A}_{\alpha_0}=\widetilde{B}_{\alpha_0}=\{0\}$, by Lemma 4.1, $\widetilde{F}_{nn}(\widetilde{A})_{\alpha_0} = \sum_{j=1}^{m}\big(\widetilde{W}_{ij}\big)_{\alpha_0}\cdot\sigma\big((\widetilde{\Theta}_j)_{\alpha_0}\big) = \widetilde{F}_{nn}(\widetilde{B})_{\alpha_0}$. So by (4.9) we obtain

$$\begin{aligned} d_H\big(T_r(\widetilde{A})_{\alpha_0},\ T_r(\widetilde{B})_{\alpha_0}\big) &\le d_H\big(T_r(\widetilde{A})_{\alpha_0},\ \widetilde{F}_{nn}(\widetilde{A})_{\alpha_0}\big) + d_H\big(\widetilde{F}_{nn}(\widetilde{A})_{\alpha_0},\ T_r(\widetilde{B})_{\alpha_0}\big) \\ &= d_H\big(T_r(\widetilde{A})_{\alpha_0},\ \widetilde{F}_{nn}(\widetilde{A})_{\alpha_0}\big) + d_H\big(\widetilde{F}_{nn}(\widetilde{B})_{\alpha_0},\ T_r(\widetilde{B})_{\alpha_0}\big) \\ &< \frac{\varepsilon_0}{2} + \frac{\varepsilon_0}{2} = \varepsilon_0. \end{aligned} \tag{4.10}$$

On the other hand, by (4.8) it follows that

$$\begin{aligned} d_H\big(T_r(\widetilde{A})_{\alpha_0},\ T_r(\widetilde{B})_{\alpha_0}\big) &= d_H\Big(\Big[\frac{\alpha_0 - 1}{2},\ \frac{1-\alpha_0}{2}\Big],\ \big[\alpha_0 - 1, 1 - \alpha_0\big]\Big) \\ &= \max\Big\{\Big|\frac{\alpha_0 - 1}{2}\Big|,\ \Big|\frac{1-\alpha_0}{2}\Big|\Big\} = \frac{1-\alpha_0}{2} = \frac{1}{8} > \varepsilon_0, \end{aligned}$$

which contradicts (4.10). So FNN (4.5) can not approximation T_r on $\mathcal{U}$ with arbitrary accuracy. □

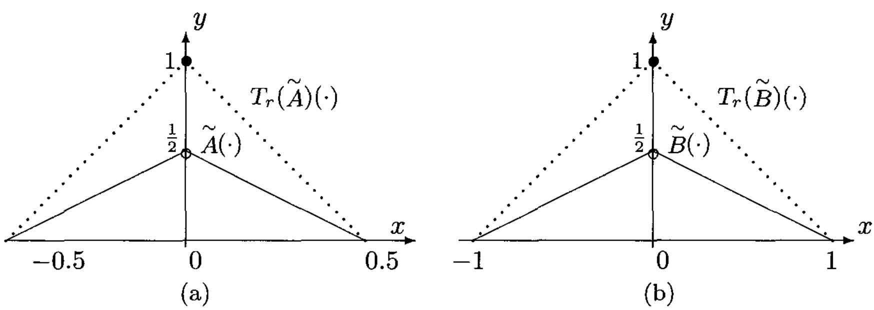


Figure 4.5 Membership curves: (a) Curves of $\widetilde{A}$, $T_r(\widetilde{A})$; (b) Curves of $\widetilde{B}$, $T_r(\widetilde{B})$

4.1.4 An example of universal approximation

By Example 4.1 and Lemma 4.2, if choose $\mathcal{C}_F$ as the collection of all continuously increasing fuzzy functions, the universal approximation of FNN (4.5) to $\mathcal{C}_F$ does not hold. An important problem is, whether can a sub-class $\mathcal{C}_F$ of the collection of continuous fuzzy functions be established to guarantee the universal approximation of FNN (4.5) to hold? The answer is yes, since we can choose $\mathcal{C}_F = \big\{$extended function of continuous real function$\big\}$. In the following we study such a problem in $\mathbb{R}^d$.

Lemma 4.3 *Let f, g be continuous functions on compact set $B \subset \mathbb{R}^d$, and let $h > 0$. Moreover, $\forall \mathbf{x} \in B$, $|f(\mathbf{x}) - g(\mathbf{x})| < h$. Then for each compact set $B_1 \subset B$, it follows that* $\Big| \bigvee_{\mathbf{x} \in B_1} \{f(\mathbf{x})\} - \bigvee_{\mathbf{x} \in B_1} \{g(\mathbf{x})\} \Big| < h$.

Proof. Since B_1 is a compact set, and f, g are continuous on B_1, there are $\mathbf{x}_0 \in B_1$, $\mathbf{y}_0 \in B_1$, satisfying $f(\mathbf{x}_0) = \bigvee_{\mathbf{x} \in B_1} \{f(\mathbf{x})\}, g(\mathbf{y}_0) = \bigvee_{\mathbf{x} \in B_1} \{g(\mathbf{x})\}$. If $|f(\mathbf{x}_0) - g(\mathbf{y}_0)| \geq h$, we have

$$f(\mathbf{x}_0) - g(\mathbf{y}_0) \leq -h, \quad \text{or} \quad f(\mathbf{x}_0) - g(\mathbf{y}_0) \geq h. \tag{4.11}$$

To the first case of (4.11), considering $f(\mathbf{y}_0) \leq f(\mathbf{x}_0)$, we obtain

$$f(\mathbf{y}_0) - g(\mathbf{y}_0) \leq f(\mathbf{x}_0) - g(\mathbf{y}_0) \leq -h \Longrightarrow |f(\mathbf{y}_0) - g(\mathbf{y}_0)| \geq h,$$

which contradicts the assumption! To the second case of (4.11), since $g(\mathbf{x}_0) \leq g(\mathbf{y}_0)$, it follows that

$$f(\mathbf{x}_0) - g(\mathbf{x}_0) \geq f(\mathbf{x}_0) - g(\mathbf{y}_0) \geq h, \Longrightarrow |f(\mathbf{x}_0) - g(\mathbf{x}_0)| \geq h,$$

which also contradicts assumption. Thus, (4.11) does not hold, that is, $-h < f(\mathbf{x}_0) - g(\mathbf{y}_0) < h$, So $|f(\mathbf{x}_0) - g(\mathbf{x}_0)| < h$, i.e. $| \bigvee_{\mathbf{x} \in B_1} f(\mathbf{x}) - \bigvee_{\mathbf{x} \in B_1} g(\mathbf{x})| < h$. The lemma is proved. □

Diamond P. & Kloedem P. have showed in [19, 20] that each compact set in $\mathcal{F}_0(\mathbb{R})$ is uniformly bounded, that is, the following conclusion holds.

Proposition 4.1 *Let $\mathcal{U} \subset \mathcal{F}_0(\mathbb{R})^d$ be a compact set. Then the supports of fuzzy sets in $\mathcal{U}$ is uniformly bounded, that is, there is a compact set $U \subset \mathbb{R}^d$, such that $\forall\ \widetilde{\mathbf{A}} \in \mathcal{U}, \mathrm{Supp}(\widetilde{\mathbf{A}}) \subset U$.*

Proof. Fuzzy number $\widetilde{\mathbf{O}} = \overbrace{\chi_{\{0\}} \times \cdots \times \chi_{\{0\}}}^{d} \in \mathcal{F}_0(\mathbb{R})^d$, where $\chi_{\{0\}}(0) = 1$, $\chi_{\{0\}}(x) = 0\ (x \neq 0)$. Since $\mathcal{U}$ is a compact set in metric space $\big(\mathcal{F}_0(\mathbb{R})^d,\ D\big)$, the set $\{D(\widetilde{\mathbf{O}},\ \widetilde{\mathbf{A}})|\widetilde{\mathbf{A}} \in \mathcal{U}\} \subset \mathbb{R}$ is a bounded set, that is, there is $K > 0$, we have, $\widetilde{\mathbf{A}}_0 = \big((\widetilde{A}_1)_0, ..., (\widetilde{A}_d)_0\big) = \big(\mathrm{Supp}(\widetilde{A}_1), ..., \mathrm{Supp}(\widetilde{A}_d)\big) = \mathrm{Supp}(\widetilde{\mathbf{A}})$, hence $\forall\ \widetilde{\mathbf{A}} \in \mathcal{U},\ D(\widetilde{\mathbf{O}},\ \widetilde{\mathbf{A}}) \leq K$, and therefore $\sup\{\|\mathbf{x}\|\ |\mathbf{x} = (x_1, ..., x_d) \in \widetilde{\mathbf{A}}_0\} \leq K$. Thus, if let $U = [-K,\ K]^d$, then $U \subset \mathbb{R}^d$ is a compact set. Moreover, $\forall\ \widetilde{\mathbf{A}} \in \mathcal{U}$, $\forall \mathbf{x} \in \mathrm{Supp}(\widetilde{\mathbf{A}})$, $\|\mathbf{x}\| \leq K$, i.e. $\mathbf{x} \in U$. Therefore, $\mathrm{Supp}(\widetilde{\mathbf{A}}) \subset U$. □

Lemma 4.4 *Let $f : \mathbb{R}^d \longrightarrow \mathbb{R}$ be a continuous function. Then extended function $f : \mathcal{F}_0(\mathbb{R})^d \longrightarrow \mathcal{F}_0(\mathbb{R})$ is continuous in $\mathcal{F}_0(\mathbb{R})^d$.*

Proof. By Lemma 4.1 we obtain the fact that for arbitrary $(\widetilde{A}_1, ..., \widetilde{A}_d) \in \mathcal{F}_0(\mathbb{R})^d$, $\mathrm{Supp}\big(f(\widetilde{A}_1, ..., \widetilde{A}_d)\big) = f\big(\mathrm{Supp}(\widetilde{A}_1), ..., \mathrm{Supp}(\widetilde{A}_d)\big)$. Since f is continuous, $f\big(\mathrm{Supp}(\widetilde{A}_1), ..., \mathrm{Supp}(\widetilde{A}_d)\big) \subset \mathbb{R}$ is a compact, i.e. $\mathrm{Supp}\big(f(\widetilde{A}_1, ..., \widetilde{A}_d)\big) \subset \mathbb{R}$

is compact. Also by Lemma 4.1 we obtain, $f(\widetilde{A}_1, ..., \widetilde{A}_d)_\alpha = f\big((\widetilde{A}_1)_\alpha, ..., (\widetilde{A}_d)_\alpha\big)$ for each $\alpha \in (0,1]$. The fact that $(\widetilde{A}_i)_\alpha \in \mathrm{cc}(\mathbb{R})$ implies $f\big((\widetilde{A}_1)_\alpha, ..., (\widetilde{A}_d)_\alpha\big) \in \mathrm{cc}(\mathbb{R})$. Obviously, $\mathrm{Ker}\big(f(\widetilde{A}_1, ..., \widetilde{A}_d)\big) \neq \emptyset$. So $f(\widetilde{A}_1, ..., \widetilde{A}_d) \in \mathcal{F}_0(\mathbb{R})$, i.e. the range of extended function f is included in $\mathcal{F}_0(\mathbb{R})$.

Arbitrarily given $(\widetilde{A}_1, ..., \widetilde{A}_d) \in \mathcal{F}_0(\mathbb{R})^d$, and $\varepsilon > 0$. There is a closed interval $[a, b]$: $\forall \alpha \in [0,1]$, $(\widetilde{A}_1)_\alpha \times \cdots \times (\widetilde{A}_d)_\alpha \subset [a,b]^d$. Obviously, f is uniformly continuous on $[a, b]^d$, so there is $\delta > 0$, such that $\forall\,(x_1, ..., x_d) \in [a, b]^d$, $\forall\,(y_1, ..., y_d) \in [a, b]^d$, it follows that

$$\forall i \in \{1, ..., d\},\ |x_i - y_i| < \delta, \Longrightarrow \big|f(x_1, ..., x_d) - f(y_1, ..., y_d)\big| < \frac{\varepsilon}{2}.$$

For each $(\widetilde{X}_1, ..., \widetilde{X}_d) \in \mathcal{F}_0(\mathbb{R})^d$, if $D(\widetilde{X}_i, \widetilde{A}_i) < \delta$ $(i = 1, ..., d)$, then $\forall \alpha \in [0,1], d_H\big((\widetilde{X}_i)_\alpha,\ (\widetilde{A}_i)_\alpha\big) < \delta$, that is

$$\Big\{\bigvee_{t \in (\widetilde{A}_i)_\alpha}\Big(\bigwedge_{s \in (\widetilde{X}_i)_\alpha}\{|t-s|\}\Big)\Big\} \vee \Big\{\bigvee_{s \in (\widetilde{X}_i)_\alpha}\Big(\bigwedge_{t \in (\widetilde{A}_i)_\alpha}\{|t-s|\}\Big)\Big\} < \delta\ (i = 1, ..., d).$$

Thus, for each $i \in \{1, ..., d\}$, if $t \in (\widetilde{A}_i)_\alpha$, there is $s' \in (\widetilde{X}_i)_\alpha$, $|t - s'| < \delta$. On the other hand, $\forall s \in (\widetilde{X}_i)_\alpha$, there is $t' \in (\widetilde{A}_i)_\alpha$, satisfying $|t' - s| < \delta$. So for any $(x_1, ..., x_d) \in (\widetilde{A}_1)_\alpha \times \cdots \times (\widetilde{A}_d)_\alpha$, there is $(y_1', ..., y_d') \in (\widetilde{X}_1)_\alpha \times \cdots \times (\widetilde{X}_d)_\alpha$, $|x_i - y_i'| < \delta$ $(i = 1, ..., d)$, satisfying $|f(x_1, ..., x_d) - f(y_1', ..., y_d')| < \varepsilon/2$. And $\forall (y_1, ..., y_d) \in (\widetilde{X}_1)_\alpha \times \cdots \times (\widetilde{X}_d)_\alpha$, there is $(x_1', ..., x_d') \in (\widetilde{A}_1)_\alpha \times \cdots \times (\widetilde{A}_d)_\alpha$: $|x_i' - y_i| < \delta$ $(i = 1, ..., d)$, such that $|f(x_1', ..., x_d') - f(y_1, ..., y_d)| < \varepsilon/2$. That is, we can conclude that

$$\bigvee_{x_1 \in (\widetilde{A}_1)_\alpha, ..., x_d \in (\widetilde{A}_d)_\alpha}\Big\{\bigwedge_{y_1 \in (\widetilde{X}_1)_\alpha, ..., y_d \in (\widetilde{X}_d)_\alpha}\{|f(x_1, ..., x_d) - f(y_1, ..., y_d)|\}\Big\} \leq \frac{\varepsilon}{2};$$

$$\bigvee_{y_1 \in (\widetilde{X}_1)_\alpha, ..., y_d \in (\widetilde{X}_d)_\alpha}\Big\{\bigwedge_{x_1 \in (\widetilde{A}_1)_\alpha, ..., x_d \in (\widetilde{A}_d)_\alpha}\{|f(x_1, ..., x_d) - f(y_1, ..., y_d)|\}\Big\} \leq \frac{\varepsilon}{2}.$$

Thus, we can conclude that the following fact holds:

$$\bigvee_{x_1 \in (\widetilde{A}_1)_\alpha, ..., x_d \in (\widetilde{A}_d)_\alpha}\Big\{\bigwedge_{y_1 \in (\widetilde{X}_1)_\alpha, ..., y_d \in (\widetilde{X}_d)_\alpha}\big\{|f(x_1, ..., x_d) - f(y_1, ..., y_d)|\big\}\Big\} \vee$$
$$\vee \Big\{\bigvee_{y_1 \in (\widetilde{X}_1)_\alpha, ..., y_d \in (\widetilde{X}_d)_\alpha}\ \bigwedge_{x_1 \in (\widetilde{A}_1)_\alpha, ..., x_d \in (\widetilde{A}_d)_\alpha}\big\{|f(x_1, ..., x_d) - f(y_1, ..., y_d)|\big\}\Big\}$$
$$\leq \frac{\varepsilon}{2}.$$

Consequently, we obtain

$$d_H\big(f(\widetilde{X}_1,...,\widetilde{X}_d)_\alpha,\ f(\widetilde{A}_1,...,\widetilde{A}_d)_\alpha\big)$$
$$= d_H\big(f((\widetilde{X}_1)_\alpha,...,(\widetilde{X}_d)_\alpha),\ f((\widetilde{A}_1)_\alpha,...,(\widetilde{A}_d)_\alpha)\big) \le \frac{\varepsilon}{2}.$$
$$D(f(\widetilde{X}_1,...,\widetilde{X}_d),\ f(\widetilde{A}_1,...,\widetilde{A}_d))$$
$$= \bigvee_{\alpha\in[0,1]} \{d_H(f(\widetilde{X}_1,...,\widetilde{X}_d)_\alpha,\ f(\widetilde{A}_1,...,\widetilde{A}_d)_\alpha)\} \le \frac{\varepsilon}{2} < \varepsilon.$$

So f is continuous at $(\widetilde{A}_1,...,\widetilde{A}_d)$. Consequently f is continuous in $\mathcal{F}_0(\mathbb{R})^d$. □

Let us now generalize the conclusion (ii) of Lemma 3.1 to general case.

Lemma 4.5 *Let I be an arbitrary index set, $\{a_i|i\in I\},\ \{b_i|i\in I\}\subset[0,1]$, and $h>0$, satisfying $\forall i\in I,\ |a_i-b_i|\le h$. Then*

$$\Big|\bigvee_{i\in I} a_i - \bigvee_{i\in I} b_i\Big| \le h;\quad \Big|\bigwedge_{i\in I} a_i - \bigwedge_{i\in I} b_i\Big| \le h.$$

Proof. By assumption easily it follows that both $\beta \triangleq \bigvee_{i\in I} a_i$ and $\lambda \triangleq \bigvee_{i\in I} b_i$ exist, moreover, $\beta,\lambda\in[0,1]$. Let us now prove $|\beta-\lambda|\le h$.

For each $\varepsilon>0$, there are $i_0,\ j_0\in I$, so that $a_{i_0}\le\beta<a_{i_0}+\varepsilon,\ b_{j_0}\le\lambda<b_{j_0}+\varepsilon$. Thus

$$a_{i_0}-b_{j_0}-\varepsilon<\beta-\lambda<a_{i_0}-b_{j_0}+\varepsilon. \tag{4.12}$$

If $i_0=j_0$, by assumption and (4.12) we obtain $-h-\varepsilon<a_{i_0}-b_{j_0}-\varepsilon<\beta-\lambda<a_{i_0}-b_{j_0}+\varepsilon<h+\varepsilon$. So $|\beta-\lambda|<h+\varepsilon$. Therefore, $|\beta-\lambda|\le h$. If $i_0\ne j_0$, it suffices to prove $|\beta-\lambda|\le h$, respectively with respect to the following cases:

$$\text{I. } a_{i_0}\ge a_{j_0},\ b_{i_0}\ge b_{j_0};\quad \text{II. } a_{i_0}\ge a_{j_0},\ b_{i_0}<b_{j_0};$$
$$\text{III. } a_{i_0}<a_{j_0},\ b_{i_0}\ge b_{j_0};\quad \text{IV. } a_{i_0}<a_{j_0},\ b_{i_0}<b_{j_0}.$$

To the case I, easily we have

$$-h<a_{j_0}-b_{j_0}\le a_{i_0}-b_{j_0}<a_{i_0}-(\lambda-\varepsilon)\le a_{i_0}-b_{i_0}+\varepsilon<h+\varepsilon.$$

So by (4.12) it follows that $-h-\varepsilon<\beta-\lambda<h+2\varepsilon$, i.e. $|\beta-\lambda|<h+2\varepsilon,\Longrightarrow |\beta-\lambda|\le h$. To the case II, the following fact holds:

$$-h<a_{j_0}-b_{j_0}\le a_{i_0}-b_{j_0}<a_{i_0}-b_{i_0}<-h.$$

By (4.12) we can conclude that $-h-\varepsilon<\beta-\lambda<h+\varepsilon\Longrightarrow|\beta-\lambda|\le h$.

With the similar reason we can show $|\beta-\lambda|\le h$ to case III and case IV, respectively. Hence the first part of lemma is proved.

Easily we can imply that if $\forall i \in I$, let $a'_i = 1 - a_i, b'_i = 1 - b_i$, then $a'_i, b'_i \in [0,1]$, and $|a'_i - b'_i| \leq h$ $(i \in I)$. So by (i) it follows that $\Big|\bigvee_{i\in I} a'_i - \bigvee_{i\in I} b'_i\Big| \leq h$. Therefore

$$\Big|\bigwedge_{i\in I} a_i - \bigwedge_{i\in I} b_i\Big| = \Big|(1 - \bigwedge_{i\in I} a_i) - (1 - \bigwedge_{i\in I} b_i)\Big| = \Big|\bigvee_{i\in I} a'_i - \bigvee_{i\in I} b'_i\Big| \leq h.$$

The lemma is proved. □

Lemma 4.6 *Assume that* $f,\ g: \mathbb{R}^d \longrightarrow \mathbb{R}$ *are continuous functions,* $h > 0$. *Moreover, for each compact set of* $\mathbb{R}^d$, *it follows that* $\forall \mathbf{x} \in B,\ |f(\mathbf{x}) - g(\mathbf{x})| < h$. *Then for arbitrary compact set* $\mathcal{U}$ *of* $\mathcal{F}_0(\mathbb{R})^d$, *and each* $(\widetilde{A}_1, ..., \widetilde{A}_d) \in \mathcal{U}$, *we have,* $D\big(f(\widetilde{A}_1, ..., \widetilde{A}_d),\ g(\widetilde{A}_1, ..., \widetilde{A}_d)\big) \leq h$.

Proof. Arbitrarily given $\mathcal{U} \subset \mathcal{F}_0(\mathbb{R})^d$, by Proposition 4.1 it follows that there is a compact set B of $\mathbb{R}^d$, so that for each $(\widetilde{A}_1, ..., \widetilde{A}_d) \in \mathcal{U}$, the set defined as $\{\mathbf{x} = (x_1, ..., x_d) | x_1 \in \text{Supp}(\widetilde{A}_1), ..., x_d \in \text{Supp}(\widetilde{A}_d)\}$ is included in B. Sincef, g are continuous, by Lemma 4.1 we obtain $\forall \alpha \in [0,1]$, and for each $(\widetilde{A}_1, ..., \widetilde{A}_d) \in \mathcal{U}$, $f(\widetilde{A}_1, ..., \widetilde{A}_d)_\alpha = f\big((\widetilde{A}_1)_\alpha, ..., (\widetilde{A}_d)_\alpha\big)$, and $g(\widetilde{A}_1, ..., \widetilde{A}_d)_\alpha = g\big((\widetilde{A}_1)_\alpha, ..., (\widetilde{A}_d)_\alpha\big)$. Therefore, the following fact holds [19, 20]:

$$\begin{aligned} & d_H\big(f(\widetilde{A}_1, ..., \widetilde{A}_d)_\alpha,\ g(\widetilde{A}_1, ..., \widetilde{A}_d)_\alpha\big) \\ & = d_H\big(f\big((\widetilde{A}_1)_\alpha, ..., (\widetilde{A}_d)_\alpha\big),\ g\big((\widetilde{A}_1)_\alpha, ..., (\widetilde{A}_d)_\alpha\big) \\ & = \bigvee_{|p|=1} \big\{\big|s\big(p, f\big((\widetilde{A}_1)_\alpha\big), ..., (\widetilde{A}_d)_\alpha\big) - s\big(p, g\big((\widetilde{A}_1)_\alpha, ..., (\widetilde{A}_d)_\alpha\big)\big|\big\}. \end{aligned} \tag{4.13}$$

Moreover, for each $p \in \mathbb{R} : |p| = 1$, we have

$$\begin{aligned} & \big|s\big(p, f\big((\widetilde{A}_1)_\alpha, ..., (\widetilde{A}_d)_\alpha\big)\big) - s\big(p, g\big((\widetilde{A}_1)_\alpha, ..., (\widetilde{A}_d)_\alpha\big)\big)\big| \\ = & \big|\sup\big\{p \cdot y | y \in f\big((\widetilde{A}_1)_\alpha, ..., (\widetilde{A}_d)_\alpha\big)\big\} - \sup\big\{p \cdot y | y \in g\big((\widetilde{A}_1)_\alpha, ..., (\widetilde{A}_d)_\alpha\big)\big\}\big| \\ = & \big|\sup\big\{p \cdot f(x_1, ..., x_d) | x_i \in (\widetilde{A}_i)_\alpha\big\} - \sup\big\{p \cdot g(x_1, ..., x_d) | x_i \in (\widetilde{A}_i)_\alpha\big\}\big|, \end{aligned} \tag{4.14}$$

Considering assumption and the fact that $\big\{(x_1, ..., x_d) | x_i \in (\widetilde{A}_i)_\alpha, i = 1, ..., d\big\} \subset B, |p| = 1$, we imply

$$\forall \mathbf{x} = (x_1, ..., x_d) \in (\widetilde{A}_1)_\alpha \times \cdots \times (\widetilde{A}_d)_\alpha, \big|p \cdot f(\mathbf{x}) - p \cdot g(\mathbf{x})\big| = |f(\mathbf{x}) - g(\mathbf{x})| < h.$$

So it follows that $d_H\big(f\big((\widetilde{A}_1)_\alpha, ..., (\widetilde{A}_d)_\alpha\big),\ g\big((\widetilde{A}_1)\big)_\alpha, ..., (\widetilde{A}_d)_\alpha\big)\big) < h$ by Lemma 4.5 and (4.13) (4.14). Therefore

$$\begin{aligned} & D\big(f(\widetilde{A}_1,\ ..., \widetilde{A}_d),\ g(\widetilde{A}_1, ..., \widetilde{A}_d)\big) \\ & \qquad = \bigvee_{\alpha\in[0,1]} \big\{d_H\big(f\big((\widetilde{A}_1)_\alpha, ..., (\widetilde{A}_d)_\alpha\big),\ g\big((\widetilde{A}_1)_\alpha, ..., (\widetilde{A}_d)_\alpha\big)\big)\big\} \leq h. \end{aligned}$$

The lemma is proved. □

Theorem 4.1 *Let $f : \mathbb{R}^d \longrightarrow \mathbb{R}$ be a continuous function, and let $\mathcal{U} \subset \mathcal{F}_0(\mathbb{R})^d$ be an arbitrary compact set. Then for each $\varepsilon > 0$, there are $m \in \mathbb{N}$, $v_j, \theta_j \in \mathbb{R}$, and $\mathbf{w}_j = (w_{1j}, ..., w_{dj}) \in \mathbb{R}^d$ $(j = 1, ..., m)$, such that*

$$\forall\, \tilde{\mathbf{X}} = (\tilde{X}_1, ..., \tilde{X}_d) \in \mathcal{U},\ D\Big(f(\tilde{\mathbf{X}}),\ \sum_{j=1}^{m} v_j \cdot \sigma\big(\langle \mathbf{w}_j, \tilde{\mathbf{X}} \rangle + \theta_j\big)\Big) < \varepsilon,$$

where σ is Tauber-Wiener function.

Proof. By Proposition 4.1 it follows that there is compact set $U \subset \mathbb{R}^d$, so that $\forall (\tilde{X}_1, ..., \tilde{X}_d) \in \mathcal{U}$, $\{\mathbf{x} = (x_1, ..., x_d) | x_i \in \text{Supp}(\tilde{X}_i), i = 1, ..., d\} \subset U$. For $\varepsilon > 0$, since σ is Tauber-Wiener function, there are $m \in \mathbb{N}$, $v_j, \theta_j \in \mathbb{R}$, and $\mathbf{w}_j = (w_{1j}, ..., w_{dj}) \in \mathbb{R}^d (j = 1, ..., m)$, satisfying

$$\forall \mathbf{x} \in U, \Big| f(\mathbf{x}) - \sum_{j=1}^{m} v_j \cdot \sigma\big(\langle \mathbf{x}, \mathbf{w}_j \rangle + \theta_j\big) \Big| < \frac{\varepsilon}{2}.$$

Let $g(\mathbf{x}) = \sum_{i=j}^{m} v_j \cdot \sigma\big(\langle \mathbf{x}, \mathbf{w}_j \rangle + \theta_j\big)$ $(\mathbf{x} \in \mathbb{R}^d)$. By Lemma 4.6 we obtain

$$\forall\, \tilde{\mathbf{X}} = (\tilde{X}_1, ..., \tilde{X}_d) \in \mathcal{U},\ D\big(f(\tilde{X}_1, ..., \tilde{X}_d),\ g(\tilde{X}_1, ..., \tilde{X}_d)\big) \le \frac{\varepsilon}{2}.$$

Therefore, $D\Big(f(\tilde{\mathbf{X}}),\ \sum_{j=1}^{m} v_j \cdot \sigma\big(\langle \mathbf{w}_j, \tilde{\mathbf{X}} \rangle + \theta_j\big)\Big) \le \varepsilon/2$, which implies the theorem.
□

By Theorem 4.1, if we choose $\mathcal{C}_F$ as the collection of all extended functions, the regular FNN's as (4.5) are the universal approximators of $\mathcal{C}_F$. Related to this fact an important problem is, whether can we establish some equivalent conditions for continuous fuzzy function class $\mathcal{C}_F$ to guarantee the universal approximation of multilayer feedforward regular FNN's to hold? Such a research topic is important and urgent to apply regular FNN's as useful tools of solving many real questions, such as, system modeling and system identification and so on. The systematic research to this topic is the main objectives of §4.4 and §4.5, respectively. To this end, let us now develop some learning algorithms for FNN (4.5).

§4.2 Learning algorithms

Since the input signals, connection weights and the thresholds related to regular FNN's are fuzzy numbers, naturally it is much more difficult to develop learning algorithms for regular FNN's than for corresponding conventional neural networks, which leads to lacking systematic achievements in the field. The basic processes to deal with the learning for FNN's are similar ones for conventional neural networks, that is, define a suitable error function, and develop some iteration schemes for fuzzy weights and fuzzy bias terms. Since we do not have the calculus for fuzzy numbers, the conventional learning algorithms for multi-layer neural networks cannot be directly fuzzified.

So far there have been two approach to complete the learning of the regular FNN's. One is to apply the level sets (α−cuts) of fuzzy numbers related, and the BP algorithm is employed to adjust the endpoints of the level sets to determine the fuzzy weights and biases. For instance, Buckley et al. [7], Hayashi et al. [36] and Ishibuchi et al. [37] apply direct fuzzification to develop the fuzzy delta rule. However, it cannot be used, practically because of the lack of theoretic support. By restricting fuzzy weights and fuzzy biases to be real numbers, Ishibuchi et al. [39] propose a fuzzy BP algorithm based on finite level sets of the fuzzy numbers related. A general fuzzy number cannot be determined by finite parameter collection. This is a main reason that causes the difficulty for developing FNN learning algorithms. To avoid such a case, specifically, many authors restrict fuzzy weights and biases to be a given *fuzzy set class, such as triangular and trapezoidal fuzzy numbers*. Ishibuchi et al. [38, 40, 41] use symmetric triangular fuzzy numbers for fuzzy weights. And in [42], Ishibuchi et al. examine the ability of regular FNN's with four types of fuzzy weights (i.e. real numbers, symmetric triangular fuzzy numbers, asymmetric triangular fuzzy numbers, and asymmetric trapezoidal fuzzy numbers) to realize approximately fuzzy IF–THEN rules. In order to call off constraint conditions for fuzzy weights, Dunyak et al. [21–23] present a transformation which does not simplify the representation of fuzzy weights. And based on the level sets of fuzzy numbers and the interval arithmetic, Park et al. [75] develop an inversion algorithm of regular FNN's.

However, no matter how different fuzzy weights and error functions these learning algorithms have, two important operations '$\vee$' and '$\wedge$' are often involved. An indispensable step to construct the fuzzy BP algorithm is to differentiate '$\vee - \wedge$' operations by using the unit step function, that is, for the given real constant a,

$$\frac{\partial(x \vee a)}{\partial x} = \begin{cases} 1, & x \geq a, \\ 0, & x < a; \end{cases} \qquad \frac{\partial(x \wedge a)}{\partial x} = \begin{cases} 1, & x \leq a, \\ 0, & x > a. \end{cases} \tag{4.15}$$

Above representations are only valid for special case $x \neq a$. And if $x = a$, they are no longer valid. Based on these two derivative formulas, the chain rules for differentiation of composition functions are only in form, and lack rigorous mathematical sense. Apply [81, 82] to fully analyze the '$\vee - \wedge$' operations and to develop a rigorous theory for the calculus of '$\vee$' and '$\wedge$' operations are two subsidiary results of this section.

Another approach for the learning of regular FNN's is to utilize genetic algorithm (GA) (see Goldberg [31], Mitchell [70]) to minimize the error function and consequently determine the fuzzy connection weights and bias terms.

When the learning for weights and biases is completed with GA, the fuzzy numbers related must be restricted to a small class, such as triangular or trapezoidal fuzzy numbers, so that they can be determined by a few of parameter related to the class of fuzzy numbers. For instance, Aliev et al. [3] and Krishnamraju er al. [48] employ simple GA to train the triangular fuzzy number weights and biases of regular FNN's. They encode all fuzzy weights as a binary string (chromosome) to complete the search process. And the transfer function σ related is assumed to be an increasing real function.

Unlike neuro-fuzzy networks of mapping a non-fuzzy input signal to a non-fuzzy output, which are the main objectives of Chapter VI (see also [12, 46, 49]), regular FNN's can directly process fuzzy information. If a real system maps fuzzy inputs to fuzzy outputs, we can employ regular FNN's not neuro-fuzzy networks to realize this system, approximately (Ishibuchi et al. [44]). Furthermore, applying regular FNN's we can solve the overfitting problem (Feuring et al. [25]). Hence regular FNN's play an important role, which can not be replaced by neuro-fuzzy networks in application.

In this section, we at first employ level sets of fuzzy numbers to develop learning algorithms of FNN's. The first step to do this is to represent the output fuzzy numbers of FNN as some functions of the endpoints of level sets related to fuzzy weights and thresholds. Here the triangular and trapezoidal fuzzy numbers are generalized to general ones. So the results in the section is general and can be widely applied in real fields.

4.2.1 Preliminaries

Before developing learning algorithms, we at first recall the interval arithmetic, the detail related please see [2, 73]. Let $[a^L,\ a^U]$, $[b^L,\ b^U]$ be closed intervals. By '$*$' we denote '$+$' '$-$' '$\times\ (\cdot)$' and '$\div$', respectively. If let

$$I_* = \inf\{x * y | x \in [a^L, a^U], y \in [b^L, b^U]\},$$
$$I^* = \sup\{x * y | x \in [a^L, a^U], y \in [b^L, b^U]\},$$

we define $[a^L, a^U]*[b^L, b^U] = [I_*,\ I^*]$. Easily we can show

$$[a^L,\ a^U]+[b^L,\ b^U] = [a^L + b^L,\ a^U + b^U],$$
$$[a^L, a^U]-[b^L, b^U] = [a^L - b^U,\ a^U - b^L],$$

moreover, $[a^L,\ a^U]\times[b^L,\ b^U] = [c^L,\ c^U]$, where

$$c^L = \min\{a^L\cdot b^L, a^L\cdot b^U, a^U\cdot b^L, a^U\cdot b^U\},\ c^U = \max\{a^L\cdot b^L, a^L\cdot b^U, a^U\cdot b^L, a^U\cdot b^U\}.$$

If $0 \notin [b^L, b^U]$, we can obtain the operation law of interval division:

$$\frac{1}{[b^L, b^U]} = \left[\frac{1}{b^U},\ \frac{1}{b^L}\right],\quad [a^L, a^U]\div[b^L, b^U] = [a^L, a^U]\times\frac{1}{[b^L, b^U]}.$$

If $f : \mathbb{R} \longrightarrow \mathbb{R}$ is a monotone function, f can be extended as follows:

$$f([a^L, a^U]) = [f(a^L) \wedge f(a^U),\ f(a^L) \vee f(a^U)].$$

Next, we restrict the general fuzzy number space to a smaller class, i.e. $\mathcal{F}_{0c}(\mathbb{R})$, satisfying $\forall\ \tilde{A} \in \mathcal{F}_{0c}(\mathbb{R})$, if let $\text{Supp}(\tilde{A}) = [a^1,\ a^2]$, $\text{Ker}(\tilde{A}) = [e^1,\ e^2]$, then $\tilde{A}(\cdot)$ is increasing on $[a^1,\ e^1)$ (i.e. for arbitrary $x_1,\ x_2 \in [a^1,\ e^1)$, $x_1 < x_2, \Longrightarrow \tilde{A}(x_1) < \tilde{A}(x_2)$); and is decreasing on $(e^2,\ a^2]$(i.e. for arbitrary $x_1,\ x_2 \in (e^2,\ a^2]$, $x_1 < x_2, \Longrightarrow \tilde{A}(x_1) > \tilde{A}(x_2)$). Obviously, $\mathcal{F}_{0c}(\mathbb{R})$ is closed under the extended operations '+', '·', and '−', respectively. Furthermore, $\mathcal{F}_{0c}(\mathbb{R})$ includes the fuzzy numbers often used in application, such as triangular fuzzy number, trapezoidal fuzzy number and so on.

Theorem 4.2 *For any* $\varepsilon > 0$, *and* $\tilde{A}, \tilde{B} \in \mathcal{F}_{0c}(\mathbb{R})$, *if there are* $m_0 \in \mathbb{N}$, *and constants* $\alpha_0, \alpha_1, ..., \alpha_{m_0} \in [0,1] : 0 = \alpha_0 < \alpha_1 < \cdots < \alpha_{m_0} = 1$, *satisfying* $d_H(\tilde{A}_{\alpha_{k-1}}, \tilde{A}_{\alpha_k}) < \varepsilon/3$, $d_H(\tilde{B}_{\alpha_{k-1}}, \tilde{B}_{\alpha_k}) < \varepsilon/3$ $(k = 1, ..., m_0)$. *Then by the fact that* $\forall k \in \{0, 1, ..., m_0\}$, $d_H(\tilde{A}_{\alpha_k}, \tilde{B}_{\alpha_k}) < \varepsilon/3$, *it follows that* $D(\tilde{A}, \tilde{B}) \le \varepsilon$.

Proof. Given arbitrarily $\alpha \in [0,1]$, it follows that there is $k \in \{1, ..., m_0\}$, satisfying $\alpha \in [\alpha_{k-1}, \alpha_k]$. So we have

$$d_H(\tilde{A}_\alpha, \tilde{B}_\alpha) \le d_H(\tilde{A}_\alpha, \tilde{A}_{\alpha_k}) + d_H(\tilde{A}_{\alpha_k}, \tilde{B}_{\alpha_k}) + d_H(\tilde{B}_{\alpha_k}, \tilde{B}_\alpha). \qquad (4.16)$$

Considering $\tilde{A}, \tilde{B} \in \mathcal{F}_{0c}(\mathbb{R})$, and the definition of $\mathcal{F}_{0c}(\mathbb{R})$, we can obtain the following inequalities: $d_H(\tilde{A}_\alpha, \tilde{A}_{\alpha_k}) \le d_H(\tilde{A}_{\alpha_{k-1}}, \tilde{A}_{\alpha_k})$ and $d_H(\tilde{B}_\alpha, \tilde{B}_{\alpha_k}) \le d_H(\tilde{B}_{\alpha_{k-1}}, \tilde{B}_{\alpha_k})$. By (4.16) it follows that

$$\begin{aligned} d_H(\tilde{A}_\alpha, \tilde{B}_\alpha) &\le d_H(\tilde{A}_{\alpha_{k-1}}, \tilde{A}_{\alpha_k}) + d_H(\tilde{A}_{\alpha_k}, \tilde{B}_{\alpha_k}) + d_H(\tilde{B}_{\alpha_{k-1}}, \tilde{B}_{\alpha_k}) \\ &< \frac{\varepsilon}{3} + \frac{\varepsilon}{3} + \frac{\varepsilon}{3} = \varepsilon, \end{aligned}$$

by which we can prove the theorem. □

For any $\gamma \in \mathbb{N}$, let $\tilde{A} \in \mathcal{F}_{0c}(\mathbb{R})$, and denote $\tilde{A}_{\alpha_k} = [a_k^1,\ a_k^2]$, where $\alpha_k = k/\gamma$ $(k = 0, 1, ..., \gamma)$. By continuously increasing curves we link up such points on the curve $y = \tilde{A}(\cdot)$ as $(a_0^1, 0)$, $(a_1^1, \tilde{A}(a_1^1)), ..., (a_\gamma^1, 1)$ successively. By continuously decreasing curves we link up the points $(a_\gamma^2, 1), (a_{\gamma-1}^2, \tilde{A}(a_{\gamma-1}^2)), ..., (a_0^2, 0)$, successively. Thus, we obtain a fuzzy number $c_\gamma(\tilde{A}) \in \mathcal{F}_{0c}(\mathbb{R}) : \text{Supp}(c_\gamma(\tilde{A})) = \text{Supp}(\tilde{A})$; $\text{Ker}(c_\gamma(\tilde{A})) = \text{Ker}(\tilde{A})$; $c_\gamma(\tilde{A})_{\alpha_k} = \tilde{A}_{\alpha_k}$ $(k = 0, 1, ..., \gamma)$. We call $c_\gamma(\tilde{A})$ a defined-piecewise fuzzy number of $\tilde{A}$. If the curve segments linked up successively become line segments, $c_\gamma(\tilde{A})$ is called a symmetric polygonal fuzzy number, which is a main objective in Chapter V. By Theorem 4.2 we can show the following fact.

Corollary 4.1 *Let* $\widetilde{A}\in \mathcal{F}_{0c}(\mathbb{R})$. *Then when* $\gamma \to +\infty$, $D(\widetilde{A},\ c_\gamma(\widetilde{A})) \to 0$, *that is,* $\lim\limits_{\gamma\to+\infty} D(\widetilde{A},\ c_\gamma(\widetilde{A})) = 0$.

Proof. For $\widetilde{A}\in \mathcal{F}_{0c}(\mathbb{R})$, $\alpha_k = k/\gamma$ $(k = 0,1,...,\gamma)$, Let $\widetilde{A}_{\alpha_k}= [a_k^1, a_k^2]$. Then by the fact $\widetilde{A}(\cdot)$ is increasing and right continuous on (a_0^1, a_γ^1), and decreasing and left continuous on (a_γ^2, a_0^2), it follows that $\forall\varepsilon > 0$, there is $\gamma_0 \in \mathbb{N}$, such that $\forall\gamma \in \mathbb{N} : \gamma > \gamma_0$, $|a_k^1-a_{k-1}^1| < \varepsilon/3$, $|a_k^2-a_{k-1}^2| < \varepsilon/3$. Thus, $d_H(\widetilde{A}_{\alpha_k}, \widetilde{A}_{\alpha_{k-1}}) < \varepsilon/3$ $(k = 1,...,\gamma)$. But $\widetilde{A}_{\alpha_k}= c_\gamma(\widetilde{A})_{\alpha_k}$, Theorem 4.2 implies $D(\widetilde{A},\ c_\gamma(\widetilde{A})) < \varepsilon$. So $\lim\limits_{\gamma\to+\infty} D(\widetilde{A},\ c_\gamma(\widetilde{A})) = 0$. □

4.2.2 Calculus of $\vee - \wedge$ functions

As the preliminaries of developing learning algorithms, let us now study the derivatives of $\vee - \wedge$ functions and derivation operation laws (see [81], [82]). By $\vee - \wedge$ function class $\mathcal{L}_{mm}$ we mean the collection of all functions generated with the following rules [81]:

(i) $C^1(\mathbb{R}) \subset \mathcal{L}_{mm}$;

(ii) If $f, g \in \mathcal{L}_{mm}$, then $f \vee g,\ f \wedge g \in \mathcal{L}_{mm}$, where

$$(f \vee g)(x) = f(x) \vee g(x),\ (f \wedge g)(x) = f(x) \wedge g(x);$$

(iii) If $f \in \mathcal{L}_{mm}$, and $F \in C^1(\mathbb{R})$, then $F(f) \in \mathcal{L}_{mm}$.

The basis forms of $\vee - \wedge$ functions are $f \vee g$ and $f \wedge g$, where $f, g \in C^1(\mathbb{R})$. In the following we mainly aim at the derivatives related to $\vee - \wedge$ functions. To this end, let

$$\mathrm{lor}(\cdot) : \mathbb{R} \longrightarrow \mathbb{R},\ \ \mathrm{lor}(x) = \begin{cases} 1, & x > 0, \\ \dfrac{1}{2}, & x = 0, \\ 0, & x < 0. \end{cases} \tag{4.17}$$

For given $p \in \mathbb{N}$, let $f_1, ..., f_p \in C^1(\mathbb{R})$. By the following theorem, we can characterize the differentiability of the $\vee$–$\wedge$ functions $f_1\vee\cdots\vee f_p$ and $f_1\wedge\cdots\wedge f_p$.

Theorem 4.3 *Suppose* $f_1, ..., f_p \in C^1(\mathbb{R})$. *Write* $F = f_1 \vee\cdots\vee f_p$, *and* $G = f_1 \wedge\cdots\wedge f_p$. *For* $x_0 \in \mathbb{R}$, *let* $I(x_0) = \{i \in \{1, ..., p\} \big| f_i(x_0) = F(x_0)\}$; $J(x_0) = \{i \in \{1, ..., p\} \big| f_i(x_0) = G(x_0)\}$. *Then*

(i) F *is differentiable at* $x_0 \iff$ *either* $I(x_0)$ *is a singleton set or,* $\forall i, j \in I(x_0)$, $f_i'(x_0) = f_j'(x_0)$;

(ii) G *is differentiable at* $x_0 \iff$ *either* $J(x_0)$ *is a singleton set or,* $\forall i, j \in J(x_0)$, $f_i'(x_0) = f_j'(x_0)$.

Proof. It suffices to prove (i) because the proof of (ii) is similar. Assume that F is differentiable at x_0, and $I(x_0)$ is not a singleton. For given $i, j \in I(x_0)$, satisfying the following equalities:

$$f_i'(x_0)=\bigvee_{k\in I(x_0)}\{f_k'(x_0)\},\ f_j'(x_0)=\bigwedge_{k\in I(x_0)}\{f_k'(x_0)\}.$$

If $f_i'(x_0)\neq f_j'(x_0)$, then $f_i'(x_0)>f_j'(x_0)$. Since $f_i,\ f_j\in C^1(\mathbb{R})$, moreover $f_i(x_0)=f_j(x_0)=F(x_0)$, there is $\delta>0$, such that $\forall x\in(x_0-\delta,\,x_0)$, $f_j(x)>f_i(x)$, and $\forall x\in(x_0,x_0+\delta), f_j(x)<f_i(x)$. Consequently, by the fact $f_i,\ f_j\in C^1(\mathbb{R})$ it follows that

$$F'(x_0-0)=\lim_{x\to x_0-0}\frac{F(x)-F(x_0)}{x-x_0}\le\lim_{x\to x_0-0}\frac{f_j(x)-f_j(x_0)}{x-x_0}=f_j'(x_0);$$
$$F'(x_0+0)=\lim_{x\to x_0+0}\frac{F(x)-F(x_0)}{x-x_0}\ge\lim_{x\to x_0+0}\frac{f_i(x)-f_i(x_0)}{x-x_0}=f_i'(x_0).$$

Considering F is differentiable at x_0, i.e. $F'(x_0-0)=F'(x_0+0)=F'(x_0)$, we obtain $f_j'(x_0)\ge F'(x_0)\ge f_i'(x_0)$, which is a contradiction! So $\forall i,j\in I(x_0), f_i'(x_0)=f_j'(x_0)$.

Conversely, if $I(x_0)$ is a singleton set, let $i_0\in\{1,...,p\}$. Then $f_{i_0}(x_0)=F(x_0)$, furthermore, $\forall i\in\{1,...,p\}$, $i\neq i_0$, $\Longrightarrow f_{i_0}(x_0)>f_i(x_0)$. So there is a neighborhood $(x_0-\delta,\,x_0+\delta)$ of x_0, satisfying $\forall x\in(x_0-\delta,\,x_0+\delta)$, $\forall i\neq i_0$, $f_{i_0}(x)>f_i(x),\Longrightarrow F(x)=f_{i_0}(x)$. Thus, F is differentiable at x_0. If $I(x_0)$ is not a singleton, then $\forall i,j\in I(x_0), f_i'(x_0)=f_j'(x_0)\triangleq\lambda$. there is $\delta'>0$, such that $\forall x\in(x_0-\delta',\,x_0+\delta')$, $\forall i\in I(x_0)$, $\forall k\notin I(x_0)$, $f_i(x)>f_k(x)$. Hence $\forall x\in(x_0-\delta',\,x_0+\delta')$, there exists $i_x\in I(x_0)$, $f_{i_x}(x)=F(x)$. Arbitrarily given a sequence $\{y_k\}\subset\mathbb{R}:\ y_k\to x_0(k\to+\infty)$. Since $I(x_0)$ is a finite set, and the following set

$$I_0(x_0)\triangleq\{j\in I(x_0)|\text{ there is}\,\{y_{k^l}\}\subset\{y_k\}:f_{i_{y_{k^l}}}(y_{k^l})=f_j(y_{k^l})=F(y_{k^l})\}\neq\emptyset.$$

For each $j\in I_0(x_0)$, we have

$$\lim_{l\to+\infty}\frac{F(y_{k^l})-F(x_0)}{y_{k^l}-x_0}=\lim_{l\to+\infty}\frac{f_{i_{y_{k^l}}}(y_{k^l})-f_{i_{y_{k^l}}}(x_0)}{y_{k^l}-x_0}=\lim_{l\to+\infty}\frac{f_j(y_{k^l})-f_j(x_0)}{y_{k^l}-x_0}=\lambda. \tag{4.18}$$

Easily we can show that $\{y_k\}$ can be finitely partitioned into infinite subsequences, each of which can guarantee (4.18) to hold. Therefore

$$\lim_{k\to+\infty}\frac{F(y_k)-F(x_0)}{y_k-x_0}=\lambda,\Longrightarrow\lim_{x\to x_0}\frac{F(x)-F(x_0)}{x-x_0}=\lambda.$$

That is, F is differentiable at x_0. □

Choose $p=2$, by Theorem 4.3 we can easily show the following conclusion.

Corollary 4.2 *Let $f,g\in C^1(\mathbb{R})$, and $x_0\in\mathbb{R}$. Then the following conditions are equivalent:*

(i) $f(x_0) = g(x_0),\ f'(x_0) \neq g'(x_0)$;
(ii) $f \vee g$ *is non-differentiable at* x_0;
(iii) $f \wedge g$ *is non-differentiable at* x_0.

Similarly with Theorem 4.3, we can also characterize the differentiability of $h_1 = \text{lor}(f(x) - g(x))$ and $h_2 = \text{lor}(g(x) - f(x))$.

Theorem 4.4 *Suppose* $f,\ g \in C^1(\mathbb{R})$. *Define the functions* $h_1,\ h_2$ *respectively as follows:* $h_1(x) = \text{lor}(f(x) - g(x)),\ h_2(x) = \text{lor}(g(x) - f(x))$. *For given* $x_0 \in \mathbb{R}$, *the following conditions are equivalent:*
(i) h_1 *is differentiable at* x_0;
(ii) h_2 *is differentiable at* x_0;
(iii) Either $f(x_0) \neq g(x_0)$ *or,* $f(x_0) = g(x_0)$ *and there is* $\delta_0 > 0 :\ \forall x \in (x_0 - \delta_0,\ x_0 + \delta_0),\ f(x) = g(x)$.

Furthermore, when h_1, h_2 are differentiable at x_0, it follows that $h_1'(x_0) = h_2'(x_0) = 0$.

Proof. By the definition (4.17) for lor(·), easily we can show that $\forall x \in \mathbb{R},\ \text{lor}(f(x) - g(x)) = 1 - \text{lor}(g(x) - f(x))$. It suffices to prove (i) $\Longleftrightarrow$ (iii). Let (i) hold, and $f(x_0) = g(x_0)$. If (iii) does not hold, then $\forall k \in \mathbb{N}$, there exists $x_k \in (x_0 - 1/k,\ x_0 + 1/k) :\ f(x_k) \neq g(x_k)$. So there is a subsequence of $\{x_k\}$, and harmlessly we choose the subsequence to be $\{x_k\}$, satisfying $\{x_k\} \subset (x_0 - 1/k,\ x_0 + 1/k)$. Moreover, either $\forall k \in \mathbb{N},\ f(x_k) > g(x_k)$ or, $\forall k \in \mathbb{N},\ f(x_k) < g(x_k)$. It is no harm to assume $\forall k \in \mathbb{N},\ f(x_k) > g(x_k)$. Therefore

$$\lim_{k\to+\infty} \frac{h_1(x_k) - h_1(x_0)}{x_k - x_0} = \lim_{k\to+\infty} \frac{\text{lor}\big(f(x_k) - g(x_k)\big) - 1/2}{x_k - x_0} = \lim_{k\to+\infty} \frac{1 - 1/2}{x_k - x_0} = +\infty,$$

which contradicts the fact that h_1 is differentiable at x_0. So (iii) holds. Conversely, if $f(x_0) \neq g(x_0)$, let $f(x_0) > g(x_0)$. Then there is a neighborhood $(x_0 - \delta,\ x_0 + \delta) : \forall x \in (x_0 - \delta,\ x_0 + \delta), f(x) > g(x), \Longrightarrow h_1(x) = 1$. Hence h_1 is differentiable at x_0, moreover $h_1'(x_0) = 0$. If there exists $\delta_0 > 0 :\ \forall x \in (x_0 - \delta_0,\ x_0 + \delta_0),\ f(x) = g(x)$, then $\forall x \in (x_0 - \delta_0,\ x_0 + \delta_0),\ h_1(x) = 1/2$. Obviously, h_1 is differentiable at x_0, and $h_1'(x_0) = 0$. Thus, (i) holds. In summary, when $h_1,\ h_2$ are differentiable at x_0, $h_1'(x_0) = h_2'(x_0) = 0$. □

For $f, g \in C^1(\mathbb{R})$, define $\mathcal{U}_{mm}(f, g) = \big\{x \in \mathbb{R} \,|\, f \vee g \text{ is differentiable at } x\big\}$, and $\mathcal{U}_{\text{lor}}(f, g) = \big\{x \in \mathbb{R} \,|\, \text{lor}(f(x) - g(x)) \text{ is differentiable at } x\big\}$. By Corollary 4.2 and Theorem 4.4, easily we have

$$\mathcal{U}_{mm}(f, g) = \big\{x \in \mathbb{R} \,|\, f \wedge g \text{ is differentiable at } x\big\};$$
$$\mathcal{U}_{\text{lor}}(f, g) = \big\{x \in \mathbb{R} \,|\, \text{lor}(g(x) - f(x)) \text{ is differentiable at } x\big\},$$

also $\mathcal{U}_{\text{lor}}(f, g) \subset \mathcal{U}_{mm}(f, g)$, and the complement $\big[\mathcal{U}_{\text{lor}}(f, g)\big]^c, \big[\mathcal{U}_{mm}(f, g)\big]^c$ can be represented as following, respectively:

$$\big[\mathcal{U}_{\text{lor}}(f, g)\big]^c = \big\{x \in \mathbb{R} \,|\, f(x) = g(x), \forall \delta > 0, \exists\, x' \in (x - \delta, x + \delta) : f(x') \neq g(x')\big\};$$
$$\big[\mathcal{U}_{mm}(f, g)\big]^c = \big\{x \in \mathbb{R} \,|\, f(x) = g(x),\ f'(x) \neq g'(x)\big\}.$$

Theorem 4.5 *Let $f, g\in C^1(\mathbb{R})$. Then $\left[\mathcal{U}_{\text{lor}}(f,g)\right]^c$ at most is a numerable set, consequently $\left[\mathcal{U}_{mm}(f,g)\right]^c$ is also at most a numerable set.*

Proof. At first we show that $\forall a, b \in \mathbb{R} : a < b, \forall k \in \mathbb{N}$, the collection

$$\mathcal{V}(a,b,k)=\left\{x\in[a,b]\,|\,f(x)=g(x), \forall\delta>0, \exists\, x'\in(x-\delta,x+\delta) : |f(x')-g(x')|>\frac{1}{k}\right\}$$

at most is a finite set. Otherwise, there is a infinite sequence $\{x_j\} \subset [a,b] : \forall j \in \mathbb{N},\ f(x_j) = g(x_j)$. Moreover, $\forall\delta > 0, \exists\, x_j' \in (x_j - \delta,\ x_j + \delta) : |f(x_j') - g(x_j')| > 1/k$. Since $\{x_j\}$ has a convergent subsequence, which can harmlessly be assumed to be $\{x_j\}$, that is, let $\lim\limits_{j\to+\infty} x_j = x_0 \in [a, b]$. By the continuities of f, g it follows that $f(x_0) = g(x_0)$, and easily we can show that

$$\forall\delta > 0,\ \exists\, x_0' \in [a,b] \cap (x_0 - \delta,\ x_0 + \delta) : |f(x_0') - g(x_0')| > \frac{1}{k}.$$

And there is $\delta_0 > 0$, so that the following fact holds:

$$\forall x \in [a,\ b] \cap (x_0 - \delta_0,\ x_0 + \delta_0),\ \left|f(x) - f(x_0)\right|< \frac{1}{3k},\ \left|g(x) - g(x_0)\right|< \frac{1}{3k}.$$

Choose $x_0' \in [a,\ b] \cap (x_0 - \delta_0,\ x_0 + \delta_0)$. Then

$$\frac{1}{k} < |f(x_0') - g(x_0')| \le |f(x_0') - f(x_0)| + |g(x_0) - g(x_0')| < \frac{1}{3k} + \frac{1}{3k} = \frac{2}{3k},$$

This is a contradiction. So at most $\mathcal{V}(a,b,k)$ is a finite set. Also it is easy to show

$$\left[\mathcal{U}_{\text{lor}}(f,g)\right]^c= \bigcup_{j=1}^{+\infty}\bigcup_{k=1}^{+\infty} \mathcal{V}(-j,j,k),$$

hence $\left[\mathcal{U}_{\text{lor}}(f,g)\right]^c$ is at most a numerable set. Easily we have $\left[\mathcal{U}_{mm}(f,g)\right]^c\subset \left[\mathcal{U}_{\text{lor}}(f,g)\right]^c$. Thus, $\left[\mathcal{U}_{mm}(f,g)\right]^c$ is also at most a numerable set. □

By Theorem 4.5, if $f,\ g \in C^1(\mathbb{R})$, we may imply that $f \vee g,\ f \wedge g$, and $\text{lor}(f(\cdot) - g(\cdot))$, $\text{lor}(g(\cdot) - f(\cdot))$ are differentiable almost everywhere (a.e.). So using Corollary 4.2, we can easily show the following conclusion.

Corollary 4.3 *Suppose the functions $f,\ g$ are differentiable on $\mathbb{R}$, $h_1 = f \vee g$, $h_2 = f \wedge g$. Then h_1, h_2 are differentiable on $\mathbb{R}$ a.e., and if h_1, h_2 are differentiable at x, we have*

$$\frac{\mathrm{d}h_1(x)}{\mathrm{d}x} = \frac{\mathrm{d}(f(x) \vee g(x))}{\mathrm{d}x} = \text{lor}(f(x)-g(x))\frac{\mathrm{d}f(x)}{\mathrm{d}x} + \text{lor}(g(x)-f(x))\frac{\mathrm{d}g(x)}{\mathrm{d}x};$$
$$\frac{\mathrm{d}h_2(x)}{\mathrm{d}x} = \frac{\mathrm{d}(f(x) \wedge g(x))}{\mathrm{d}x} = \text{lor}(f(x)-g(x))\frac{\mathrm{d}g(x)}{\mathrm{d}x} + \text{lor}(g(x)-f(x))\frac{\mathrm{d}f(x)}{\mathrm{d}x}.$$

Specifically, if $a \in \mathbb{R}$, then

$$\frac{\mathrm{d}(a \vee f(x))}{\mathrm{d}x} = \text{lor}(f(x) - a)\frac{\mathrm{d}f(x)}{\mathrm{d}x};\quad \frac{\mathrm{d}(a \wedge f(x))}{\mathrm{d}x} = \text{lor}(a - f(x))\frac{\mathrm{d}f(x)}{\mathrm{d}x}.$$

Proof. Arbitrarily given $x \in \mathbb{R}$, it follows that

$$\begin{cases} f(x) \vee g(x) = f(x)\mathrm{lor}(f(x) - g(x)) + g(x)\mathrm{lor}(g(x) - f(x)), \\ f(x) \wedge g(x) = g(x)\mathrm{lor}(f(x) - g(x)) + f(x)\mathrm{lor}(g(x) - f(x)). \end{cases}$$

Assume that $f \vee g$, $f \wedge g$ are differentiable at x. If $\mathrm{lor}(f(\cdot) - g(\cdot))$, $\mathrm{lor}(g(\cdot) - f(\cdot))$ are differentiable at x, we can employ derivation laws and Theorem 4.4 to imply the conclusion. If $\mathrm{lor}(f(\cdot) - g(\cdot))$, $\mathrm{lor}(g(\cdot) - f(\cdot))$ are non-differentiable at x, By Corollary 4.2 and Theorem 4.4, $f(x) = g(x)$, $f'(x) = g'(x)$, which can also show our conclusion. □

If our study topics are restricted to a bounded interval $[a, b]$, the corresponding conclusions also hold, that is, we have

Remark 4.1 If we substitute the closed interval $[a, b]$ for $\mathbb{R}$, Theorem 4.3, Theorem 4.4 and Corollary 4.3 also hold.

4.2.3 Error function

In the sequel, we write $\alpha_k = k/\gamma$ $(k = 0, 1, ..., \gamma)$. Related to the regular FNN's as (4.5), introduce the following notations for $i = 1, ..., d$; $j = 1, ..., m$:

$$\big(\widetilde{X}_i\big)_{\alpha_k} = \big[x^1_{i(k)},\ x^2_{i(k)}\big];\quad \big(\widetilde{W}_{ij}\big)_{\alpha_k} = \big[w^1_{ij(k)},\ w^2_{ij(k)}\big];$$
$$\big(\widetilde{V}_j\big)_{\alpha_k} = \big[v^1_{j(k)},\ v^2_{j(k)}\big];\quad \big(\widetilde{\Theta}_j\big)_{\alpha_k} = \big[\theta^1_{j(k)},\ \theta^2_{j(k)}\big].$$

In the subsection we assume that the transfer function σ is continuous, the input $\widetilde{X}_i \in \mathcal{F}_{0c}(\mathbb{R})$ $(i = 1, ..., d)$. By (4.5) and the interval arithmetic we have

$$\widetilde{F}_{nn}(\widetilde{X}_1, ..., \widetilde{X}_d)_{\alpha_k} = \sum_{j=1}^{m} \big[v^1_{j(k)},\ v^2_{j(k)}\big] \cdot \sigma\big(\big[X^1_{j(k)},\ \ X^2_{j(k)}\big]\big), \tag{4.19}$$

where $X^1_{j(k)}$, $X^2_{j(k)}$ are defined respectively as follows:

$$\begin{cases} X^1_{j(k)} = \theta^1_{j(k)} + \sum\limits_{i=1}^{d} \min\big\{x^1_{i(k)} w^1_{ij(k)},\ x^1_{i(k)} w^2_{ij(k)},\ x^2_{i(k)} w^1_{ij(k)},\ x^2_{i(k)} w^2_{ij(k)}\big\}; \\ X^2_{j(k)} = \theta^2_{j(k)} + \sum\limits_{i=1}^{d} \max\big\{x^1_{i(k)} w^1_{ij(k)},\ x^1_{i(k)} w^2_{ij(k)},\ x^2_{i(k)} w^1_{ij(k)},\ x^2_{i(k)} w^2_{ij(k)}\big\}. \end{cases} \tag{4.20}$$

Since $\sigma(\cdot)$ is a continuous function, for $j = 1, ..., m$; $k = 1, ..., \gamma$, we conclude

$$\sigma\big(\big[X^1_{j(k)},\ X^2_{j(k)}\big]\big) \triangleq \big[\Psi_1\big(X^1_{j(k)}, X^2_{j(k)}\big),\ \Psi_2\big(X^1_{j(k)}, X^2_{j(k)}\big)\big], \tag{4.21}$$

where both $\Psi_1(\cdot,\cdot)$, $\Psi_2(\cdot,\cdot)$ are continuous. By (4.19), $\widetilde{F}_{nn}(\widetilde{X}_1, ..., \widetilde{X}_n)_{\alpha_k} \triangleq$

$\sum\limits_{j=1}^{m}\left[R^1_{j(k)},\ R^2_{j(k)}\right]$:

$$\begin{cases} R^1_{j(k)} &= \Big\{v^1_{j(k)}\cdot\Psi_1\big(X^1_{j(k)},\ X^2_{j(k)}\big)\Big\}\wedge\Big\{v^1_{j(k)}\cdot\Psi_2\big(X^1_{j(k)},\ X^2_{j(k)}\big)\Big\}\wedge \\ &\qquad \wedge\Big\{v^2_{j(k)}\cdot\Psi_1\big(X^1_{j(k)},\ X^2_{j(k)}\big)\Big\}\wedge\Big\{v^2_{j(k)}\cdot\Psi_2\big(X^1_{j(k)},\ X^2_{j(k)}\big)\Big\}; \\ R^2_{j(k)} &= \Big\{v^1_{j(k)}\cdot\Psi_1\big(X^1_{j(k)},\ X^2_{j(k)}\big)\Big\}\vee\Big\{v^1_{j(k)}\cdot\Psi_2\big(X^1_{j(k)},\ X^2_{j(k)}\big)\Big\}\vee \\ &\qquad \vee\Big\{v^2_{j(k)}\cdot\Psi_1\big(X^1_{j(k)},\ X^2_{j(k)}\big)\Big\}\vee\Big\{v^2_{j(k)}\cdot\Psi_2\big(X^1_{j(k)},\ X^2_{j(k)}\big)\Big\}. \end{cases} \tag{4.22}$$

If σ is nonnegative and increasing, by (4.21) it follows that

$$\Psi_1\big(X^1_{j(k)},\ X^2_{j(k)}\big)=\sigma\big(X^1_{j(k)}\big);\ \Psi_2\big(X^1_{j(k)},\ X^2_{j(k)}\big)=\sigma\big(X^2_{j(k)}\big).$$

(4.19) becomes as, $\widetilde{F}_{nn}(\widetilde{X}_1,...,\widetilde{X}_n)_{\alpha_k}=\sum\limits_{j=1}^{m}\left[v^1_{j(k)},\ v^2_{j(k)}\right]\cdot\left[\sigma\big(X^1_{j(k)}\big),\ \sigma\big(X^2_{j(k)}\big)\right]$. And (4.22) is as follows:

$$\begin{cases} R^1_{j(k)}=\Big(v^1_{j(k)}\cdot\sigma\big(X^1_{j(k)}\big)\Big)\wedge\Big(v^1_{j(k)}\cdot\sigma\big(X^2_{j(k)}\big)\Big); \\ R^2_{j(k)}=\Big(v^2_{j(k)}\cdot\sigma\big(X^1_{j(k)}\big)\Big)\vee\Big(v^2_{j(k)}\cdot\sigma\big(X^2_{j(k)}\big)\Big). \end{cases} \tag{4.23}$$

Let $\big((\widetilde{X}_1(1),...,\widetilde{X}_d(1));\ \widetilde{O}(1)\big),...,\big((\widetilde{X}_1(L),...,\widetilde{X}_d(L));\ \widetilde{O}(L)\big)$ be a family of fuzzy patterns for training FNN's, that is, when $(\widetilde{X}_1(l),...,\widetilde{X}_d(l))$ is the input of a FNN, the corresponding desired output is $\widetilde{O}(l)$, where $l=1,...,L$. Let

$$\big(\widetilde{X}_i(l)\big)_{\alpha_k}=\left[x^1_{i(k)}(l),\ x^2_{i(k)}(l)\right];\quad \big(\widetilde{O}(l)\big)_{\alpha_k}=\left[o^1_{(k)}(l),\ o^2_{(k)}(l)\right].$$

By the definition of metric $D(\cdot,\cdot)$ and Corollary 4.1, we conclude that

$$D\big(\widetilde{F}_{nn}(\widetilde{X}_1(l),...,\widetilde{X}_d(l)),\ \widetilde{O}(l)\big)\approx 0$$

if and only if for arbitrarily sufficiently large $\gamma\in\mathbb{N}$, the following fact holds: $\sum\limits_{k=1}^{\gamma}d_H\big((\widetilde{F}_{nn}(\widetilde{X}_1(l),...,\widetilde{X}_d(l)))_{\alpha_k},\ (\widetilde{O}(l))_{\alpha_k}\big)\approx 0$. Considering the equivalence between $d_H(\cdot,\ \cdot)$ and $d_E(\cdot,\ \cdot)$, we define the error function as follows:

$$E=\frac{1}{2}\sum_{l=1}^{L}\sum_{k=0}^{\gamma}\bigg(\Big[o^1_{(k)}(l)-\sum_{j=1}^{m}R^1_{j(k)}(l)\Big]^2+\Big[o^2_{(k)}(l)-\sum_{j=1}^{m}R^2_{j(k)}(l)\Big]^2\bigg), \tag{4.24}$$

where $R^1_{j(k)}(l),\ R^2_{j(k)}(l)$ and $X^1_{j(k)}(l),\ X^2_{j(k)}(l)$ are determined by (4.20) and (4.22) when the input of a FNN is $\big(\widetilde{X}_1(l),...,\widetilde{X}_d(l)\big)$ $(l=1,...,L)$.

4.2.4 Partial derivatives of error function

Now we focus on the differentiability of the error function $E(\cdot)$ with respect to the parameters $w^1_{ij(k)}$, $w^2_{ij(k)}$, $v^1_{j(k)}$, $v^2_{j(k)}$, $\theta^1_{j(k)}$, $\theta^2_{j(k)}$ $(i=1,...,d;\ j=1,...,m;\ k=0,1,...,\gamma)$, respectively. Assume that there are N such parameters. Then easily $N=2m(\gamma+1)(d+2)$. For given $i\in\{1,...,d\}$; $j\in\{1,...,m\}$; $k\in\{1,...,\gamma\}$, let $D^q_{(k)}(l)=\sum\limits_{j=1}^{m}R^q_{j(k)}(l)-o^q_{(k)}(l)$ $(q=1,2)$. By (4.24), if let $\rho=w^q_{ij(k)}, v^q_{j(k)}, \theta^q_{j(k)}$ $(q=1,2)$, respectively, we have

$$\frac{\partial E}{\partial \rho}=\sum_{l=1}^{L}\Big(D^1_{(k)}(l)\frac{\partial R^1_{j(k)}(l)}{\partial \rho}+D^2_{(k)}(l)\frac{\partial R^2_{j(k)}(l)}{\partial \rho}\Big). \tag{4.25}$$

Let $\sigma:\mathbb{R}\longrightarrow\mathbb{R}_+$ be a differentiable and increasing function. Let parameter ρ be $w^1_{ij(k)}$, $w^2_{ij(k)}$, $\theta^1_{j(k)}$, $\theta^2_{j(k)}$, respectively. By (4.23) (4.25) and Corollary 4.3, and the fact $\sigma(X^1_{j(k)})\le\sigma(X^2_{j(k)})$, it follows that

$$\begin{cases}\dfrac{\partial R^1_{j(k)}(l)}{\partial \rho}=v^1_{j(k)}\cdot\sum\limits_{t=1}^{2}\mathrm{lor}\big((-1)^{t+1}v^1_{j(k)}\big)\sigma'\big(X^t_{j(k)}(l)\big)\dfrac{\partial X^t_{j(k)}(l)}{\partial \rho};\\ \dfrac{\partial R^2_{j(k)}(l)}{\partial \rho}=v^2_{j(k)}\cdot\sum\limits_{p=1}^{2}\mathrm{lor}\big((-1)^{t}v^2_{j(k)}\big)\sigma'\big(X^t_{j(k)}(l)\big)\cdot\dfrac{\partial X^t_{j(k)}(l)}{\partial \rho};\\ \dfrac{\partial R^1_{j(k)}(l)}{\partial v^1_{j(k)}}=\mathrm{lor}(v^1_{j(k)})\cdot\sigma\big(X^1_{j(k)}(l)\big)+\mathrm{lor}(-v^1_{j(k)})\cdot\sigma\big(X^2_{j(k)}(l)\big);\\ \dfrac{\partial R^2_{j(k)}(l)}{\partial v^2_{j(k)}}=\mathrm{lor}(-v^2_{j(k)})\cdot\sigma\big(X^1_{j(k)}(l)\big)+\mathrm{lor}(v^2_{j(k)})\cdot\sigma\big(X^2_{j(k)}(l)\big).\end{cases} \tag{4.26}$$

Furthermore, $\dfrac{\partial R^1_{j(k)}(l)}{\partial v^2_{j(k)}}=\dfrac{\partial R^2_{j(k)}(l)}{\partial v^1_{j(k)}}\equiv 0$. We choose $\rho=w^q_{ij(k)}, \theta^q_{j(k)}$ $(q=1,2)$, respectively. By (4.20), Corollary 4.3, and $w^1_{ij(k)}\le w^2_{ij(k)}$, we get

$$\begin{cases}\dfrac{\partial X^1_{j(k)}(l)}{\partial w^1_{ij(k)}}=\sum\limits_{t=1}^{2}x^t_{i(k)}(l)\mathrm{lor}\big((-1)^{t+1}\underline{\Psi}_{ij(k)}(l)\big)\mathrm{lor}\big(x^t_{i(k)}(l)\big);\\ \dfrac{\partial X^2_{j(k)}(l)}{\partial w^1_{ij(k)}}=\sum\limits_{t=1}^{2}x^{3-t}_{i(k)}(l)\mathrm{lor}\big((-1)^{t+1}\overline{\Psi}_{ij(k)}(l)\big)\mathrm{lor}\big(-x^{3-t}_{i(k)}(l)\big);\\ \dfrac{\partial X^1_{j(k)}(l)}{\partial w^2_{ij(k)}}=\sum\limits_{t=1}^{2}x^t_{i(k)}(l)\mathrm{lor}\big((-1)^{t+1}\underline{\Psi}_{ij(k)}(l)\big)\mathrm{lor}\big(-x^t_{i(k)}(l)\big);\\ \dfrac{\partial X^2_{j(k)}(l)}{\partial w^2_{ij(k)}}=\sum\limits_{t=1}^{2}x^{3-t}_{i(k)}(l)\mathrm{lor}\big((-1)^{t+1}\overline{\Psi}_{ij(k)}(l)\big)\mathrm{lor}\big(x^{3-t}_{i(k)}(l)\big).\end{cases} \tag{4.27}$$

Moreover, $\dfrac{\partial X^q_{j(k)}(l)}{\partial \theta^q_{j(k)}} = 1;\ \dfrac{\partial X^{3-q}_{j(k)(l)}}{\partial \theta^q_{j(k)}} = 0\ (q = 1, 2)$, where

$$\underline{\Psi}_{ij(k)}(l) = \big(w^1_{ij(k)}x^2_{i(k)}(l)\big)\wedge\big(w^2_{ij(k)}x^2_{i(k)}(l)\big) - \big(w^1_{ij(k)}x^1_{i(k)}(l)\big)\wedge\big(w^2_{ij(k)}x^1_{i(k)}(l)\big);$$
$$\overline{\Psi}_{ij(k)}(l) = \big(w^1_{ij(k)}x^2_{i(k)}(l)\big)\vee\big(w^2_{ij(k)}x^2_{i(k)}(l)\big) - \big(w^1_{ij(k)}x^1_{i(k)}(l)\big)\vee\big(w^2_{ij(k)}x^1_{i(k)}(l)\big).$$

Synthesizing (4.25) (4.26) and (4.27) we obtain the following conclusion:

Theorem 4.6 *Let the transfer function $\sigma : \mathbb{R} \longrightarrow \mathbb{R}_+$ be a non-negatively differentiable, increasing function. Then the error function E is differentiable a.e. on $\mathbb{R}$ with respect to the parameters $w^q_{ij(k)}$, $v^q_{j(k)}$, $\theta^q_{j(k)}$ $(q = 1, 2; i = 1, ..., d;\ j = 1, ..., m;\ k = 0, 1, ..., \gamma)$. Moreover, if let*

$$\Phi^q_{ij(k)}(l) = \sum_{t=1}^{2} x^{3-t}_{i(k)}(l)\mathrm{lor}\big((-1)^{t+1}\overline{\Psi}_{ij(k)}(l)\big)\mathrm{lor}\big((-1)^q x^{3-t}_{i(k)}(l)\big);$$
$$\Gamma^q_{ij(k)}(l) = \sum_{t=1}^{2} x^{t}_{i(k)}(l)\mathrm{lor}\big((-1)^{t+1}\underline{\Psi}_{ij(k)}(l)\big)\mathrm{lor}\big((-1)^{q+1} x^{t}_{i(k)}(l)\big);$$
$$\Lambda_{j(k)}(l) = v^1_{j(k)}\mathrm{lor}(v^1_{j(k)})\cdot D^1_{(k)}(l) + v^2_{j(k)}\mathrm{lor}(-v^2_{j(k)})\cdot D^2_{(k)}(l);$$
$$\Delta_{j(k)}(l) = v^1_{j(k)}\mathrm{lor}(-v^1_{j(k)})\cdot D^1_{(k)}(l) + v^2_{j(k)}\mathrm{lor}(v^2_{j(k)})\cdot D^2_{(k)}(l).$$

The following partial derivative formula with respect to $w^q_{ij(k)}$, $v^q_{j(k)}$, $\theta^q_{j(k)}$ hold:

(i) $$\frac{\partial E}{\partial w^q_{ij(k)}} = \sum_{l=1}^{L}\Big(\Lambda_{j(k)}(l)\Gamma^q_{ij(k)}(l)\sigma'\big(X^1_{j(k)}(l)\big) + \Delta_{j(k)}(l)\Phi^q_{ij(k)}(l)\sigma'\big(X^2_{j(k)}(l)\big)\Big);$$

(ii) $$\frac{\partial E}{\partial v^1_{j(k)}} = \sum_{l=1}^{L} D^1_{(k)}(l)\cdot\Big(\mathrm{lor}\big(v^1_{j(k)}\big)\sigma\big(X^1_{j(k)}(l)\big) + \mathrm{lor}\big(-v^1_{j(k)}\big)\sigma\big(X^2_{j(k)}(l)\big)\Big);$$
$$\frac{\partial E}{\partial v^2_{j(k)}} = \sum_{l=1}^{L} D^2_{(k)}(l)\cdot\Big(\mathrm{lor}\big(-v^2_{j(k)}\big)\sigma\big(X^1_{j(k)}(l)\big) + \mathrm{lor}\big(v^2_{j(k)}\big)\sigma\big(X^2_{j(k)}(l)\big)\Big);$$

(iii) $$\frac{\partial E}{\partial \theta^1_{j(k)}} = \sum_{l=1}^{L}\big(\Lambda_{j(k)}(l)\big)\cdot\sigma(X^1_{j(k)});\quad \frac{\partial E}{\partial \theta^2_{j(k)}} = \sum_{l=1}^{L}\big(\Delta_{j(k)}(l)\big)\cdot\sigma(X^2_{j(k)}).$$

Proof. By (4.26) (4.27) and Theorem 4.4, we conclude that the following partial derivative formulas with respect to $w^q_{ij(k)}$ hold a.e.:

$$\begin{cases} \dfrac{\partial R^1_{j(k)}(l)}{\partial w^q_{ij(l)}} = \big[\sigma'\big(X^1_{j(k)}(l)\big)\Gamma^q_{ij(k)}\mathrm{lor}\big(v^1_{j(k)}\big) + \sigma'\big(X^2_{j(k)}(l)\big)\Phi^q_{ij(k)}\mathrm{lor}\big(-v^1_{j(k)}\big)\big]v^1_{j(k)}; \\ \dfrac{\partial R^2_{j(k)}(l)}{\partial w^q_{ij(l)}} = \big[\sigma'\big(X^1_{j(k)}(l)\big)\Gamma^q_{ij(k)}\mathrm{lor}\big(-v^2_{j(k)}\big) + \sigma'\big(X^2_{j(k)}(l)\big)\Phi^q_{ij(k)}\mathrm{lor}\big(v^2_{j(k)}\big)\big]v^2_{j(k)}. \end{cases} \quad (4.28)$$

By substituting (4.26) (4.28) for corresponding ones in (4.25) it follows that (i) holds. Similarly we can show (ii) (iii). □

4.2.5 Learning algorithm and simulation

Using the partial derivatives determined by Theorem 4.5, we can develop following iteration schemes related to α_k−level sets $\left[w^1_{ij(k)},\ w^2_{ij(k)}\right]$, $\left[v^1_{j(k)},\ v^2_{j(k)}\right]$ and $\left[\theta^1_{j(k)},\ \theta^2_{j(k)}\right]$ $(k = 0, 1, ..., \gamma)$ of the fuzzy weights $\widetilde{W}_{ij}$, $\widetilde{V}_j$ and the fuzzy threshold $\widetilde{\Theta}_j$. By (4.23) (4.24) the adjustable parameters are the endpoints: $w^1_{ij(k)},\ w^2_{ij(k)},\ v^1_{j(k)},\ v^2_{j(k)},\ \theta^1_{j(k)},\ \theta^2_{j(k)}$, where $i = 1, ..., n,\ j = 1, ..., m,\ k = 0, 1, ..., \gamma$. Rewriting the parameters as $w_1, ..., w_N$, we obtain the following iteration scheme:

$$w_r[t+1] = w_r[t] - \eta \cdot \frac{\partial E[t]}{\partial w_r[t]} + \alpha \cdot \triangle w_r[t-1]\ (r = 1, ..., N). \tag{4.29}$$

where $t = 0, 1, ...,$ is iteration step. And in each iteration we re-rank the following sets:

$$\begin{array}{l} \left\{w^1_{ij(0)}[t+1], ..., w^1_{ij(1)}[t+1], ..., w^2_{ij(\gamma)}[t+1]\right\}, \\ \left\{v^1_{j(0)}[t+1], .., v^1_{j(1)}[t+1], ..., v^2_{j(\gamma)}[t+1]\right\}, \\ \left\{\theta^1_{j(0)}[t+1], ..., \theta^1_{j(1)}[t+1], ..., \theta^2_{j(\gamma)}[t+1]\right\}, \end{array} .$$

respectively, so that

$$\begin{cases} w^1_{ij(0)}[t+1] \le w^1_{ij(1)}[t+1] \le \cdots \le w^1_{ij(\gamma)}[t+1] \le w^2_{ij(\gamma)}[t+1] \le \cdots \le w^2_{ij(0)}[t+1]; \\ v^1_{j(0)}[t+1] \le v^1_{j(1)}[t+1] \le \cdots \le v^1_{j(\gamma)}[t+1] \le v^2_{j(\gamma)}[t+1] \le \cdots \le v^2_{j(0)}[t+1]; \\ \theta^1_{j(0)}[t+1] \le \theta^1_{j(1)}[t+1] \le \cdots \le \theta^1_{j(\gamma)}[t+1] \le \theta^2_{j(\gamma)}[t+1] \le \cdots \le \theta^2_{j(0)}[t+1]; \end{cases}$$

To examine the learning capability of the regular FNN (4.5), let us now study a simulation example, that is, the FNN is employed to realize a family of fuzzy IF–THEN inference rules, approximately. Also we present the numerical comparison of our model with other fuzzified neural networks developed in [37], [38]. Suppose the inputs related to be a two dimensional variable (x_1, x_2), and the output to be an one dimensional variable y. IF–THEN rules are defined as follows:

$$\begin{array}{l} \text{IF } x_1 \text{ is high AND } x_2 \text{ is high THEN } y \text{ is high;} \\ \text{IF } x_1 \text{ is high AND } x_2 \text{ is low THEN } y \text{ is medium;} \\ \text{IF } x_1 \text{ is low AND } x_2 \text{ is high THEN } y \text{ is medium;} \\ \text{IF } x_1 \text{ is low AND } x_2 \text{ is low THEN } y \text{ is low.} \end{array} \tag{4.30}$$

Here the antecedent and consequent fuzzy sets 'high' 'low' 'medium' are fuzzy numbers defined on the closed interval [0, 4], respectively. Denote $\widetilde{H}_i$='high',

$\tilde{M}_e$= 'medium', $\tilde{L}_o$='low', which are shown in (a) of Figure 4.6, respectively. $\{((\tilde{H}_i, \tilde{H}_i), \tilde{H}_i), ((\tilde{H}_i, \tilde{L}_o), \tilde{M}_e), ((\tilde{L}_o, \tilde{H}_i), \tilde{M}_e), ((\tilde{L}_o, \tilde{L}_o), \tilde{L}_o)\}$ are chosen as the training patterns for designing the learning algorithms of FNN's as (4.5) and those in [37], [38]. In FNN (4.5) let $\gamma = 5$, i.e. $\alpha_k = k/5$ $(k = 0, 1, ..., 5)$, and there are five neurons in hidden layer, that is, $m = 5$. Suppose the transfer function $\sigma(x) = 1/(1+\exp(-x))$. With the following procedure we may complete learning process:

Algorithm 4.1: The fuzzy BP algorithm.

Step 0. Initialize connection weights and biases. Let iterative step $t = 0$, and all initial weights and biases are all generated, randomly in [-1,1].

Step 1. Let $w[t]$ be $w^1_{ij(k)}[t]$, $w^2_{ij(k)}[t]$, $v^1_{j(k)}[t]$, $v^2_{j(k)}[t]$, $\theta^1_{j(k)}[t]$, $\theta^2_{j(k)}[t]$, respectively. By Theorem 4.6 calculate $\partial E[t]/\partial w[t]$, where $i = 1, 2$; $j = 1, ..., 5$; $k = 0, 1, ..., 5$.

Step 2. Using (4.29) we iterate $w_r[t]$ to obtain $w_r[t+1]$.

Step 3. If $t \geq 2000$, or $|E[t] - E[t+1]| \leq 0.001$, stop the iterative procedure and output results, otherwise let $t = t + 1$, go to Step 1.

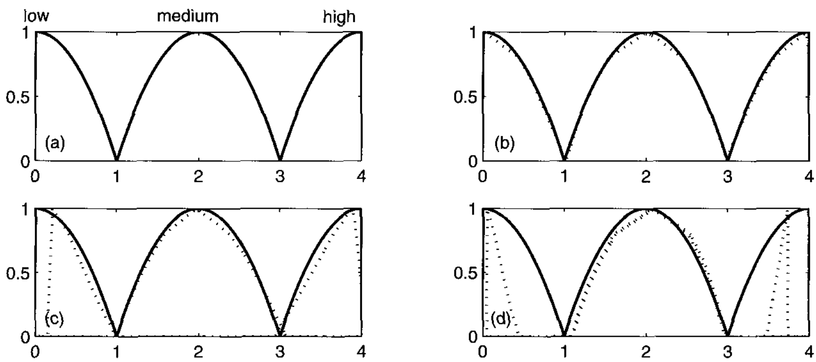


Figure 4.6 Membership curves of fuzzy numbers 'high' 'low' 'medium' : (a) desired curves; (b) desired curves (—) and actual output curves by our method (···); (c) desired curves (—) and actual output curves by Ishibuchi method in [38] (···); (d) desired curves (—) and actual output curves by Ishibuchi method in [37] (···).

Using Algorithm 4.1, we can train the regular FNN (4.5). And corresponding to the input patterns in the training set, we get the actual outputs of this FNN, as shown (b) of Figure 4.6.

In [38], Ishibuchi et al use the symmetric triangular fuzzy number weights and biases to construct a regular FNN, whose BP type learning algorithms adjust only two kinds of parameters — two endpoints of the triangles. Using such a FNN model we can also realize the inference rules in (4.30), approximately. The corresponding actual and desired outputs are shown in (c) of Figure 4.6 after 2000 iteration steps. Also, Ishibuchi et al take the real numbers as the connection weights and biases to construct a FNN model in [37]. And similarly with the convenient BP method they use (4.15) to develop a learning algorithm of the FNN. By iterating 2000 steps, we can get the actual outputs to approximate the rules in (4.30), as shown in (d) of Figure 4.6.

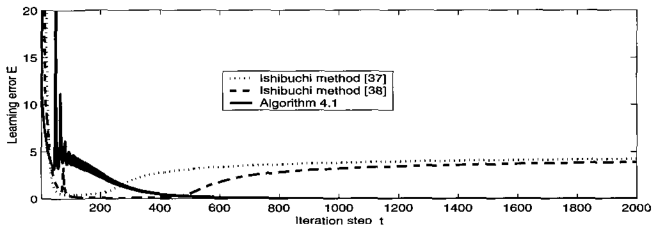


Figure 4.7 Error curves of different models

By comparing (b) (c) (d) of Figure 4.6, we can easily find that our FNN model (4.5) gives the best results, i.e. the error of FNN (4.5) is significantly lower than those Ishibuchi's FNN models [37], [38]. Also we can see from (d) of Figure 4.6 the larger error of the model in [37], and even getting the different outputs through this FNN when the corresponding inputs are $(\widetilde{Hi},\ \widetilde{Lo})$ and $(\widetilde{Lo},\ \widetilde{Hi})$, whereas the desired outputs are equal.

Figure 4.7 shows the curves of the error function defined by (4.24), corresponding to above three FNN models, i.e. the regular FNN (4.5) and two Ishibuchi's models [37], [38], where we let $\eta = 0.1$, $\alpha = 0.04$ in our Algorithm 4.1 and $\eta = 0.1$, $\alpha = 0.18$ in Ishibuchi's two algorithms, respectively. Also we can find from Figure 4.7 that despite some large fluctuation at the beginning of iteration, the square error of FNN (4.5) is lowest, and corresponding Ishibuchi's two FNN models, the square errors are approximately equal. So our result is also the best.

From the simulation example we can also find, like the convenient BP algorithm [12], the fuzzy BP algorithms are very sensitive to the learning constant η and momentum constant α, that is, very small variations of η and α can result in a large change of the square error E. Furthermore, the choices for η and α are blind. So in the next section we will focus on developing some systematic methods for choosing the learning constant η, rationally. One of the basic tools to do that is GA.

§4.3 Conjugate gradient algorithm for fuzzy weights

Since the learning constant η keeps unchanged in the iteration process of the fuzzy BP algorithm 4.1 for fuzzy weights of FNN's, the learning error can not be controlled, efficiently. Also the fuzzy BP algorithm is liable to fall into local minimum points, and it can not ensure some convergent speed. With different learning constants it can result in different results. Moreover, we have so far not solved the theoretic problem how suitable η can be selected. To study such a question systematically, this section will develop the fuzzy conjugate gradient(CG) algorithm for fuzzy weights. In each iteration step of the fuzzy CG algorithm, the learning constant η is adjusted, rationally. The optimal η is achieved by the genetic algorithm(GA). Also we show the algorithm is convergent.

4.3.1 Fuzzy CG algorithm and convergence

By $\mathbf{w} = (w_1, ..., w_d)$ we denote the vector consisting of all adjustable parameters $w^1_{ij(k)}$, $w^2_{ij(k)}$; $v^1_{j(k)}$, $v^2_{j(k)}$; $\theta^1_{j(k)}, \theta^2_{j(k)}$ $(k = 0, 1, ..., \gamma;\ i = 1, ..., n;\ j = 1, ..., m)$ related to (6). And $\nabla E(\mathbf{w})$ means the gradient vector of $E \triangleq E(\mathbf{w})$ defined by (4.24):

$$\begin{aligned}\nabla E(\mathbf{w}) = \Bigg(& \frac{\partial E(\mathbf{w})}{\partial w^1_{11(0)}}, ..., \frac{\partial E(\mathbf{w})}{\partial w^1_{11(\gamma)}}, \frac{\partial E(\mathbf{w})}{\partial w^2_{11(\gamma)}}, ..., \frac{\partial E(\mathbf{w})}{\partial w^2_{11(0)}}, ..., \frac{\partial E(\mathbf{w})}{\partial w^2_{dm(0)}},\\ & \frac{\partial E(\mathbf{w})}{\partial v^1_{1(0)}}, ..., \frac{\partial E(\mathbf{w})}{\partial v^1_{1(\gamma)}}, \frac{\partial E(\mathbf{w})}{\partial v^2_{1(\gamma)}}, ..., \frac{\partial E(\mathbf{w})}{\partial v^2_{1(0)}}, ..., \frac{\partial E(\mathbf{w})}{\partial v^2_{m(0)}}, \frac{\partial E(\mathbf{w})}{\partial \theta^1_{1(0)}}, ..., \frac{\partial E(\mathbf{w})}{\partial \theta^1_{1(\gamma)}},\\ & \frac{\partial E(\mathbf{w})}{\partial \theta^2_{1(\gamma)}}, ..., \frac{\partial E(\mathbf{w})}{\partial \theta^2_{1(0)}}, ..., \frac{\partial E(\mathbf{w})}{\partial \theta^2_{m(0)}}\Bigg)\\ = & \Big(\frac{\partial E(\mathbf{w})}{\partial w_1}, ..., \frac{\partial E(\mathbf{w})}{\partial w_N}\Big).\end{aligned}$$

For convenience, in this section let the input pattern $\big((\widetilde{X}_1(l), ..., \widetilde{X}_d(l)\big)$ $(l = 1, ..., L)$ of FNN (4.5) belong to $\mathcal{F}_0(\mathbb{R}_+)$. Theorem 4.6 can be simplified:

$$\begin{aligned}
&X^1_{j(k)} = \theta^1_{j(k)} + \sum_{i=1}^{d}\big(x^1_{i(k)}w^1_{ij(k)} \wedge x^2_{i(k)}w^1_{ij(k)}\big),\\
&X^2_{j(k)} = \theta^2_{j(k)} + \sum_{i=1}^{d}\big(x^1_{i(k)}w^2_{ij(k)} \vee x^2_{i(k)}w^2_{ij(k)}\big);\\
&\underline{\Psi}_{ij(k)}(l) = (x^2_{i(k)}(l) - x^1_{i(k)}(l))w^1_{ij(k)},\ \ \overline{\Psi}_{ij(k)}(l) = (x^2_{i(k)}(l) - x^1_{i(k)}(l))w^2_{ij(k)};\\
&\Phi^2_{ij(k)}(l) = x^2_{i(k)}(l)\mathrm{lor}(w^2_{ij(k)}) + x^1_{i(k)}(l)\mathrm{lor}(-w^2_{ij(k)}),\ \ \Phi^1_{ij(k)}(l) = 0;\\
&\Gamma^1_{ij(k)}(l) = x^1_{i(k)}(l)\mathrm{lor}(w^1_{ij(k)}) + x^2_{i(k)}(l)\mathrm{lor}(-w^1_{ij(k)}),\ \ \Gamma^2_{ij(k)}(l) = 0.
\end{aligned}$$

By Theorem 4.6 and (4.34) (4.35) we can show the following theorem.

Theorem 4.7 *For given fuzzy pattern pair* $\big(\widetilde{X}_1(l), ..., \widetilde{X}_d(l); \widetilde{O}(l)\big)$ $(l = 1, ..., L)$, *define the following space*

$$\mathcal{W}_{nn} = \{\mathbf{w} = (w_1, ..., w_N) \in \mathbb{R}^N \big| w_1 \cdots w_N \neq 0\}.$$

If the transfer function σ is non-negatively increasing and continuously differentiable, we have the following conclusions:

(i) The error function $E(\cdot)$ is differentiable in $\mathcal{W}_{nn}$, and $\nabla E(\cdot)$ is continuous in $\mathcal{W}_{nn}$;

(ii) For given $l_1 > 0, ..., l_N > 0$, let $\mathcal{W}_{nn}^{l_1...l_N} = \{(w_1, ..., w_N) \in \mathcal{W}_{nn} \big| |w_1| \geq l_1, ..., |w_N| \geq l_N\}$. Then $\nabla E(\cdot)$ is uniformly continuous in $\mathcal{W}_{nn}^{l_1...l_N}$.

Henceforth we call $\mathcal{W}_{nn}$ to be the nonzero weight space of the regular FNN (4.5). Using the derivatives established in Theorem 4.6 and Theorem 4.7, we can calculate $\nabla E(\mathbf{w})$ in $\mathcal{W}_{nn}$. Consequently, we design the following conjugate gradient algorithm.

Algorithm 4.2 The fuzzy CG algorithm.

Step 1. Initialize the weight vector $\mathbf{w} = \mathbf{w}[0] = (w_1, ..., w_N) \in \mathcal{W}_{nn}$, and let $t = 0$.

Step 2. Calculate the gradient $\nabla E(\mathbf{w}[t])$, and discriminate $\|\nabla E(\mathbf{w}[t])\| < \varepsilon$? if yes, go to Step 10; otherwise go to the following step.

Step 3. Put $\mathbf{v}[t] = \nabla E(\mathbf{w}[t])$, and let the direction $\mathbf{h}[t] = -\mathbf{v}[t]$.

Step 4. Calculate $\eta[t] > 0$:

$$\eta[t] = \max\{\eta \geq 0 | \, E(\mathbf{w}[t] + \lambda \mathbf{h}[t]) \text{ is decreasing with respect to } \lambda \in [0, \eta]\}.$$

Step 5. Let $\mathbf{w}[t+1] = \mathbf{w}[t] + \eta[t] \cdot \mathbf{h}[t]$.

Step 6. Discriminate $\mathbf{w}[t+1] \triangleq (w_1[t+1], ..., w_N[t+1]) \in \mathcal{W}_{nn}$? if yes, go to Step 8; otherwise we let $i_0 = \min\{i \in \{1, ..., N\} | w_i[t+1] = 0\}$, and $\delta = \min\{|w_i[t+1]| \, \big| w_i[t+1] \neq 0\}$, and set $0 < \delta' \ll \delta$.

Step 7. If $i_0 > 1$, let $w_{i_0}[t+1] = \delta' \cdot w_{i_0-1}[t+1]/|w_{i_0-1}[t+1]|$; if $i_0 = 1$, let $w_{i_0}[t+1] = \delta' \cdot w_N[t]/|w_N[t]|$. Go to Step 6.

Step 8. Calculate the gradient $\nabla E(\mathbf{w}[t+1])$, and discriminate whether $\|\nabla E(\mathbf{w}[t+1])\| < \varepsilon$? if yes, go to Step 10; otherwise go to the following step.

Step 9. Put $\mathbf{v}[t+1] = \nabla E(\mathbf{w}[t+1])$, and let $\mathbf{h}[t+1] = -\mathbf{v}[t+1] + \beta[t] \cdot \mathbf{h}[t]$. set $t = t+1$, go to Step 4. Here we choose $\beta[t]$ as follows [11, 76]:

$$\beta[t] = \frac{\langle \nabla E(\mathbf{w}[t]), \nabla E(\mathbf{w}[t]) \rangle}{\langle \nabla E(\mathbf{w}[t-1]), \nabla E(\mathbf{w}[t-1]) \rangle} = \frac{\|\nabla E(\mathbf{w}[t])\|^2}{\|\nabla E(\mathbf{w}[t-1])\|^2}.$$

Step 10. Output the weight vector $\mathbf{w}[t+1]$.

Remark 4.2 (i) Algorithm 4.2 is a fuzzy BP algorithm, in which the learning constant η and momentum constant α are adjusted in each iteration:

$$\mathbf{w}[t+1] = \mathbf{w}[t] - \eta[t] \cdot \nabla E(\mathbf{w}[t]) + \frac{\eta[t]\beta[t-1]}{\eta[t-1]} \cdot (\mathbf{w}[t] - \mathbf{w}[t-1]);$$

(ii) By Step 4 calculating the learning constant $\eta[t]$ is equivalent to solving the following minimum value problem:

$$E(\mathbf{w}[t] + \eta[t]\mathbf{h}[t]) = \min_{\lambda > 0}\{E(\mathbf{w}[t] + \lambda \mathbf{h}[t])\}. \tag{4.31}$$

The key step to realize Algorithm 4.2 is to find the learning constant $\eta[t]$, that is, get a solution of (4.31). To this end, we employ a non-accurate line search method, i.e. Armijo-Goldstein(A-G) line search [80] to establish $\eta[t]$ in Step 4. At first, for simplicity we leave out the iteration step t, and let $\eta[t] = \eta$, $\mathbf{w}[t] = \mathbf{w}$, $\mathbf{h}[t] = \mathbf{h}$. Let (see [11, 77, 80])

$$\begin{cases} E(\mathbf{w}) - E(\mathbf{w}+\eta\mathbf{h}) \geq -\eta b_1 \cdot \langle \mathbf{h}, \nabla E(\mathbf{w})\rangle, \\ \langle \mathbf{h}, \nabla E(\mathbf{w}+\eta\mathbf{h})\rangle \geq b_2 \langle \mathbf{h}, \nabla E(\mathbf{w})\rangle, \end{cases} \tag{4.32}$$

where b_1, b_2 are constants: $0 < b_1 < b_2 < 1$. For given $\mathbf{w} = (w_1, ..., w_N) \in \mathcal{W}_{nn}$, and $x > 0$, let $\psi(x) = E(\mathbf{w} + x\mathbf{h})$.

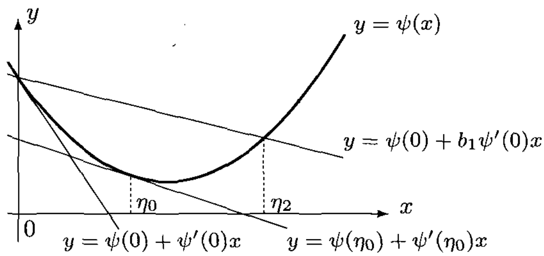


Figure 4.8 Illustration of constraints in line search.

Theorem 4.8 *(i) The left and right derivatives of the function $\psi(\cdot)$ exist in $[0, +\infty)$; (ii) For arbitrary $\mathbf{w} = (w_1, ..., w_N) \in \mathcal{W}_{nn}, \mathbf{h} = (h_1, ..., h_N) \in \mathbb{R}^N$, the function $\psi(\cdot)$ is differentiable in $\mathbb{R}_+$; (iii) If $t \in \mathbb{N}$, so that $\mathbf{w} = \mathbf{w}[t]$, $\mathbf{h} = \mathbf{h}[t]$, then ψ is differentiable at $\eta = 0$, and $\psi'(0) = \langle \mathbf{h}, \nabla E(\mathbf{w})\rangle < 0$.*

Proof. (i): By definition (4.24) for $E(\cdot)$ and Theorem 4.6, we can show that, the left and right derivatives of $\psi(\cdot)$ exist in $[0, +\infty)$, that is, (i) holds.

(ii): Obviously, $\mathbb{R}^n \setminus \mathcal{W}_{nn}$ is a finite set, so $\forall \mathbf{w} \in \mathcal{W}_{nn}, \mathbf{h} \in \mathbb{R}^N$. The set $\{x \in \mathbb{R} | \mathbf{w} + x\mathbf{h} \notin \mathcal{W}_{nn}\}$ is finite. Thus, Apart from finite x's, $\mathbf{w} + x\mathbf{h} \in \mathcal{W}_{nn}$, So Theorem 4.7 implies, $\psi(\cdot)$ is differentiable in $\mathbb{R}_+$ a.e..

(iii): Let $\mathbf{w} = (w_1, ..., w_N) = \mathbf{w}[t], \mathbf{h} = (h_1, ..., h_N) = \mathbf{h}[t]$. By Algorithm 4.2,$\mathbf{w} \in \mathcal{W}_{nn}$. So Theorem 4.7 implies, $E(\cdot)$ is differentiable at $\mathbf{w}$. Therefore

$$\begin{aligned} &\lim_{x\to 0+0} \frac{E(\mathbf{w}+x\mathbf{h}) - E(\mathbf{w})}{x} \\ = &\lim_{x\to 0+0} \frac{E(w_1 + xh_1, ..., w_N + xh_N) - E(w_1, w_2 + xh_2, ..., w_N + xh_N)}{x} \\ &+ \cdots + \lim_{x\to 0+0} \frac{E(w_1, ..., w_{N-1}, w_N + xh_N) - E(w_1, , ..., w_N)}{x} \\ = &\, h_1 \cdot \frac{\partial E(\mathbf{w})}{\partial w_1} + \cdots + h_N \cdot \frac{\partial E(\mathbf{w})}{\partial w_N} = \langle \mathbf{h}, \nabla E(\mathbf{w})\rangle, \end{aligned}$$

i.e. $\psi(\cdot)$ is differentiable at $x = 0$, $\psi'(0) = \langle \mathbf{h},\ \nabla E(\mathbf{w})\rangle$. By the definition of $\mathbf{h}$, $\langle \mathbf{h},\ \nabla E(\mathbf{w})\rangle < 0$, which implies the theorem. □

Let us now use Theorem 4.8 to show the existence of constants b_1, b_2 in the A-G condition (4.32). By Figure 4.8, we can find the first condition in (4.32) refers to the straight line $y = \psi(0) + b_1\psi'(0)x$; The second condition refers to the tangent line $y = \psi(\eta_0) + \psi'(\eta_0)x$.

Theorem 4.9 *Let* $\{\mathbf{w}[t]\,|t \in \mathbb{N}\} \subset \mathcal{W}_{nn}$, $\{\mathbf{h}[t]\,|t \in \mathbb{N}\} \subset \mathbb{R}^N$. *For each* $t \in \mathbb{N}$, *if denote* $\mathbf{w} = \mathbf{w}[t], \mathbf{h}[t] = \mathbf{h}$, *then there are* $b_1, b_2 : 0 < b_1 \le b_2 < 1$, *and* $\eta_0 \in [0, +\infty)$, *such that* $E(\cdot)$ *is differentiable at* $\mathbf{w} + \eta\mathbf{h}$, *and the A-G condition (4.32) holds.*

Proof. By Theorem 4.8, $\psi'(0) < 0$, so $x = 0$ is not a minimum point of $\psi(\cdot)$ in $[0, +\infty)$. Since $\psi(x) \ge 0$, we choose $b_1 \in (0, 1/2)$, so that at least there exist two points of intersection between the curve $y = \psi(x)$ and the straight line $y = \psi(0) + b_1\psi'(0)x$, as shown in Figure 4.8. Let

$$\eta_2 = \min\big\{\lambda > 0 | \psi(b) = \psi(0) + b_1\psi'(0)\lambda\big\}.$$

Then $\forall\,\lambda \in [0, \eta_2]$, $\psi(\lambda) \le \psi(0) + b_1\lambda\psi'(0)$, and $\psi(\eta_2) = \psi(0) + b_1\eta_2\psi'(0)$. that is, $\forall\lambda \in [0, \eta_2]$, we have

$$E(\mathbf{w}) - E(\mathbf{w} + \lambda\mathbf{h}) \ge -\lambda b_1\langle \mathbf{h}, \nabla E(\mathbf{w})\rangle.$$

Choose $b_2 \in (0, 1) : b_2 > b_1$. Obviously, there is $x_0 \in [0, +\infty)$, satisfying $\psi'(x_0) \ge 0 > b_1\psi'(0) \ge b_2\psi'(0)$. So let

$$\eta_0 = \min\big\{x \in \mathbb{R}_+ \big| \psi'(x)\,\text{exists, and}\,\psi'(x) \ge b_2\psi'(0)\big\};$$
$$\eta_1 = \min\big\{x \in \mathbb{R}_+ \big| \psi'(x)\,\text{exists, and}\,\psi'(x) \ge b_1\psi'(0)\big\}.$$

Then $\eta_0, \eta_1 \in [0, \eta_2] : \eta_0 < \eta_1$, and $\forall x \in (\eta_0, \eta_1)$, if $\psi'(x)$ exists, we have, $b_2\psi'(0) < \psi'(x) < b_1\psi'(0)$. Therefore, choose $\eta \in (\eta_0,\ \eta_1)$, so that $\psi(\cdot)$ is differentiable at η, i.e. $E(\cdot)$ is differentiable at $\mathbf{w} + \eta\mathbf{h}$. And $\langle \nabla E(\mathbf{w} + \eta\mathbf{h}), \mathbf{h}\rangle \ge b_2\psi'(0)$. Thus

$$E(\mathbf{w}) - E(\mathbf{w} + \eta\mathbf{h}) \ge -b_1\eta\langle \mathbf{h}, \nabla E(\mathbf{w})\rangle;\ \langle \nabla E(\mathbf{w} + \eta\mathbf{h}), \mathbf{h}\rangle \ge b_2\langle \mathbf{h}, \nabla E(\mathbf{w})\rangle.$$

The theorem is proved. □

By Theorem 4.9, it follows that the A–G condition (4.32) is rational as the constraint of minimum problem (4.31). Therefore, it can ensure some convergent speed of Algorithm 4.2.

Theorem 4.10 *Let* $\{\mathbf{w}[t]\big|t \in \mathbb{N}\}$ *be a vector sequence generated by Algorithm 4.2. Let there be* $l_1, ..., l_N : l_i > 0\ (i = 1, ..., N)$, *satisfying* $\{\mathbf{w}[t]\big|t \in \mathbb{N}\} \subset \mathcal{W}_{nn}^{l_1 \dots l_N}$, *and there be* $\theta_0 \in (0, \pi/2)$, *the angle of intersection between* $\mathbf{h}[t]$ *and* $-\nabla E(\mathbf{w}[t])$ *be* $\theta[t] :\ 0 \le \theta[t] \le \pi/2 - \theta_0$. *And in learning constant* *sequence* $\{\eta[t]\big|t \in \mathbb{N}\}$, $\eta[t]$ *satisfies the A-G condition (4.32) for each* $t \in \mathbb{N}$. *Then* $\lim_{p\to+\infty}\big\|\nabla E(\mathbf{w}[t])\big\|= 0$, *moreover, each cluster point of* $\{\mathbf{w}[t]\big|t \in \mathbb{N}\}$ *is the minimum value point of* $E(\cdot)$.

Proof. By Algorithm 4.2, $\forall t \in \mathbb{N}, \mathbf{w}[t] \in \mathcal{W}_{nn}$. So using Theorem 4.7 we imply the error function $E(\cdot)$ is continuously differentiable at $\mathbf{w}[t]$. Moreover, $\forall t \in \mathbb{N}, E(\mathbf{w}[t]) \geq 0$, by Theorem 4.9 it follows that

$$E(\mathbf{w}[t]) - E(\mathbf{w}[p+1]) \geq -b_1\eta[t]\cdot\big\langle\nabla E(\mathbf{w}[t]),\mathbf{h}[t]\big\rangle > 0. \qquad (4.33)$$

That is, $\{E(\mathbf{w}[t])\big|t \in \mathbb{N}\}$ is decreasing. So the limit $\lim_{t\to\infty} E(\mathbf{w}[t])$ exists, and $\lim_{t\to+\infty}\big(E(\mathbf{w}[t]) - E(\mathbf{w}[t+1])\big)= 0$. By A-G condition (4.32) we obtain

$$\begin{aligned} 0 &\leq \eta[t]\cdot\big\|\mathbf{h}[t]\big\|\cdot\big\|\nabla E(\mathbf{w}[t])\big\|\cdot\sin(\theta_0) \\ &\leq \eta[t]\cdot\big\|\mathbf{h}[t]\big\|\cdot\big\|\nabla E(\mathbf{w}[t])\big\|\cdot\cos(\theta[t]) \\ &= -\eta[t]\cdot\big\langle\nabla E(\mathbf{w}[t]),\mathbf{h}[t]\big\rangle\leq \frac{1}{b_1}\big(E(\mathbf{w}[t]) - E(\mathbf{w}[t+1])\big)\to 0(t\to+\infty), \end{aligned}$$

So $\lim_{t\to+\infty}\eta[t]\cdot\big\|\mathbf{h}[t]\big\|\cdot\big\|\nabla E(\mathbf{w}[t])\big\|= 0$. If let $\big\|\nabla E(\mathbf{w}[t])\big\|\not\to 0$ $(t\to+\infty)$, then there are $\varepsilon_0 > 0$, and a subsequence $\big\{\big\|\nabla E(\mathbf{w}[t_q])\big\|, q \in \mathbb{N}\big\}$: $\big\|\nabla E(\mathbf{w}[t_q])\big\|\geq \varepsilon_0$. Therefore, $\lim_{q\to+\infty}\eta[t]\cdot\big\|\mathbf{h}[t_q]\big\|= 0$. Since $\{\mathbf{w}[t]\,|t \in \mathbb{N}\} \subset \mathcal{W}_{nn}^{l_1\dots l_N}$, by Theorem 4.7, $\nabla E(\cdot)$ is uniformly continuous on $\mathcal{W}_{nn}^{l_1\dots l_N}$. Considering

$$\big\|\mathbf{w}[t_q+1] - \mathbf{w}[t_q]\big\|= \eta[t_q]\cdot\big\|\mathbf{h}[t_q]\big\|\to 0(q\to+\infty), \qquad (4.34)$$

we imply for sufficiently large q, $\big\|\nabla E(\mathbf{w}[t_q+1]) - \nabla E(\mathbf{w}[t_q])\big\|< \cos(\theta[t_q])\cdot\big\|\nabla E(\mathbf{w}[t_q])\big\|$. Thus

$$\begin{aligned} &\big|\big\langle\nabla E(\mathbf{w}[t_q+1]),\mathbf{h}[t_q]\big\rangle-\big\langle\nabla E(\mathbf{w}[t_q]),\mathbf{h}[t_q]\big\rangle\big| \\ &\quad\leq \big\|\nabla E(\mathbf{w}[t_q+1]) - \nabla E(\mathbf{w}[t_q])\big\|\cdot\big\|\mathbf{h}[t_q]\big\| \\ &\quad< \cos(\theta[t_q])\cdot\big\|\nabla E(\mathbf{w}[t_q])\big\|\cdot\big\|\mathbf{h}[t_q]\big\|= \big\langle\nabla E(\mathbf{w}[t_q]),\mathbf{h}[t_q]\big\rangle. \end{aligned}$$

Hence we can imply that

$$\big\langle\nabla E(\mathbf{w}[t_q+1]),\mathbf{h}[t_q]\big\rangle< 2\big\langle\nabla E(\mathbf{w}[t_q]),\mathbf{h}[t_q]\big\rangle< 0,$$

which contradicts A-G condition. So $\lim_{t\to+\infty}\big\|E(\mathbf{w}[t])\big\|= 0$. The second part of the theorem is the direct result of the fact that $\{E(\mathbf{w}[t]\,|t \in \mathbb{N})\}$ is decreasing. □

Using Theorem 4.10 and Theorem 4.9, we can show that Algorithm 4.2 converges to a minimum point of $E(\cdot)$ with some convergent speed. So Algorithm

4.2 can improve the fuzzy BP algorithm to overcome the drawbacks of choosing η, blindly and being liable to fall into local minimum points. In the following lets us establish optimal η at each iteration of Algorithm 4.2 by solving constraint type minimum problem (4.31) (4.32). The basic tool to do that is the GA.

4.3.2 GA for finding optimal learning constant

In Algorithm 4.2, the learning constant $\eta[t] \triangleq \eta$ can be obtained by solving the following constraint type minimum problem:

$$\begin{cases} E(\mathbf{w}+\eta\mathbf{h}) = \min\limits_{\lambda>0}\{E(\mathbf{w}+\lambda\mathbf{h})\}, \\ \text{Subject to} \begin{cases} E(\mathbf{w}) - E(\mathbf{w}+\lambda\mathbf{h}) + \lambda b_1 \cdot \langle \mathbf{h}, \nabla E(\mathbf{w})\rangle \geq 0; \\ \langle \mathbf{h}, \nabla E(\mathbf{w}+\lambda\mathbf{h})\rangle - b_2 \cdot \langle \mathbf{h}, \nabla E(\mathbf{w})\rangle \geq 0. \end{cases} \end{cases} \tag{4.35}$$

Now we employ a simple GA to find a solution of (4.35). To this end, considering $E(\mathbf{w}+\lambda\mathbf{h}) \geq 0$, we transform the constraint type problem into a unconstraint one [48, 60]:

$$\max_{\lambda>0}\{G(\lambda)\} \triangleq \max_{\lambda>0}\left\{\frac{1}{1+[1.1]^{E(\mathbf{w}+\lambda\mathbf{h})}} \cdot \frac{1}{[1.1]^{C(\lambda)}}\right\}, \tag{4.36}$$

where $C(\lambda)$ is defined as follows:

$$\begin{aligned} C(\lambda) = &\left|E(\mathbf{w}) - E(\mathbf{w}+\lambda\mathbf{h}) + \lambda b_1 \cdot \langle \mathbf{h}, \nabla E(\mathbf{w})\rangle\right| \\ &+\left|\langle \mathbf{h}, \nabla E(\mathbf{w}+\lambda\mathbf{h})\rangle - b_2 \cdot \langle \mathbf{h}, \nabla E(\mathbf{w})\rangle\right|. \end{aligned}$$

To guarantee the error sequence to belong to the given interval, let the learning constant lie in $[0, 1/5]$. The main steps of using GA to solve (4.36) are as follows:

(1) Code. Using a binary number β to express $\eta \in [0, 1/5]$, approximately. Such a β can represent a possible solution of (4.36), all of which constitute the solution space S_o of (4.36).

(2) Initialize. Let $n_0 \in \mathbb{N}$ be the population size at each evolution generation. Randomly we choose from S_o n_0 individuals $\lambda(0,1), \lambda(0,2), ..., \lambda(0, n_0)$ to constitute the initial population $P_u(0) = \{\lambda(0,1), ..., \lambda(0, n_0)\}$, and set $t = 0$.

(3) Calculate fitting. Given $\lambda(t,j) \in P_u(t)$, calculate $G(\lambda(t,j))$, where $P_u(t)$ means the $t-$ th generation population.

(4) Genetic selection. By roulette wheel selection method, the live probability of $\lambda(t,j)$ is $p_j = G(\lambda(t,j))/\sum\limits_{j'=1}^{n_0} G(\lambda(t,j'))$, that is, the probability that $\lambda(t,j)$ is reproduced to offspring is p_j.

(5) Genetic operator. Randomly choose from $P_u(t)$ the individual pair $(\lambda(t,j_1), \lambda(t,j_2))$ to match. C_r is the single-point crossover operator, and the crossover probability is p_c, that is, with probability p_c cross over the pair at a

randomly chosen point to form two offspring $C_r(\lambda(t,j_1),\lambda(t,j_2))$, which is the set $\{\lambda(t+1,j_1'),\lambda(t+1,j_2')\}$. Take an example as follows:

$$\begin{array}{llcl} \lambda(t,j_1):\ 1\,0\,1 & \overset{\text{crossover point}}{|} & 1\,0\,1 & \xrightarrow{C_r} \quad \lambda(t+1,j_1'):\ 1\,0\,1\,1\,1\,0 \\ \lambda(t.j_2):\ 0\,1\,0 & | & 1\,1\,0 & \qquad\quad \lambda(t+1,j_2'):\ 0\,1\,0\,1\,0\,1 \end{array}$$

If no crossover takes place, two offspring are exact copies of their respective parents.

By mutation operator the offspring mutate at each locus: $0 \longrightarrow 1$ or $1 \longrightarrow 0$ with mutation probability p_m. By the mutation operator we can accelerate the convergence to optimal solution, and insure the population against permanent fixation at any particular locus. To ensure the individuals close to optimal solution not to be violated, we let p_m be small, for example $p_m = 0.005$.

(6) Stop condition. Repeat above steps (3)–(5) until a suitable solution is achieved, or the iteration number exceeds the given bound.

4.3.3 Simulation examples

Let us now proceed to analyze the approximation realization of fuzzy IF—THEN rules by FNN's as (4.5) based on the fuzzy CG algorithm 4.2. Moreover, we present the comparison between the results based on our method and those in [23], [41]. Also we study the generalization capability of FNN (4.5). Choose the fuzzy IF—THEN rules as (4.30), which can be realized by FNN (4.5) using the fuzzy CG algorithm 4.2, approximately. In Algorithm 4.2, when employing GA to determine $\eta[t]$, we choose $n_0 = 20$. The parameters are represented by binary numbers with length $l = 6$. So code size is $2^6 = 64$, and code accuracy is $\delta = (1-0)/(5(2^l-1)) = 1/315$. The initial population $P_u(0) = \{\lambda(0,1),...,\lambda(0,20)\}$ is chosen, randomly. Define $\beta[t]$ as follows:

$$\beta[t] = \min\big(\|\nabla E(\mathbf{w}[t])\|^2/\|\nabla E(\mathbf{w}[t-1])\|^2,\ 0.05\cdot\eta[t-1]/\eta[t]\big).$$

In A-G condition let $b_1 = 0.4, b_2 = 0.45$. The crossover probability $p_c = 0.5$, mutation probability $p_m = 0,005$. The parameters and their codes have following relationships:

$$\begin{array}{llll} 0\,0\,0\,0\,0\,0 \longrightarrow 0; & & 0\,0\,0\,0\,0\,1 \longrightarrow \delta; \\ 0\,0\,0\,0\,1\,0 \longrightarrow 2\delta; & \cdots\cdots & 1\,1\,1\,1\,1\,1 \longrightarrow 1/5. \end{array}$$

And we can obtain the real value x corresponding to the binary code $\lambda = b_6b_5b_4b_3b_2b_1$:

$$x = 0 + \Big(\sum_{i=1}^{l} b_i\cdot 2^{i-1}\Big)\frac{1}{2^l-1} = \frac{1}{315}\cdot\sum_{i=1}^{6} b_i\cdot 2^{i-1}.$$

So we have, $\eta[t]\in[0,1/5]$. We choose the iteration number of Algorithm 4.2 to be 500, which is much smaller than that of Algorithm 4.1. After iteration process, we can get a suitable parameter vector $\mathbf{w}$, which will be used to complete a fuzzy rule table based on (4.30). As the iteration progresses, the learning constant η changes as Figure 4.9. As shown in Figure 4.10, the rational variation of η can result in quick decreasing of the square error $E(\cdot)$, which is approximately equal to that of Algorithm 4.1.

In [21–23], Dunyak et al establish a transformation for endpoints of level sets of fuzzy weights to leave out the constraints of designing the gradient descend learning algorithms of fuzzified neural networks. Using Dunyak's method, we can train regular FNN's based on rules (4.30). And we get the corresponding square error curve, as shown Figure 4.10 after 500 iterations. Ishibuchi et al in [41] use the trapezoidal fuzzy number weights and biases to construct a FNN model, in which four parameters related to four vertices of a trapezoid are adjustable. Also, based on (4.30) we train this FNN and the square error $E(\cdot)$ is shown in Figure 4.10. By comparison, obviously our result is best, and Ishibuchi's models [41] is better than those in [37, 38] since using the trapezoids to approximate the curves $\widetilde{Lo}$, $\widetilde{Me}$, $\widetilde{Hi}$, respectively is better than using real numbers or triangles. The error of Dunyak's method is largest since the corresponding transformation is defined by a series of sum formulas, which can result in a large accumulated error.

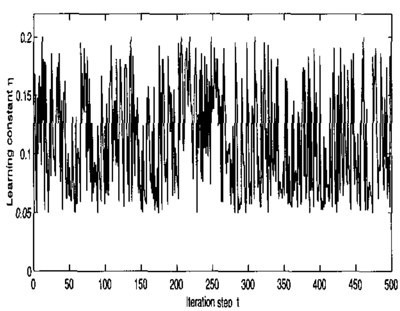


Figure 4.9 Learning constant curves

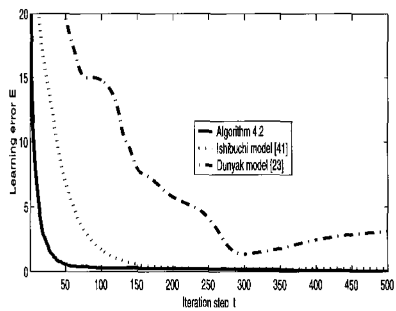


Figure 4.10 Error function curves

The fuzzy rules in (4.30) are shown as Figure 4.13 in the framework of a fuzzy rule table, in which only four rules out of 25 fuzzy IF—THEN rules are presented and others are missing.

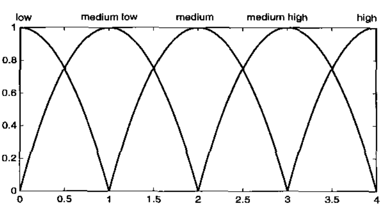


Figure 4.11

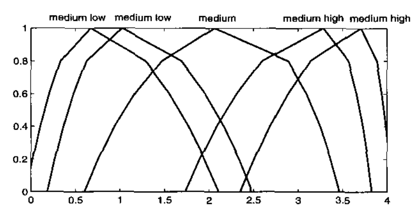


Figure 4.12

Figure 4.11: Antecedent and consequent fuzzy sets; Figure 4.12: Actual outputs of FNN (4.5)

$x_2 \backslash x_1$	$\widetilde{L_o}$	$\widetilde{M_L}$	$\widetilde{M_e}$	$\widetilde{M_H}$	$\widetilde{H_i}$
$\widetilde{L_o}$	$\widetilde{L_o}$				$\widetilde{M_e}$
$\widetilde{M_L}$					
$\widetilde{M_e}$					
$\widetilde{M_H}$					
$\widetilde{H_i}$	$\widetilde{M_e}$				$\widetilde{H_i}$

Figure 4.13

$x_2 \backslash x_1$	$\widetilde{L_o}$	$\widetilde{M_L}$	$\widetilde{M_e}$	$\widetilde{M_H}$	$\widetilde{H_i}$
$\widetilde{L_o}$	$\widetilde{L_o}$	$\widetilde{M_L}$	$\widetilde{M_L}$	$\widetilde{M_L}$	$\widetilde{M_e}$
$\widetilde{M_L}$	$\widetilde{M_L}$	$\widetilde{M_L}$	$\widetilde{M_L}$	$\widetilde{M_e}$	$\widetilde{M_H}$
$\widetilde{M_e}$	$\widetilde{M_L}$	$\widetilde{M_L}$	$\widetilde{M_e}$	$\widetilde{M_H}$	$\widetilde{M_H}$
$\widetilde{M_H}$	$\widetilde{M_L}$	$\widetilde{M_e}$	$\widetilde{M_H}$	$\widetilde{M_H}$	$\widetilde{M_H}$
$\widetilde{H_i}$	$\widetilde{M_e}$	$\widetilde{M_H}$	$\widetilde{M_H}$	$\widetilde{M_H}$	$\widetilde{H_i}$

Figure 4.14

Figure 4.13: Uncompleted fuzzy rule table; Figure 4.14: Fuzzy rule table completed by FNN (4.5)

Let us complete the rule table by assigning one of the five linguistic values as 'high' 'medium high ($\widetilde{M_H}$)' 'medium' 'medium low ($\widetilde{M_L}$)' and 'low', whose membership curves are shown in Figure 4.11, respectively, to the consequent of each missing rule. For example, we choose the input vector (x_1, x_2) to be

$$\big(\widetilde{L_o}, \widetilde{M_L}\big), \ \big(\widetilde{L_o}, \widetilde{M_e}\big), \ \big(\widetilde{M_L}, \widetilde{M_H}\big), \ \big(\widetilde{M_e}, \widetilde{H_i}\big), \ \big(\widetilde{M_H}, \widetilde{H_i}\big),$$

respectively, the corresponding outputs of FNN (4.5) are respectively shown in Figure 4.12, from which we can obtain their respective linguistic values: '$\widetilde{M_L}$' '$\widetilde{M_L}$' '$\widetilde{M_e}$' '$\widetilde{M_H}$' and '$\widetilde{M_H}$'. Similarly we can complete the other missing rules, as shown in Figure 4.14. Obviously, these consequents conform to inference sense of (4.30). So FNN's as (4.5) possesses strong generalization capability, which is advantageous over that of Ishibuchi's model in [38], since the similar rule table is completed based on nine fuzzy rules.

§4.4 Universal approximation to fuzzy valued functions

The connection weights and thresholds of the regular FNN (4.5) are fuzzy numbers. If no constraint exists for such fuzzy weights, it will be very difficult to study this FNN and its learning algorithms. So in this section we shall restrict the inputs related to be real numbers to construct a four-layer regular feedforward FNN, and develop a systematic theory about the universal approximation of such regular FNN's. It is show that the four-layer feerforward regular FNN's can be universal approximator of continuous fuzzy valued functions that $\mathbb{R}^d \longrightarrow \mathcal{F}_0(\mathbb{R})$, which provides us with the theoretic basis for choosing fuzzy functions to design fuzzy systems and fuzzy controllers in application.

4.4.1 Fuzzy valued Bernstein polynomial

Let us now extend a conventional multi-variate Bernstein polynomial to general one, which serves as a bridge to study universal approximation of regular FNN's. Given $m \in \mathbb{N}$, and a multi-variate function $f : [0,1]^d \longrightarrow \mathbb{R}$, then Bernstein polynomial $B_m(f)$ with respect to f is defined as follows: For $\mathbf{x}=(x_1,...,x_d)\in[0,1]^d$, we have

$$B_m(f;\mathbf{x})=\sum_{i_1,\dots,i_d=0}^{m}\binom{m}{i_1}\cdots\binom{m}{i_d}f\Big(\frac{i_1}{m},...,\frac{i_d}{m}\Big)x_1^{i_1}\cdots x_d^{i_d}(1-x_1)^{m-i_1}\cdots(1-x_d)^{m-i_d}. \tag{4.37}$$

Denote $K_{m;i_1\dots i_d}(\mathbf{x})=\binom{m}{i_1}\cdots\binom{m}{i_d}x_1^{i_1}\cdots x_d^{i_d}(1-x_1)^{m-i_1}\cdots(1-x_d)^{m-i_d}$, where $\mathbf{x}=(x_1,...,x_d)\in[0,1]^d$. Easily we obtain

$$\begin{aligned}\sum_{i_1,\dots,i_d=0}^{m}K_{m;i_1\dots i_d}(\mathbf{x})&=\Big[\sum_{i=0}^{m}\binom{m}{i}x_1^{i}(1-x_1)^{m-i}\Big]\cdots\Big[\sum_{i=0}^{m}\binom{m}{i}x_d^{i}(1-x_d)^{m-i}\Big]\\&=(x_1+1-x_1)^m\cdots(x_d+1-x_d)^m=1.\end{aligned} \tag{4.38}$$

Define the fuzzy valued Bernstein polynomial with respect to the fuzzy valued function $J:[0,1]^d\longrightarrow\mathcal{F}_0(\mathbb{R})$ as follows:

$$B_m(J;x_1,...,x_d)=\sum_{i_1,\dots,i_d=0}^{m}K_{m;i_1\dots i_d}(x_1,...,x_d)\cdot J\Big(\frac{i_1}{m},...,\frac{i_d}{m}\Big).$$

Now let us study the approximation of the fuzzy valued Bernstein polynomial. At first we present a useful conclusion.

Lemma 4.7 *Assume that* $\widetilde{A},\ \widetilde{A}_1,\ \widetilde{A}_2\in\mathcal{F}_0(\mathbb{R})$, $m\in\mathbb{N}$, *and the fuzzy set family* $\{\widetilde{W}_k\,|k=1,...,m\}$, $\{\widetilde{V}_k\,|k=1,...,m\}\subset\mathcal{F}_0(\mathbb{R})$. *Then*

(i) $D(\widetilde{A}\cdot\widetilde{A}_1,\widetilde{A}\cdot\widetilde{A}_2)\le|\widetilde{A}|\cdot D(\widetilde{A}_1,\widetilde{A}_2)$;

(ii) $D\Big(\sum_{k=1}^{m}\widetilde{W}_k,\sum_{k=1}^{m}\widetilde{V}_k\Big)\le\sum_{k=1}^{m}D(\widetilde{W}_k,\widetilde{V}_k)$.

Proof. $\forall\alpha\in[0,1]$, let $(\widetilde{A}_i)_\alpha=[a_{i\alpha}^1,a_{i\alpha}^2]$ $(i=1,2)$, $\widetilde{A}_\alpha=[a_\alpha^1,a_\alpha^2]$, and $(\widetilde{W}_k)_\alpha=[w_{k\alpha}^1,w_{k\alpha}^2]$, $(\widetilde{V}_k)_\alpha=[v_{k\alpha}^1,v_{k\alpha}^2]$ $(k=1,...,m)$. By the definition of $d_H(\cdot,\cdot)$ we imply the following equalities:

$$d_H\big((\widetilde{A}_1)_\alpha,(\widetilde{A}_2)_\alpha\big)=d_H\big([a_{1\alpha}^1,a_{1\alpha}^2],\ [a_{2\alpha}^1,a_{2\alpha}^2]\big)=|a_{1\alpha}^2-a_{2\alpha}^2|\vee|a_{1\alpha}^1-a_{2\alpha}^1|.$$

We shall prove (i) and (ii), respectively.

(i) Since $\forall\alpha\in[0,1]$, $(\widetilde{A}\cdot\widetilde{A}_1)_\alpha=\widetilde{A}_\alpha\cdot(\widetilde{A}_1)_\alpha$, if let $(\widetilde{A}\cdot\widetilde{A}_1)_\alpha=[\eta^1,\ \eta^2]$, we have

$$\eta^1=(a_\alpha^1\cdot a_{1\alpha}^1)\wedge(a_\alpha^2\cdot a_{1\alpha}^1)\wedge(a_\alpha^1\cdot a_{1\alpha}^2)\wedge(a_\alpha^2\cdot a_{1\alpha}^2),$$
$$\eta^2=(a_\alpha^1\cdot a_{1\alpha}^1)\vee(a_\alpha^2\cdot a_{1\alpha}^1)\vee(a_\alpha^1\cdot a_{1\alpha}^2)\vee(a_\alpha^2\cdot a_{1\alpha}^2).$$

With the same reason if let $(\widetilde{A}\cdot\widetilde{A}_2)_\alpha=[\xi^1,\ \xi^2]$, it follows that

$$\xi^1=(a_\alpha^1\cdot a_{2\alpha}^1)\wedge(a_\alpha^2\cdot a_{2\alpha}^1)\wedge(a_\alpha^1\cdot a_{2\alpha}^2)\wedge(a_\alpha^2\cdot a_{2\alpha}^2),$$
$$\xi^2=(a_\alpha^1\cdot a_{2\alpha}^1)\vee(a_\alpha^2\cdot a_{2\alpha}^1)\vee(a_\alpha^1\cdot a_{2\alpha}^2)\vee(a_\alpha^2\cdot a_{2\alpha}^2).$$

Furthermore, we can easily show the following facts:

$$|a^1_{\alpha}\cdot a^1_{1\alpha}-a^1_{\alpha}\cdot a^1_{2\alpha})|\leq|a^1_{1\alpha}-a^1_{2\alpha}|\cdot|\widetilde{A}\,|\leq|\widetilde{A}\,|\cdot D(\widetilde{A}_1,\widetilde{A}_2\,);$$
$$|a^2_{\alpha}\cdot a^1_{1\alpha}-a^2_{\alpha}\cdot a^1_{2\alpha})|\leq|a^1_{1\alpha}-a^1_{2\alpha}|\cdot|\widetilde{A}\,|\leq|\widetilde{A}\,|\cdot D(\widetilde{A}_1,\widetilde{A}_2\,);$$
$$|a^1_{\alpha}\cdot a^2_{1\alpha}-a^1_{\alpha}\cdot a^2_{2\alpha})|\leq|a^2_{1\alpha}-a^2_{2\alpha}|\cdot|\widetilde{A}\,|\leq|\widetilde{A}\,|\cdot D(\widetilde{A}_1,\widetilde{A}_2\,);$$
$$|a^2_{\alpha}\cdot a^2_{1\alpha}-a^2_{\alpha}\cdot a^2_{2\alpha})|\leq|a^2_{1\alpha}-a^2_{2\alpha}|\cdot|\widetilde{A}\,|\leq|\widetilde{A}\,|\cdot D(\widetilde{A}_1,\widetilde{A}_2\,).$$

So by Lemma 4.5 and (1.1) it follows that

$$\begin{aligned}d_H\big((\widetilde{A}\cdot\widetilde{A}_1)_{\alpha},\,(\widetilde{A}\cdot\widetilde{A}_2)_{\alpha}\big) &= d_H\big([\eta^1,\eta^2],\,[\xi^1,\xi^2]\big)\\ &= |\eta^1-\xi^1|\vee|\eta^2-\xi^2|\leq|\widetilde{A}\,|\cdot D(\widetilde{A}_1,\widetilde{A}_2\,).\end{aligned}$$

Therefore, $D(\widetilde{A}\cdot\widetilde{A}_1,\ \widetilde{A}\cdot\widetilde{A}_2)\leq|\widetilde{A}\,|\cdot D(\widetilde{A}_1,\ \widetilde{A}_2)$. which implies (i) holds.

(ii) At first it is easy to show that for arbitrary $t_1,...,t_m,s_1,...,s_m\in\mathbb{R}$,

$$\Big(\sum_{k=1}^m|t_k|\Big)\vee\Big(\sum_{k=1}^m|s_k|\Big)\leq\sum_{k=1}^m(|t_k|\vee|s_k|). \tag{4.39}$$

So $\forall\alpha\in[0,1]$, by (4.39) it follows that

$$\begin{aligned}&d_H\Big(\Big(\sum_{k=1}^m\widetilde{W}_k\,\Big)_{\alpha},\Big(\sum_{k=1}^m\widetilde{V}_k\,\Big)_{\alpha}\Big)=d_H\Big(\Big[\sum_{k=1}^m w^1_{k\alpha},\sum_{k=1}^m w^2_{k\alpha}\Big],\Big[\sum_{k=1}^m v^1_{k\alpha},\sum_{k=1}^m v^2_{k\alpha}\Big]\Big)\\ &=\Big|\sum_{k=1}^m(w^1_{k\alpha}-v^1_{k\alpha})\Big|\vee\Big|\sum_{k=1}^m(w^2_{k\alpha}-v^2_{k\alpha})\Big|\leq\Big[\sum_{k=1}^m|w^1_{k\alpha}-v^1_{k\alpha}|\Big]\vee\Big[\sum_{k=1}^m|w^2_{k\alpha}-v^2_{k\alpha}|\Big]\\ &\leq\sum_{k=1}^m\big(|w^1_{k\alpha}-v^1_{k\alpha}|\vee|w^2_{k\alpha}-v^2_{k\alpha}|\big)=\sum_{k=1}^m d_H\big((\widetilde{W}_k)_{\alpha},\ (\widetilde{V}_k)_{\alpha}\big).\end{aligned}$$

So $D\Big(\sum_{k=1}^m\widetilde{W}_k,\ \sum_{k=1}^m\widetilde{V}_k\,\Big)\leq\bigvee_{\alpha\in[0,1]}\Big(\sum_{k=1}^m d_H\big((\widetilde{W}_k)_{\alpha},\ (\widetilde{V}_k)_{\alpha}\big)\leq\sum_{k=1}^m D\big(\widetilde{W}_k,\ \widetilde{V}_k\big)$.
Thus, (ii) holds. □

Using Lemma 4.7 we can present the universal approximation of fuzzy valued Berstein polynomials.

Theorem 4.11 *Let* $J:\,[a_1,b_1]\times\cdots\times[a_d,b_d]\longrightarrow\mathcal{F}_0(\mathbb{R})$ *be a continuous fuzzy valued function. Then* $\forall\varepsilon>0$, *there is* $m\in\mathbb{N}$, *such that*

$$\forall(x_1,...,x_d)\in[a_1,b_1]\times\cdots\times[a_d,b_d],\ D(B_m(J;x_1,...,x_d),\ J(x_1,...,x_d))<\varepsilon.$$

Proof. Since by some suitable linear transformation we can turn $[a_i,b_i]$ $(i=1,...,d)$ into $[0,1]$, without loss generality, we can assume $[a_i,\,b_i]=[0,1]$ $(i=$

$1,...,d)$. By real analysis, we can show the following fact:

$$\forall t \in [0,1],\ m \in \mathbb{N},\ \sum_{i=0}^{m}\binom{m}{i}(i-mt)^2 t^i(1-t)^{m-i} = mt(1-t) \leq \frac{m}{4}. \quad (4.40)$$

Since J is continuous on $[0,1]^d$, it follows that J is uniformly continuous on $[0,1]^d$. So for $\varepsilon > 0$, there is $\delta > 0$, so that $\forall \mathbf{x} = (x_1, ..., x_d),\ \mathbf{y} = (y_1, ..., y_d) \in [0,1]^d,\ \|\mathbf{x}-\mathbf{y}\| < \delta, \Longrightarrow D(J(\mathbf{x}),\ J(\mathbf{y})) < \varepsilon/2$. Thus, there exists $m \in \mathbb{N}$, satisfying $\forall i_1, ..., i_d = 1, ..., m$,

$$\forall \mathbf{x}, \mathbf{y} \in \Big[\frac{i_1 - 1}{m},\ \frac{i_1}{m}\Big] \times \cdots \Big[\frac{i_d - 1}{m},\ \frac{i_d}{m}\Big],\ D(J(\mathbf{x}),\ J(\mathbf{y})) < \frac{\varepsilon}{2}. \quad (4.41)$$

By the continuity of J it is easy to show that $D(J(\mathbf{x}),\ J(\mathbf{y}))$ is continuous with respect to $(\mathbf{x},\mathbf{y})$ on $[0,1]^d \times [0,1]^d$. So there is $M > 0$, such that $\forall \mathbf{x},\ \mathbf{y} \in [0,1]^d,\ D\big(J(\mathbf{x}),\ J(\mathbf{y})\big) < M$. For each $\mathbf{x} = (x_1, ..., x_d) \in [0,1]^d$, by (4.38) and (4.40) (4.41), and Lemma 4.7 it follows that

$$\begin{aligned}
&D(B_m(J;\mathbf{x}),\ J(\mathbf{x})) = D\Big(\sum_{i_1,\dots,i_d=0}^{m} K_{m;i_1\dots i_d}(\mathbf{x}) J\Big(\frac{i_1}{m},...,\frac{i_d}{m}\Big),\ J(\mathbf{x})\Big)\\
&= D\Big(\sum_{i_1,\dots,i_d=0}^{m} K_{m;i_1\dots i_d}(\mathbf{x}) J\Big(\frac{i_1}{m},...,\frac{i_d}{m}\Big),\ \sum_{i_1,\dots,i_d=0}^{m} K_{m;i_1\dots i_d}(\mathbf{x}) J(\mathbf{x})\Big)\\
&\leq \sum_{i_1,\dots,i_d=0}^{m} K_{m;i_1\dots i_d}(\mathbf{x}) D\Big(J\Big(\frac{i_1}{m},...,\frac{i_d}{m}\Big),\ J(\mathbf{x})\Big)\\
&\leq \sum_{i_1,\dots,i_d:\|\mathbf{x}-(\frac{i_1}{m},\dots,\frac{i_d}{m})\|<\delta}^{m} K_{m;i_1\dots i_d}(\mathbf{x}) D\Big(J\Big(\frac{i_1}{m},...,\frac{i_d}{m}\Big),\ J(\mathbf{x})\Big) +\\
&\quad + \sum_{i_1,\dots,i_d:\|\mathbf{x}-(\frac{i_1}{m},\dots,\frac{i_d}{m})\|\geq\delta}^{m} K_{m;i_1\dots i_d}(\mathbf{x}) D\Big(J\Big(\frac{i_1}{m},...,\frac{i_d}{m}\Big),\ J(\mathbf{x})\Big)\\
&< \frac{\varepsilon}{2} + M \cdot \sum_{i_1,\dots,i_d:\|\mathbf{x}-(\frac{i_1}{m},\dots,\frac{i_d}{m})\|\geq\delta}^{m} K_{m;i_1\dots i_d}(\mathbf{x}) \Big[\frac{\|\mathbf{x}-(\frac{i_1}{m},\dots,\frac{i_d}{m})\|}{\delta}\Big]^2\\
&\leq \frac{\varepsilon}{2} + M \cdot \sum_{i_1,\dots,i_d:\|\mathbf{x}-(\frac{i_1}{m},\dots,\frac{i_d}{m})\|\geq\delta}^{m} K_{m;i_1\dots i_d}(\mathbf{x}) \Big[\frac{\sum_{k=1}^{d}(x_k - \frac{i_k}{m})^2}{\delta^2}\Big]\\
&\leq \frac{\varepsilon}{2} + \frac{M}{\delta^2} \cdot \sum_{i_1,\dots,i_d=0}^{m} K_{m;i_1\dots i_d}(x_1, \cdots, x_d) \Big[\sum_{k=1}^{d}\Big(x_k - \frac{i_k}{m}\Big)^2\Big]\\
&\leq \frac{\varepsilon}{2} + \frac{M}{m^2\delta^2} \cdot \sum_{k=1}^{d}\Big(\sum_{i_k=0}^{m}\tbinom{m}{i_k} x_k^{i_k}(1-x_k)^{m-i_k}(mx_k - i_k)^2\Big)\\
&\leq \frac{\varepsilon}{2} + \frac{Md}{m^2\delta^2} \cdot \frac{m}{4} = \frac{\varepsilon}{2} + \frac{Md}{4m\delta^2}.
\end{aligned}$$

Select $m > Md/(2\varepsilon \cdot \delta^2)$, the theorem is proved. □

Remark 4.3 If J is continuous on $\mathbb{R}^d$, then in Theorem 4.11, the set $[a_1, b_1] \times \cdots \times [a_d, b_d]$ can be extended as an arbitrary compact set of $\mathbb{R}^d$, in which the theorem holds.

4.4.2 Four-layer regular feedforward FNN

This subsection focuses mainly on proving that four-layer regular feedforward FNN's with real number inputs can be universal approximators of fuzzy valued functions. The basic tool is the fuzzy valued Bernstein polynomial. Let

$$\mathcal{P} = \Big\{P : \mathbb{R}^d \longrightarrow \mathcal{F}_0(\mathbb{R}) \big| P(\mathbf{x}) = \sum_{k=1}^{q} \widetilde{W}_k \cdot P_k(\mathbf{x}),$$
$$q \in \mathbb{N},\ \widetilde{W}_k \in \mathcal{F}_0(\mathbb{R}),\ P_k(\cdot) \text{ is } d \text{ variate polynomial}\Big\}.$$

Definition 4.2 Let $\mathcal{A}$ be a subclass of all fuzzy valued function that $\mathbb{R}^d \longrightarrow \mathcal{F}_0(\mathbb{R})$. Given $J : \mathbb{R}^d \longrightarrow \mathcal{F}_0(\mathbb{R})$, if $\forall \varepsilon > 0$, and for arbitrary compact set $U \subset \mathbb{R}^d$, there is $H \in \mathcal{A}$, such that $\forall \mathbf{x} \in U,\ D(J(\mathbf{x}),\ H(\mathbf{x})) < \varepsilon$. Then $\mathcal{A}$ is called a universal approximator of J.

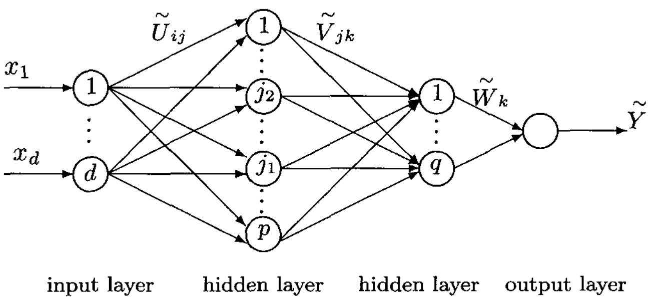


Figure 4.15 Four-layer regular feedforward FNN

By Theorem 4.11 and Remark 4.3, $\mathcal{P}$ is a universal approximator of arbitrary continuous fuzzy valued function $J : \mathbb{R}^d \longrightarrow \mathcal{F}_0(\mathbb{R})$. Let

$$\mathcal{H}[\sigma] = \Big\{F_N : \mathbb{R}^d \longrightarrow \mathcal{F}_0(\mathbb{R}) \big| F_N(\mathbf{x}) = \sum_{k=1}^{q} \widetilde{W}_k \cdot \Big(\sum_{j=1}^{p} \widetilde{V}_{jk} \cdot \sigma\big(\langle \mathbf{x}, \widetilde{\mathbf{U}}(j)\rangle + \widetilde{\Theta}_j\big)\Big)$$
$$p,\ q \in \mathbb{N},\ \widetilde{W}_k,\ \widetilde{V}_{jk},\ \widetilde{\Theta}_j \in \mathcal{F}_0(\mathbb{R}),\ \widetilde{\mathbf{U}}(j) \in \mathcal{F}_0(\mathbb{R})^d\Big\}. \tag{4.42}$$

Obviously, for each $H \in \mathcal{H}[\sigma]$, H is a regular feedforward FNN with two hidden layers, in which the first layer hidden neurons have the transfer function $\sigma(\cdot)$, and fuzzy threshold $\widetilde{\Theta}_j$, the second layer hidden neurons are linear, as shown in Figure 4.15. $\widetilde{\mathbf{U}}(j) = \big(\widetilde{U}_{1j}, ..., \widetilde{U}_{dj}\big)$.

Specifically, if let $\widetilde{\mathbf{U}}(j)$, $\widetilde{V}_{jk}$, $\widetilde{\Theta}_j$ be $\mathbf{u}(j) \in \mathbb{R}^d$, v_{jk}, $\theta_j \in \mathbb{R}$, respectively, we obtain the subset $\mathcal{H}_0[\sigma]$ of $\mathcal{H}[\sigma]$ as

$$\mathcal{H}_0[\sigma] = \Big\{F_N : \mathbb{R}^d \longrightarrow \mathcal{F}_0(\mathbb{R}) \big| F_N(\mathbf{x}) = \sum_{i=1}^{q} \widetilde{W}_k \cdot \Big(\sum_{j=1}^{p} v_{jk} \cdot \sigma\big(\langle \mathbf{u}(j), \mathbf{x}\rangle + \theta_j\big)\Big)$$
$$p,\ q \in \mathbb{N},\ \widetilde{W}_k \in \mathcal{F}_0(\mathbb{R}),\ v_{jk},\ \theta_j \in \mathbb{R},\ \mathbf{u}(j) \in \mathbb{R}^d\Big\}. \tag{4.43}$$

Before studying the universal approximation of $\mathcal{H}[\sigma]$ or $\mathcal{H}_0[\sigma]$, we present a useful property about universal approximators.

Lemma 4.8 *Let $\mathcal{A}$ be a universal approximator of the fuzzy valued function $J : \mathbb{R}^d \longrightarrow \mathcal{F}_0(\mathbb{R})$, and $\forall F_N \in \mathcal{A}$, F_N be continuous. Then J is also continuous.*

Proof. If assume the conclusion to be false, there are $\mathbf{x}^0 = (x_1^0, ..., x_d^0) \in \mathbb{R}^d$ and $\varepsilon_0 > 0$, such that $\forall \delta > 0$, there is $\mathbf{x}' = (x_1', ..., x_d') \in \mathbb{R}^d$: $\|\mathbf{x}_0 - \mathbf{x}'\| < \delta$, and $D(J(\mathbf{x}'), J(\mathbf{x}_0)) \geq \varepsilon_0$. Choose $\delta = 1, 1/2, ..., 1/k, ...$, respectively, we obtain the sequence $\{\mathbf{x}^k = (x_1^k, ..., x_d^k) | k \in \mathbb{N}\} \subset \mathbb{R}^d$, satisfying

$$\forall k \in \mathbb{N},\ \|\mathbf{x}^k - \mathbf{x}^0\| < \frac{1}{k},\ D\big(J(\mathbf{x}^k),\ J(\mathbf{x}^0)\big) \geq \varepsilon_0.$$

Let $U = \{\mathbf{x}^0, \mathbf{x}^1, ..., \mathbf{x}^k, ...\}$. Obviously, $U \subset \mathbb{R}^d$ is a compact set. By the assumption there is $F_N \in \mathcal{A}$, satisfying $\forall \mathbf{x} \in U,\ D\big(J(\mathbf{x}),\ F_N(\mathbf{x})\big) < \varepsilon_0/4$. On the other hand, since F_N is continuous at $\mathbf{x}^0$, there exists $\delta_0 > 0$ satisfying the following condition:

$$\forall \mathbf{x} \in U,\ \|\mathbf{x} - \mathbf{x}^0\| < \delta_0 \Longrightarrow D\big(F_N(\mathbf{x}), F_N(\mathbf{x}^0)\big) < \frac{\varepsilon_0}{4}.$$

Let $n_0 \in \mathbb{N}$, so that $1/n_0 < \delta_0$. Thus, $D\big(F_N(\mathbf{x}^{n_0}),\ F_N(\mathbf{x}^0)\big) < \varepsilon_0/4$. Therefore

$$\begin{aligned}\varepsilon_0 &\leq D\big(J(\mathbf{x}^{n_0}),\ J(\mathbf{x}^0)\big)\\ &\leq D\big(J(\mathbf{x}^0),\ F_N(\mathbf{x}^0)\big) + D\big(F_N(\mathbf{x}^0),\ F_N(\mathbf{x}^{n_0})\big) + D\big(F_N(\mathbf{x}^{n_0}),\ J(\mathbf{x}^{n_0})\big)\\ &< \frac{\varepsilon_0}{4} + \frac{\varepsilon_0}{4} + \frac{\varepsilon_0}{4} < \varepsilon_0,\end{aligned}$$

which is contradictory. So J is continuous on $\mathbb{R}^d$. □

Now, we proceed to analyze the properties of four layer regular feedforward FNN's, which are similar with ones of conventional neural networks.

Theorem 4.12 *Let $\sigma : \mathbb{R} \longrightarrow \mathbb{R}$ be a Tauber-Wiener function. Then*
(i) For each $P \in \mathcal{P}$, $\mathcal{H}_0[\sigma]$ is the universal approximator of P.
(ii) If $J : \mathbb{R}^d \longrightarrow \mathcal{F}_0(\mathbb{R})$ is continuous, then both $\mathcal{H}_0[\sigma]$ and $\mathcal{H}[\sigma]$ are the universal approximators of J, respectively.

Proof. (i) Let $P(\mathbf{x}) = \sum_{k=1}^{q} \widetilde{W}_k \cdot P_k(\mathbf{x})$ $(\mathbf{x} \in \mathbb{R}^d)$, and without losing generality to assume not all $|\widetilde{W}_1|, ..., |\widetilde{W}_q|$ are 0. Choose arbitrarily the compact set $U \subset \mathbb{R}^d$, and $\varepsilon > 0$. Since the polynomial P_k $(k = 1, ..., q)$ is continuous on $\mathbb{R}^d$, and σ is a Tauber-Wiener function, there exist $p_k \in \mathbb{N}$, $v'_{1k}, ..., v'_{p_k k}, \theta'_{1k}, ..., \theta'_{p_k k} \in \mathbb{R}$, $\mathbf{u}'_k(1), ..., \mathbf{u}'_k(p_k) \in \mathbb{R}^d$, satisfying $\forall \mathbf{x} = (x_1, ..., x_d) \in U$,

$$\Big|P_k(\mathbf{x}) - \sum_{j=1}^{p_k} v'_{jk} \cdot \sigma\big(\langle \mathbf{x}, \mathbf{u}'_k(j)\rangle + \theta'_{jk}\big)\Big| < \frac{\varepsilon}{2q \cdot \bigvee_{1\le k\le q} \{|\widetilde{W}_k|\}}. \tag{4.44}$$

Let $p = \sum_{k=1}^{q} p_k$. For $k = 2, ..., q$, denote $\beta_k = \sum_{r=1}^{k-1} p_r$, $\beta_1 = 0$. If $k = 1, ..., q$, $j = 1, ..., p$, let

$$v_{jk} = \begin{cases} v'_{(j-\beta_k)k}, & \beta_k < j \le \beta_{k+1}, \\ 0, & \text{otherwise}; \end{cases} \qquad \theta_j = \begin{cases} \theta'_{(j-\beta_k)k}, & \beta_k < j \le \beta_{k+1}, \\ 0, & \text{otherwise}; \end{cases}$$

$$\mathbf{u}(j) = \begin{cases} \mathbf{u}'_k(j - \beta_k), & \beta_k < j \le \beta_{k+1}, \\ 0, & \text{otherwise}. \end{cases}$$

$\forall k \in \{1, ..., q\}$, it is easy to show

$$\sum_{j=1}^{p} v_{jk} \cdot \sigma\big(\langle \mathbf{x}, \mathbf{u}(j)\rangle + \theta_j\big) = \sum_{j=1}^{p_k} v'_{jk} \cdot \sigma\big(\langle \mathbf{x}, \mathbf{u}'_k(j)\rangle + \theta'_{jk}\big). \tag{4.45}$$

Let $G(\mathbf{x}) = \sum_{k=1}^{q} \widetilde{W}_k \cdot \sum_{j=1}^{p} v_{jk} \cdot \sigma\big(\langle \mathbf{x}, \mathbf{u}(j)\rangle + \theta_j\big)$. Then for $\mathbf{x} \in U$, using Lemma 4.7 and Lemma 4.8 we can show

$$\begin{aligned} D\big(G(\mathbf{x}), P(\mathbf{x})\big) &= D\Big(\sum_{k=1}^{q} \widetilde{W}_k \cdot \sum_{j=1}^{p} v_{jk} \cdot \sigma\big(\langle \mathbf{x}, \mathbf{u}(j)\rangle + \theta_j\big), \sum_{k=1}^{q} \widetilde{W}_k \cdot P_k(\mathbf{x})\Big) \\ &\le \sum_{k=1}^{q} |\widetilde{W}_k| \cdot \Big|\sum_{j=1}^{p} v_{jk} \cdot \sigma\big(\langle \mathbf{x}, \mathbf{u}(j)\rangle + \theta_j\big) - P_k(\mathbf{x})\Big| \\ &\le \bigvee_{1\le k\le q} \{|\widetilde{W}_k|\} \sum_{k=1}^{q} \Big|P_k(\mathbf{x}) - \sum_{j=1}^{p_k} v'_{jk} \cdot \sigma\big(\langle \mathbf{x}, \mathbf{u}'_k(j)\rangle + \theta'_{jk}\big)\Big| \\ &< \bigvee_{1\le k\le q} \{|\widetilde{W}_k|\} \cdot q \cdot \frac{\varepsilon}{2q \cdot \bigvee_{1\le k\le q} \{|\widetilde{W}_k|\}} = \frac{\varepsilon}{2}. \end{aligned}$$

Thus, $\forall \mathbf{x} \in U$, $D\big(J(\mathbf{x}), G(\mathbf{x})\big) \le D\big(J(\mathbf{x}), P(\mathbf{x})\big) + D\big(P(\mathbf{x}), G(\mathbf{x})\big) < \varepsilon/2 + \varepsilon/2 = \varepsilon$.

(ii): It is the direct result of Theorem 4.11 and (i). Thus, the theorem is proved. □

Let us now give some equivalent conditions for the universal approximation holding.

Theorem 4.13 *Let $\sigma : \mathbb{R} \longrightarrow \mathbb{R}$ be a continuous Tauber-Wiener function, and $J : \mathbb{R}^d \longrightarrow \mathcal{F}_0(\mathbb{R})$. Then the following propositions are equivalent:*

(i) J is continuous;

(ii) $\mathcal{H}_0[\sigma]$ is the universal approximator of J;

(iii) $\mathcal{H}[\sigma]$ is the universal approximator of J;

(iv) $\mathcal{P}$ is a universal approximator of J.

Proof. By Theorem 4.12, (i)$\Longrightarrow$(ii); And by $\mathcal{H}_0[\sigma] \subset \mathcal{H}[\sigma]$, then (ii)$\Longrightarrow$(iii) is trivial; By the continuity of $\sigma(\cdot)$, it is easy to show, $\forall F_N \in \mathcal{H}[\sigma]$, F_N is continuous. By Lemma 4.8, (iii)$\Longrightarrow$(i); And easily it follows that each function in $\mathcal{P}$ is continuous. So by Theorem 4.11 and Lemma 4.8, (i)$\Longleftrightarrow$(iv). $\square$

4.4.3 An example

In this subsection we illustrate by a real example the realization steps of the approximation process established by Theorem 4.12 and Theorem 4.13. Let the transfer function $\sigma(x) = 1/\big(1+\exp(-x)\big)$. Then σ is a Tauber-Wiener function (see [13–15]). Choose the error bound $\varepsilon = 0.2$, and the dimensionality $d = 1$. At first, we need the following lemma.

Lemma 4.9 *Let $f : \mathbb{R} \longrightarrow \mathbb{R}$ be a continuously differentiable function, and f' be a derived function of f, $C \subset \mathbb{R}$ be a compact set. By $|f'|_C$ we denote the maximum norm of f' on C. For $\widetilde{W}_1, \widetilde{W}_2 \in \mathcal{F}_0(\mathbb{R})$, satisfying* $\text{Supp}(\widetilde{W}_j) \subset C$ *$(j = 1, 2)$. Then $D\big(f(\widetilde{W}_1), f(\widetilde{W}_2)\big) \leq \|f'\|_C \cdot D\big(\widetilde{W}_1, \widetilde{W}_2\big)$.*

Proof. By the definition of the metric $D(\cdot, \cdot)$ it is easy to show

$$D\big(f(\widetilde{W}_1), f(\widetilde{W}_2)\big) = \bigvee_{\alpha\in[0,1)} \big\{d_H\big(f(\widetilde{W}_{1\underset{\bullet}{\alpha}}), f(\widetilde{W}_{2\underset{\bullet}{\alpha}})\big)\big\}$$

$$= \bigvee_{\alpha\in[0,1)} \big\{d_H\big(f(\widetilde{W}_{1\underset{\bullet}{\alpha}}), f(\widetilde{W}_{2\underset{\bullet}{\alpha}})\big)\big\}$$

$$= \bigvee_{\alpha\in[0,1)} \Big\{\max\Big\{ \bigvee_{u\in f(\widetilde{W}_{1\underset{\bullet}{\alpha}})} \bigwedge_{v\in f(\widetilde{W}_{2\underset{\bullet}{\alpha}})} \{|u-v|\}, \bigvee_{u\in f(\widetilde{W}_{2\underset{\bullet}{\alpha}})} \bigwedge_{v\in f(\widetilde{W}_{1\underset{\bullet}{\alpha}})} \{|u-v|\}\Big\}\Big\}$$

$$\leq \bigvee_{\alpha\in[0,1)} \Big\{\max\Big\{ \bigvee_{x\in \widetilde{W}_{1\underset{\bullet}{\alpha}}} \bigwedge_{y\in \widetilde{W}_{2\underset{\bullet}{\alpha}}} \{|f(x)-f(y)|\}, \bigvee_{x\in \widetilde{W}_{2\underset{\bullet}{\alpha}}} \bigwedge_{y\in \widetilde{W}_{1\underset{\bullet}{\alpha}}} \{|f(x)-f(y)|\}\Big\}\Big\}$$

$$\leq \|f'\|_C \cdot \bigvee_{\alpha\in[0,1)} \Big\{\max\Big\{ \bigvee_{x\in \widetilde{W}_{1\underset{\bullet}{\alpha}}} \bigwedge_{y\in \widetilde{W}_{2\underset{\bullet}{\alpha}}} \{|x-y|\}, \bigvee_{x\in \widetilde{W}_{2\underset{\bullet}{\alpha}}} \bigwedge_{y\in \widetilde{W}_{1\underset{\bullet}{\alpha}}} \{|x-y|\}\Big\}\Big\}$$

$$= \|f'\|_C \cdot \bigvee_{\alpha\in[0,1)} \{d_H((\widetilde{W}_1)_{\underset{\bullet}{\alpha}}, (\widetilde{W}_2)_{\underset{\bullet}{\alpha}})\} = \|f'\|_C \cdot D(\widetilde{W}_1, \widetilde{W}_2).$$

The lemma is completed. $\square$

Using the conclusions in [14–16], [47], [79], we can directly construct a three layer feedforward neural network related to a continuous function $f:\mathbb{R}\longrightarrow\mathbb{R}$ to represent f with the approximate sense $\approx_\varepsilon$, that is

Proposition 4.2 *Assume that* $f:[0,1]\longrightarrow\mathbb{R}$ *be a continuous function. There is* $N\in\mathbb{N}$, *and write* $x_j=j/N$ $(j=0,1,...,N)$. *Moreover, for any* $t\in\big[(j-1)/N,\ j/N\big]$, $\big|f(t)-f\big((j-1)/N\big)\big|<\varepsilon/4$, *so that if let*

$$g(t)=f(0)+\sum_{j=1}^{N}\Big(f\Big(\frac{j}{N}\Big)-f\Big(\frac{j-1}{N}\Big)\Big)\sigma\big(K(t-t_j)\big),$$

where K *satisfying:* $K/N>W,\ t>W\Longrightarrow|\sigma(t)-1|<1/N, t<-W\Longrightarrow|\sigma(t)|<1/N$, *and* $t_j=(x_j+x_{j-1})/2$ $(j=1,...,N)$. *Then* $\forall t\in[0,1]$, *we have,* $|f(t)-g(t)|<\varepsilon$.

Proof. For $t\in[0,1]$, let $j\in\{1,...,N\}$, so that $t\in\big[x_{j-1},\ x_j\big]$. Denote

$$h(t)=f(0)+\big(f(x_j)-f(x_{j-1})\big)\cdot\sigma\big(K(t-t_j)\big)+\sum_{i=1}^{j}\big(f(x_i)-f(x_{i-1})\big).$$

By the definition of $g(\cdot)$ and the assumption we can show

$$\begin{aligned}\big|g(t)-h(t)\big| &\le \sum_{i=1}^{j-1}\big|f(x_i)-f(x_{i-1})\big|\cdot\big|\sigma\big(K(t-t_i)\big)-1\big|\\ &\quad+\sum_{i=j+1}^{N}\big|f(x_i)-f(x_{i-1})\big|\cdot\big|\sigma\big(K(t-t_i)\big)\big|\\ &\le \sum_{i=1}^{j-1}\frac{\varepsilon}{4N}+\sum_{i=j}^{N}\frac{\varepsilon}{4N}<\frac{\varepsilon}{2}.\end{aligned}$$

Moreover, easily we have, $h(t)=f(x_{j-1})+\big(f(x_j)-f(x_{j-1})\big)\cdot\sigma\big(K(t-t_j)\big)$. So

$$\begin{aligned}\big|g(t)-f(t)\big| &\le \big|f(t)-h(t)\big|+\big|f(t)-g(t)\big|\\ &< \big|f(t)-f(x_j)\big|+\big|\big(f(x_j)-f(x_{j-1})\big)\cdot\sigma\big(K(t-t_j)\big)\big|+\frac{\varepsilon}{2}\\ &\le \frac{\varepsilon}{2}+\frac{\varepsilon}{4}+\frac{\varepsilon}{4}=\varepsilon.\end{aligned}$$

The proposition is completed. □

Define the fuzzy valued function $S:\mathbb{R}\longrightarrow\mathcal{F}_0(\mathbb{R})$ as follows: $\forall t\in\mathbb{R},\ S(t)=\exp\big\{\cos(\widetilde{W}\cdot t)\big\}$, where $\widetilde{W}\in\mathcal{F}_0(\mathbb{R})$ is a given fuzzy number:

$$\forall x\in\mathbb{R},\ \ \widetilde{W}(x)=\begin{cases}2x, & 0\le x\le\dfrac{1}{2},\\ 2-2x, & \dfrac{1}{2}<x\le 1,\\ 0, & \text{otherwise.}\end{cases}$$

By Lemma 4.9 it is easy to show S is continuous on $\mathbb{R}$. For the error bound $\varepsilon = 0.2$, using the following steps we can construct a four layer feedforward FNN $F_N \in \mathcal{H}_0[\sigma]$ to represent S, approximately.

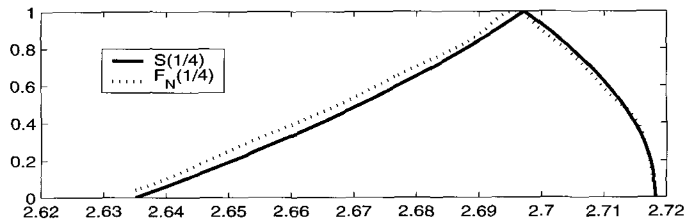


Figure 4.16 Fuzzy set $S(1/4)$ and fuzzy set $F_N(1/4)$

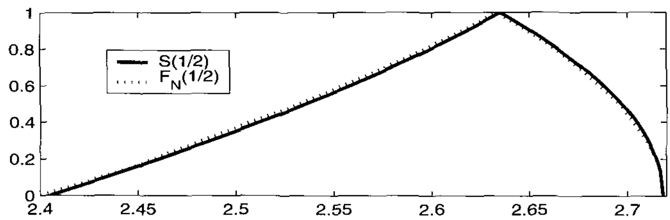


Figure 4.17 Fuzzy set $S(1/2)$ and fuzzy set $F_N(1/2)$

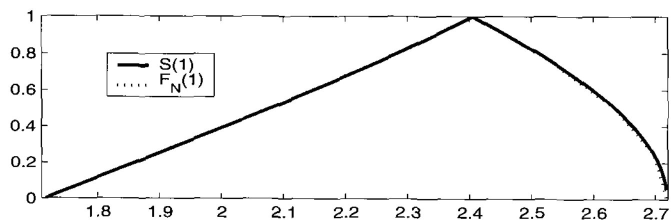


Figure 4.18 Fuzzy set $S(1)$ and fuzzy set $F_N(1)$

Step 1. Determine the fuzzy valued polynomial $P(\cdot)$ with respect to S. Obviously, $|\widetilde{W}| \leq 1$, and using Lemma 4.9, we have, $f(t) = \exp(\cos(t))$. $\forall t_1, t_2 \in [0,1]$, it is easy to show, $\text{Supp}(\widetilde{W} \cdot t_i) \subset [0,1]$ $(i = 1,2)$. Then

$$\begin{aligned} D\big(S(t_1),\, S(t_2)\big) &= D\big(f(\widetilde{W} \cdot t_1),\, f(\widetilde{W} \cdot t_2)\big) \leq \big\|f'\big\|_{[0,1]} \cdot D(\widetilde{W} \cdot t_1,\, \widetilde{W} \cdot t_2) \\ &\leq \bigvee_{t \in [0,1]} \big\{\sin(t) \exp\{\cos(t)\}\big\} \cdot |\widetilde{W}| \cdot |t_1 - t_2| < 2 \cdot |t_1 - t_2| \end{aligned}$$

Let $q \in \mathbb{N}$. Then $D\Big(S\Big(\dfrac{i}{q}\Big),\, S\Big(\dfrac{i-1}{q}\Big)\Big) \leq \dfrac{2}{q}$ $(i = 1, ..., q)$. By Theorem 4.11, it follows that

$$M \triangleq \bigvee_{t_1,t_2\in[0,1]} \{D(S(t_1),\ S(t_2))\}\leq 2,\ \ \delta=0.025.$$

When $q \geq M/(\varepsilon\cdot\delta^2)$, i.e. if $q \geq 1.6\times 10^4$, then $D\big(B_q(S;t),\ S(t)\big)< \varepsilon/2$ for each $t\in[0,\ 1]$. For simplicity, we proceed to construct our approximate process by choosing $q=40$. Hence

$$B_q(S;t)=\sum_{k=0}^{40}\binom{40}{k}t^k(1-t)^{40-k}\cdot\exp\Big(\cos\Big(\widetilde{W}\cdot\frac{k}{40}\Big)\Big).$$

Step 2. For $k=0,1,...,40$, construct the three layer feedforward network N_k. Let $P_k(t)=\binom{40}{k}t^k(1-t)^{40-k}$, then

$$W_k \triangleq \exp\Big(\cos\Big(\widetilde{W}\cdot\frac{k}{40}\Big)\Big), \Longrightarrow |\ \widetilde{W}_k\ |\leq 2.712.$$

By Proposition 4.2 we construct the neural network as

$$N_k(t)=P_k(0)+\sum_{j=1}^{p_k}\Big(P_k\Big(\frac{j}{p_k}\Big)-P_k\Big(\frac{j-1}{p_k}\Big)\Big)\cdot\sigma\Big(K_k\Big(t-\frac{j-1}{p_k}\Big)\Big), \quad (4.46)$$

where K_j, p_k satisfy the following conditions:

$$\begin{cases} |t_1-t_2|<\dfrac{1}{p_k}, \Longrightarrow |P_k(t_1)-P_k(t_2)|<\dfrac{\varepsilon}{2q\cdot\max\limits_{1\leq k\leq q}\{|\ \widetilde{W}_k\ |\}}; \\ \dfrac{K_k}{p_k}\geq W: |\sigma(t)-1|<\dfrac{1}{p_k}(t>W),\ |\sigma(t)|<\dfrac{1}{p_k}(t<-W). \end{cases} \quad (4.47)$$

It is easy to show, $|P_k'|_{[0,1]}\leq 4$ $(k=0,1,...,40)$. So using (4.47) and Lemma 4.9, we can choose

$$p_k=440,\ K_k=2630\ (k=0,1,...,40);\ W=6.2.$$

Step 3. Construct a four layer regular feedforward FNN as $F_N:\mathbb{R}\longrightarrow\mathcal{F}_0(\mathbb{R})$. By Theorem 4.12 and (4.46). Let

$$F_N(t)=\sum_{k=0}^{q}\widetilde{W}_k\cdot\sum_{j=1}^{p}v_{jk}\cdot\sigma(u_j\cdot t+\theta_j).$$

Then $\forall t\in[0,1]$, $D(S(t),\ F_N(t))<\varepsilon$. Using the fact: $P_0(0)=1=2\sigma(0)$, $\widetilde{W}_0=\exp\big(\cos(\widetilde{W}\cdot 0)\big)=\exp(1)$, $P_k(0)=0$ $(k=1,...,40)$, we have

$$F_N(t)=\Big[2\sigma(0)+\sum_{j=1}^{440}\sigma\Big(2630t-\frac{263(j-1)}{44}\Big)\Big(\Big(1-\frac{j}{440}\Big)^{40}-\Big(1-\frac{j-1}{440}\Big)^{40}\Big)\Big]\exp(1)$$
$$+\sum_{k=1}^{40}\exp\Big(\cos\Big(\widetilde{W}\cdot\frac{k}{40}\Big)\Big)\cdot\sum_{j=1}^{440}\Big(P_k\Big(\frac{j}{440}\Big)-P_k\Big(\frac{j-1}{440}\Big)\Big)\sigma\Big(2630t-\frac{263(j-1)}{44}\Big).$$

As shown in Figure 4.16, Figure 4.17 and Figure 4.18, they illustrate the values of the original fuzzy valued function $S(\cdot)$ and its approximate fuzzy valued function $F_N(\cdot)$ determined by the FNN, at $t = 1/4$, $t = 1/2$ and $t = 1$, respectively, that is, the membership curves of the fuzzy sets $S(1/4)$ and $F_N(1/4)$, $S(1/2)$ and $F_N(1/2)$, $S(1)$ and $F_N(1)$, respectively.

By comparison among Figures 4.16, 4.17 and 4.18, we can find that even choosing the polynomial whose order is much lower than ones needed in Theorem 4.11, we can construct a four layer regular feedforward FNN $F_N(\cdot)$ to approximate the fuzzy valued function $S(\cdot)$ with the given accuracy.

§4.5 Approximation analysis of regular FNN

In previous section, we show that a four layer regular feedforward FNN with real inputs is the universal approximator of continuous fuzzy valued functions. If the input signals are fuzzy sets, whether do the regular FNN's possess the same property for the fuzzy functions defined on a collection of fuzzy sets? Buckley and Hayashi [5] firstly study such a topic, and show that the continuity of fuzzy functions is not a sufficient condition for ensuring the universal approximation of regular FNN's. In addition to the continuity, we must introduce other conditions for fuzzy functions. To this end, we shall restrict our discussions on the bounded fuzzy number collection $\mathcal{F}_c(\mathbb{R})$.

4.5.1 Closure fuzzy mapping

Definition 4.3 Let $F : \mathcal{F}_c(\mathbb{R})^d \longrightarrow \mathcal{F}_c(\mathbb{R})$, and following conditions hold:

(i) Let $I_1, ..., I_d \subset \mathbb{R}$ be index sets, $\{(\widetilde{A}_{1k_1}, ..., \widetilde{A}_{dk_d}) | k_1 \in I_1, ..., k_d \in I_d\} \subset \mathcal{F}_c(\mathbb{R})^d$ be a pre-compact set, i.e. $\overline{\{(\widetilde{A}_{1k_1}, ..., \widetilde{A}_{dk_d}) | k_1 \in I_1, ..., k_d \in I_d\}}$ is a compact set, then if $\forall i \in \{1, ..., d\}$, $\bigcup_{k \in I_i} \widetilde{A}_{ik} \in \mathcal{F}_c(\mathbb{R})$, the following fact holds:

$$\forall \alpha \in [0,1),\ \overline{F\Big(\bigcup_{k\in I_1} \widetilde{A}_{1k}, ..., \bigcup_{k\in I_d} \widetilde{A}_{dk}\Big)_{\underset{\bullet}{\alpha}}} = \overline{\bigcup_{k_1\in I_1,...,k_d\in I_d} F\big(\widetilde{A}_{1k_1}, ..., \widetilde{A}_{dk_d}\big)_{\underset{\bullet}{\alpha}}};$$

(ii) $\forall \alpha \in [0,1), \forall (\widetilde{A}_1, ..., \widetilde{A}_d) \in \mathcal{F}_c(\mathbb{R})^d$, we have

$$\overline{F\big(\widetilde{A}_1, ..., \widetilde{A}_d\big)_{\underset{\bullet}{\alpha}}} = \overline{F\big((\widetilde{A}_1)_{\underset{\bullet}{\alpha}}, ..., (\widetilde{A}_d)_{\underset{\bullet}{\alpha}}\big)_{\underset{\bullet}{\alpha}}}.$$

We call F a closure fuzzy mapping.

Now we give an example of closure fuzzy mapping.

Proposition 4.3 *Let the function $f : \mathbb{R}^d \longrightarrow \mathbb{R}$ be continuous. Then the extended function of f to $\mathcal{F}_c(\mathbb{R})^d$, also denoted by f is a closure fuzzy mapping.*

Proof. At first define the extended function $f : \mathcal{F}_c(\mathbb{R})^d \longrightarrow \mathcal{F}_c(\mathbb{R})$. $\forall \alpha \in [0,1)$, $\widetilde{\mathbf{X}} = (\widetilde{X}_1, ..., \widetilde{X}_d) \in \mathcal{F}_c(\mathbb{R})^d$, $f(\widetilde{X}_1, ..., \widetilde{X}_d)_{\underset{\bullet}{\alpha}} = f\big((\widetilde{X}_1)_{\underset{\bullet}{\alpha}}, ..., (\widetilde{X}_d)_{\underset{\bullet}{\alpha}}\big) \subset \mathbb{R}$.

So we conclude that

$$\overline{f(\widetilde{X}_1,...,\widetilde{X}_d)_{\underset{\bullet}{\alpha}}} = \overline{f((\widetilde{X}_1)_{\underset{\bullet}{\alpha}},...,(\widetilde{X}_d)_{\underset{\bullet}{\alpha}})_{\underset{\bullet}{\alpha}}}.$$

For any index sets $I_1,...,I_d \subset \mathbb{R}$, and for each pre-compact set of $\mathcal{F}_c(\mathbb{R})^d$: $\{(\widetilde{A}_{1k_1},...,\widetilde{A}_{dk_d})|k_1 \in I_1,...,k_d \in I_d\}$, if $\forall i = 1,...,d,\ \bigcup_{k\in I_i} \widetilde{A}_{ik} \in \mathcal{F}_c(\mathbb{R})$, then for any $y \in \mathbb{R}$, it is easy to show the following fact:

$$f\Big(\bigcup_{k\in I_1} \widetilde{A}_{1k},..., \bigcup_{k\in I_d} \widetilde{A}_{dk}\Big)(y) = \bigvee_{f(x_1,...,x_d)=y} \Big\{\bigwedge_{i=1}^{d}\Big(\bigcup_{k\in I_i} \widetilde{A}_{ik}\Big)(x_i)\Big\}$$

$$= \bigvee_{f(x_1,...,x_d)=y} \Big\{\bigwedge_{i=1}^{d}\Big(\bigvee_{k\in I_i} \widetilde{A}_{ik}(x_i)\Big)\Big\} = \bigvee_{k_1\in I_1,...,k_d\in I_d}\Big(\bigvee_{f(x_1,...,x_d)=y}\Big\{\bigwedge_{i=1}^{d} \widetilde{A}_{ik_i}(x_i)\Big\}\Big)$$

$$= \bigvee_{k_1\in I_1,...,k_d\in I_d} \{f(\widetilde{A}_{1k_1},...,\widetilde{A}_{dk_d})(y)\} = \Big(\bigcup_{k_1\in I_1,...,k_d\in I_d} f(\widetilde{A}_{1k_1},...,\widetilde{A}_{dk_d})\Big)(y).$$

Therefore, $f\big(\bigcup_{k\in I_1} \widetilde{A}_{1k},..., \bigcup_{k\in I_d} \widetilde{A}_{dk}\big) = \bigcup_{k_1\in I_1,...,k_d\in I_d} f(\widetilde{A}_{1k_1},...,\widetilde{A}_{dk_d})$. Thus, $\forall \alpha \in [0,1)$, we have

$$\begin{aligned}\overline{f\Big(\bigcup_{k\in I_1} \widetilde{A}_{1k},..., \bigcup_{k\in I_d} \widetilde{A}_{dk}\Big)_{\underset{\bullet}{\alpha}}} &= \overline{\bigcup_{k_1\in I_1,...,k_d\in I_d} f(\widetilde{A}_{1k_1},...,\widetilde{A}_{dk_d})_{\underset{\bullet}{\alpha}}} \\ &= \overline{\bigcup_{k_1\in I_1,...,k_d\in I_d} f\big((\widetilde{A}_{1k_1})_{\underset{\bullet}{\alpha}},...,(\widetilde{A}_{dk_d})_{\underset{\bullet}{\alpha}}\big)_{\underset{\bullet}{\alpha}}}.\end{aligned}$$

So f is a closure fuzzy mapping. □

By Proposition 4.3, we can imply, $\forall P \in \mathcal{P},\ P$ is a closure fuzzy mapping.

Let $\mathcal{A}$ be a subclass of all fuzzy functions $\mathcal{F}_c(\mathbb{R})^d \longrightarrow \mathcal{F}_c(\mathbb{R})$, and F : $\mathcal{F}_c(\mathbb{R})^d \longrightarrow \mathcal{F}_c(\mathbb{R})$. Similarly with Definition 4.2, we can define $\mathcal{A}$ as a universal approximator of F.

Lemma 4.10 *Let I be an arbitrary index set, and A, B, A_i, $B_i \subset \mathbb{R}$ $(i \in I)$ be nonempty, $h > 0$. Then*

(i) $d_H(A,B) = d_H(\overline{A},\overline{B})$;

(ii) $\forall i \in I,\ d_H(A_i,B_i) \le h \Longrightarrow d_H\Big(\overline{\bigcup_{i\in I} A_i},\ \overline{\bigcup_{i\in I} B_i}\Big) \le h.$

Proof. (i) By (1.3) easily we imply, $d_H(A,\overline{A}) = d_H(B,\overline{B}) = 0$. Hence

$$d_H(A,B) \le d_H(A,\overline{A}) + d_H(\overline{A},\overline{B}) + d_H(\overline{B},B) = d_H(\overline{A},\overline{B}).$$

Similarly we have, $d_H(\overline{A},\overline{B}) \le d_H(A,B)$. Thus, $d_H(A,B) = d_H(\overline{A},\overline{B})$. (i) holds.

(ii) Let $W = \bigcup_{i\in I} A_i,\ V = \bigcup_{i\in I} B_i$. By the assumption and (1.3) it follows that

$$\forall i \in I, \bigvee_{x\in A_i}\bigwedge_{y\in B_i}\{|x-y|\} \le h,\ \bigvee_{y\in B_i}\bigwedge_{x\in A_i}\{|x-y|\} \le h. \tag{4.48}$$

$\forall x_0 \in W = \bigcup_{i\in I} A_i$, there is $i_0 \in I$, satisfying $x_0 \in A_{i_0}$. By (4.48) we obtain

$$\bigwedge_{y\in V}\{|x_0-y|\} \le \bigwedge_{y\in B_{i_0}}\{|x_0-y|\} \le h.$$

So $\bigvee_{x\in W}\bigwedge_{y\in V}\{|x-y|\} \le h$. Similarly we can show, $\bigvee_{y\in V}\bigwedge_{x\in W}\{|x-y|\} \le h$. Therefore, $d_H(W,V) \le h$. Hence by (i) we have, $d_H\Big(\overline{\bigcup_{i\in I} A_i},\ \overline{\bigcup_{i\in I} B_i}\Big) = d_H(W,V) \le h$.
□

Using Lemma 4.10 we can get a useful property about the universal approximators.

Theorem 4.14 *Let for each $G \in \mathcal{A}$, G be a closure fuzzy mapping, and $\mathcal{A}$ be the universal approximator of $F : \mathcal{F}_c(\mathbb{R})^d \longrightarrow \mathcal{F}_c(\mathbb{R})$. Then also F is a closure fuzzy mapping.*

Proof. We apply reduction to absurdity to show the conditions (i) (ii) in Definition 4.3 hold for F. Ar first we show (i).

If condition (i) in Definition 4.3 does not hold for F, there exist index sets $I_1^0, ..., I_d^0 \subset \mathbb{R}$ and $\{(\widetilde{A}_{1k_1}, ..., \widetilde{A}_{dk_d})|k_1 \in I_1^0, ..., k_d \in I_d^0\} \subset \mathcal{F}(\mathbb{R})^d$, a pre-compact set family, so that $\forall i \in \{1, ..., d\}$, $\bigcup_{k\in I_i^0} \widetilde{A}_{ik} \in \mathcal{F}_c(\mathbb{R})$, and there is $\alpha \in [0,1)$, satisfying

$$\overline{F\Big(\bigcup_{k\in I_1^0} \widetilde{A}_{1k}, ..., \bigcup_{k\in I_d^0} \widetilde{A}_{dk}\Big)_{\underset{\bullet}{\alpha}}} \ne \overline{\bigcup_{k_1\in I_1^0,...,k_d\in I_d^0} F\big(\widetilde{A}_{1k_1}, ..., \widetilde{A}_{dk_d}\big)_{\underset{\bullet}{\alpha}}}.$$

Choose

$$\varepsilon_0 = \frac{1}{2} d_H\Bigg(\overline{F\Big(\bigcup_{k\in I_1^0} \widetilde{A}_{1k}, ..., \bigcup_{k\in I_d^0} \widetilde{A}_{dk}\Big)_{\underset{\bullet}{\alpha}}},\ \overline{\bigcup_{k_1\in I_1^0,...,k_d\in I_d^0} F(\widetilde{A}_{1k_1}, ..., \widetilde{A}_{dk_d})_{\underset{\bullet}{\alpha}}}\Bigg),$$

$$\mathcal{U} = \Big\{(\widetilde{A}_{1k_1}, ..., \widetilde{A}_{dk_d})|k_1 \in I_1^0, ..., k_d \in I_d^0\Big\} \bigcup \Big\{\Big(\bigcup_{k\in I_1^0} \widetilde{A}_{1k}, ..., \bigcup_{k\in I_d^0} \widetilde{A}_{dk}\Big)\Big\}.$$

Obviously, $\varepsilon_0 > 0$, and $\mathcal{U} \subset \mathcal{F}(\mathbb{R})^d$ is a pre-compact set. By assumption, there is $G \in \mathcal{A}:\ \forall\, \widetilde{\mathbf{X}} \in \mathcal{U},\ D\big(F(\widetilde{\mathbf{X}}),\ G(\widetilde{\mathbf{X}})\big) < \varepsilon_0/2$. So using (1.6) we imply

$$\forall k_1 \in I_1^0, ..., k_d \in I_d^0,\ d_H\big(F(\widetilde{A}_{1k_1}, ..., \widetilde{A}_{dk_d})_{\underset{\bullet}{\alpha}},\ G(\widetilde{A}_{1k_1}, ..., \widetilde{A}_{dk_d})_{\underset{\bullet}{\alpha}}\big) < \frac{\varepsilon_0}{2},$$

Thus, using Lemma 4.10 we obtain the following facts:

$$\begin{cases} d_H\Big(\overline{\bigcup_{k_1\in I_1^0,...,k_d\in I_d^0} F(\widetilde{A}_{1k_1},...,\widetilde{A}_{dk_d})_{\underset{\bullet}{\alpha}}},\ \overline{\bigcup_{k_1\in I_1^0,...,k_d\in I_d^0} G(\widetilde{A}_{1k_1},...,\widetilde{A}_{dk_d})_{\underset{\bullet}{\alpha}}}\Big)\leq \dfrac{\varepsilon_0}{2}, \\ d_H\Big(\overline{F\Big(\bigcup_{k\in I_1^0}\widetilde{A}_{1k},...,\bigcup_{k\in I_d^0}\widetilde{A}_{dk}\Big)_{\underset{\bullet}{\alpha}}},\ \overline{G\Big(\bigcup_{k\in I_1^0}\widetilde{A}_{1k},...,\bigcup_{k\in I_d^0}\widetilde{A}_{dk}\Big)_{\underset{\bullet}{\alpha}}}\Big)\leq \dfrac{\varepsilon}{2}. \end{cases} \tag{4.49}$$

By assumption G is a closure fuzzy mapping, so we have

$$\overline{G\Big(\bigcup_{k\in I_1^0}\widetilde{A}_{1k},...,\bigcup_{k\in I_d^0}\widetilde{A}_{dk}\Big)_{\underset{\bullet}{\alpha}}} = \overline{\bigcup_{k_1\in I_1^0,...,k_d\in I_d^0} G(\widetilde{A}_{1k_1},...,\widetilde{A}_{dk_d})_{\underset{\bullet}{\alpha}}}.$$

By the triangle inequality for d_H and (4.49) it follows that

$$\begin{aligned} \frac{3}{2}\,\varepsilon_0 &< d_H\Big(\overline{F\Big(\bigcup_{k\in I_1^0}\widetilde{A}_{1k},...,\bigcup_{k\in I_d^0}\widetilde{A}_{dk}\Big)_{\underset{\bullet}{\alpha}}},\ \overline{\bigcup_{k_1\in I_1^0,...,k_d\in I_d^0} F(\widetilde{A}_{1k_1},...,\widetilde{A}_{dk_d})_{\underset{\bullet}{\alpha}}}\Big) \\ &\leq d_H\Big(\overline{F\Big(\bigcup_{k\in I_1^0}\widetilde{A}_{1k},...,\bigcup_{k\in I_d^0}\widetilde{A}_{dk}\Big)_{\underset{\bullet}{\alpha}}},\ \overline{G\Big(\bigcup_{k\in I_1^0}\widetilde{A}_{1k},...,\bigcup_{k\in I_d^0}\widetilde{A}_{dk}\Big)_{\underset{\bullet}{\alpha}}}\Big) \\ &\quad + d_H\Big(\overline{G\Big(\bigcup_{k\in I_1^0}\widetilde{A}_{1k},...,\bigcup_{k\in I_d^0}\widetilde{A}_{dk}\Big)_{\underset{\bullet}{\alpha}}},\ \overline{\bigcup_{k_1\in I_1^0,...,k_d\in I_d^0} G(\widetilde{A}_{1k_1},...,\widetilde{A}_{dk_d})_{\underset{\bullet}{\alpha}}}\Big), \\ &\quad + d_H\Big(\overline{\bigcup_{k_1\in I_1^0,...,k_d\in I_d^0} G(\widetilde{A}_{1k_1},...,\widetilde{A}_{dk_d})_{\underset{\bullet}{\alpha}}},\ \overline{\bigcup_{k_1\in I_1^0,...,k_d\in I_n^0} F(\widetilde{A}_{1k_1},...,\widetilde{A}_{dk_d})_{\underset{\bullet}{\alpha}}}\Big), \\ &\leq \frac{\varepsilon_0}{2}+\frac{\varepsilon_0}{2}=\varepsilon_0, \end{aligned}$$

Which is contradictory. So (i) in Definition 4.3 holds. With the same reason, we can show (ii) in Definition 4.3. Thus, F is a closure fuzzy mapping. □

Theorem 4.15 *Let $F,\ G:\mathcal{F}_c(\mathbb{R})^d \longrightarrow \mathcal{F}_c(\mathbb{R})$ be a closure fuzzy mapping, and $U_1,...,U_d\subset\mathbb{R}$ be compact sets. Let for any $\mathbf{x}=(x_1,...,x_d)\in U_1\times\cdots\times U_d$, $D\big(F(\mathbf{x}),\ G(\mathbf{x})\big)\leq h$. Then*

(i) For each $\mathbf{X}=(X_1,...,X_d)$, where $X_i\subset U_i$ $(i=1,...,d)$ is a convex set, we have $D\big(F(\mathbf{X}),\ G(\mathbf{X})\big)\leq h$;

(ii) For each $\widetilde{\mathbf{X}}=(\widetilde{X}_1,...,\widetilde{X}_d)\in\mathcal{F}_c(\mathbb{R})^d$, and $\mathrm{Supp}(\widetilde{X}_i)\subset U_i$ $(i=1,...,d)$, it follows that $D\big(F(\widetilde{\mathbf{X}}),\ G(\widetilde{\mathbf{X}})\big)\leq h$.

Proof. (i) At first by [19, 20] we have, $\big\{\{x\}|x\in X_i\big\}\subset\mathcal{F}_c(\mathbb{R})$ $(i=1,...,d)$ is a pre-compact set. Moreover by assumption, $\forall\mathbf{x}=(x_1,...,x_d):x_i\in X_i$ $(i=1,...,d)$, it follows that $\mathbf{x}\in U_1\times\cdots\times U_d$. So $D\big(F(\mathbf{x}),\ G(\mathbf{x})\big)\leq h$. Thus, $\forall\alpha\in[0,1)$, $\mathbf{x}=(x_1,...,x_d):x_i\in X_i$ $(i=1,...,d)$, $d_H\big(F(\mathbf{x})_{\underset{\bullet}{\alpha}},\ G(\mathbf{x})_{\underset{\bullet}{\alpha}}\big)\leq h$.

Therefore, using assumption and Lemma 4.10 we imply

$$
\begin{aligned}
d_H\big(F\,(\mathbf{X})_{\underset{\bullet}{\alpha}},\, G(\mathbf{X})_{\underset{\bullet}{\alpha}}\big) &= d_H\big(\overline{F(\mathbf{X})_{\underset{\bullet}{\alpha}}},\, \overline{G(\mathbf{X})_{\underset{\bullet}{\alpha}}}\big)\\
&= d_H\Big(\overline{F\Big(\bigcup_{x\in X_1}\{x\},\dots,\bigcup_{x\in X_d}\{x\}\Big)_{\underset{\bullet}{\alpha}}},\, \overline{G\Big(\bigcup_{x\in X_1}\{x\},\dots,\bigcup_{x\in X_d}\{x\}\Big)_{\underset{\bullet}{\alpha}}}\Big)\\
&= d_H\Big(\overline{\bigcup_{x_1\in X_1,\dots,x_d\in X_d} F(x_1,\dots,x_d)_{\underset{\bullet}{\alpha}}},\, \overline{\bigcup_{x_1\in X_1,\dots,x_d\in X_d} G(x_1,\dots,x_d)_{\underset{\bullet}{\alpha}}}\Big)\le h.
\end{aligned}
$$

Thus, $D\big(F(\mathbf{X}),\, G(\mathbf{X})\big)= \bigvee_{\alpha\in[0,1)}\big\{d_H\big(F(\mathbf{X})_{\underset{\bullet}{\alpha}},\, G(\mathbf{X})_{\underset{\bullet}{\alpha}}\big)\big\}\le h$. (i) holds.

(ii) Since F, G are closure fuzzy mapping, $\forall\alpha\in[0,1)$, we have

$$
\overline{F(\widetilde{\mathbf{X}})_{\underset{\bullet}{\alpha}}} = \overline{F\big((\widetilde{X}_1)_{\underset{\bullet}{\alpha}},\dots,(\widetilde{X}_d)_{\underset{\bullet}{\alpha}}\big)_{\underset{\bullet}{\alpha}}},\quad \overline{G(\widetilde{\mathbf{X}})_{\underset{\bullet}{\alpha}}} = \overline{G\big((\widetilde{X}_1)_{\underset{\bullet}{\alpha}},\dots,(\widetilde{X}_d)_{\underset{\bullet}{\alpha}}\big)_{\underset{\bullet}{\alpha}}}.
$$

Thus, by (i) and Lemma 4.10 we obtain

$$
\begin{aligned}
d_H\big(F(\widetilde{\mathbf{X}})_{\underset{\bullet}{\alpha}},\ G(\widetilde{\mathbf{X}})_{\underset{\bullet}{\alpha}}\big) &= d_H\big(\overline{F(\widetilde{\mathbf{X}})_{\underset{\bullet}{\alpha}}},\, \overline{G(\widetilde{\mathbf{X}})_{\underset{\bullet}{\alpha}}}\big)\\
&= d_H\big(\overline{F\big((\widetilde{X}_1)_{\underset{\bullet}{\alpha}},\dots,(\widetilde{X}_d)_{\underset{\bullet}{\alpha}}\big)_{\underset{\bullet}{\alpha}}},\, \overline{G\big((\widetilde{X}_1)_{\underset{\bullet}{\alpha}},\dots,(\widetilde{X}_d)_{\underset{\bullet}{\alpha}}\big)_{\underset{\bullet}{\alpha}}}\big)\\
&= d_H\big(F\big((\widetilde{X}_1)_{\underset{\bullet}{\alpha}},\dots,(\widetilde{X}_d)_{\underset{\bullet}{\alpha}}\big)_{\underset{\bullet}{\alpha}},\, G\big((\widetilde{X}_1)_{\underset{\bullet}{\alpha}},\dots,(\widetilde{X}_d)_{\underset{\bullet}{\alpha}}\big)_{\underset{\bullet}{\alpha}}\big)\\
&\le D\big(F\big((\widetilde{X}_1)_{\underset{\bullet}{\alpha}},\dots,(\widetilde{X}_d)_{\underset{\bullet}{\alpha}}\big), G\big((\widetilde{X}_1)_{\underset{\bullet}{\alpha}},\dots,(\widetilde{X}_d)_{\underset{\bullet}{\alpha}}\big)\big)\le h.
\end{aligned}
$$

Therefore, $D\big(F(\widetilde{\mathbf{X}}), G(\widetilde{\mathbf{X}})\big)= \bigvee_{\alpha\in[0,1)}\big\{d_H\big(F(\widetilde{\mathbf{X}})_{\underset{\bullet}{\alpha}},\, G(\widetilde{\mathbf{X}})_{\underset{\bullet}{\alpha}}\big)\big\}\le h$. □

Let us now aim at the universal approximation of $\mathcal{P}$ to continuously closure fuzzy mappings.

Theorem 4.16 *Let $F:\mathcal{F}_c(\mathbb{R})^d \longrightarrow \mathcal{F}_c(\mathbb{R})$ be a continuous fuzzy function. Then the universal approximation of $\mathcal{P}$ holds for F if and only if F is a closure fuzzy mapping.*

Proof. Necessity is a direct result of Theorem 4.14 and the fact: $\forall P\in\mathcal{P}$, P is a closure fuzzy mapping. So it suffices to prove sufficiency. By assumption, if F is restricted to $\mathbb{R}^d$, then F is continuous fuzzy valued function. For each $\varepsilon>0$ and compact set $\mathcal{U}\subset\mathcal{F}(\mathbb{R})^d$, and let $U=U_1\times\cdots\times U_d\subset\mathbb{R}^d$ be a compact set corresponding to $\mathcal{U}$. By Remark 4.3, there is a fuzzy valued polynomial $P_0:\ P_0(\mathbf{x})=\sum\limits_{i=1}^{m}P_i(\mathbf{x})\cdot\widetilde{A}_i\ \ (\mathbf{x}\in\mathbb{R}^d,\ \widetilde{A}_i\in\mathcal{F}_c(\mathbb{R}))$, so that $\forall\mathbf{x}\in U,\ D\big(F(\mathbf{x}),\, P_0(\mathbf{x})\big)<\varepsilon/2$. Define fuzzy polynomial P as follows:

$$
\forall\,\widetilde{\mathbf{X}}=(\widetilde{X}_1,\dots,\widetilde{X}_d)\in\mathcal{F}_c(\mathbb{R})^d,\ P(\widetilde{\mathbf{X}})=\sum_{i=1}^{m}\widetilde{A}_i\cdot P_i(\widetilde{\mathbf{X}}).
$$

Considering $\widetilde{\mathbf{X}}=(\widetilde{X}_1,...,\widetilde{X}_d)\in\mathcal{U}$, we obtain $\mathrm{Supp}(\widetilde{X}_i)\subset U_i$ $(i=1,...,d)$. Then using Theorem 4.15 $\forall\,\widetilde{\mathbf{X}}\in\mathcal{U}$, we imply that $D\big(F(\widetilde{\mathbf{X}}),\ P(\widetilde{\mathbf{X}})\big)\leq\varepsilon/2<\varepsilon$. that is, the universal approximation of $\mathcal{P}$ to F holds. □

In the following we extend the inputs of the regular FNN defined by (4.42) (4.43) as the fuzzy sets in $\mathcal{F}_c(\mathbb{R})$. We obtain the general four layer regular feedforward FNN, i.e. let

$$\widetilde{\mathcal{H}}[\sigma]=\Big\{\widetilde{F}_N\,\Big|\widetilde{F}_N(\widetilde{\mathbf{X}})=\sum_{k=1}^{q}\widetilde{W}_k\cdot\Big(\sum_{j=1}^{p}\widetilde{V}_{jk}\cdot\sigma\big(\langle\widetilde{\mathbf{X}},\ \widetilde{\mathbf{U}}(j)\rangle+\widetilde{\Theta}_j\big)\Big)$$
$$\text{for }\widetilde{\mathbf{X}}\in\mathcal{F}_c(\mathbb{R})^d,\ p,\ q\in\mathbb{N},\ \widetilde{W}_k,\ \widetilde{V}_{jk},\ \widetilde{\Theta}_j\in\mathcal{F}_c(\mathbb{R}),\ \widetilde{\mathbf{U}}(j)\in\mathcal{F}_c(\mathbb{R})^d\Big\}. \tag{4.50}$$

Specifically, if let $\widetilde{V}_{jk},\ \widetilde{\Theta}_j$ be $v_{jk},\ \theta_j\in\mathbb{R}$, respectively, and $\widetilde{\mathbf{U}}(j)\in\mathcal{F}_c(\mathbb{R})^d$ be a vector $\mathbf{u}_k(j)\in\mathbb{R}^d$, then we obtain a subset $\widetilde{\mathcal{H}}_0[\sigma]$ of $\widetilde{\mathcal{H}}[\sigma]$:

$$\widetilde{\mathcal{H}}_0[\sigma]=\Big\{\widetilde{F}_N\,\Big|\widetilde{F}_N(\widetilde{\mathbf{X}})=\sum_{k=1}^{q}\widetilde{W}_k\cdot\Big(\sum_{j=1}^{p}v_{jk}\cdot\sigma\big(\langle\mathbf{u}(j),\ \widetilde{\mathbf{X}}\rangle+\theta_j\big)\Big)$$
$$\text{for }\widetilde{\mathbf{X}}\in\mathcal{F}_c(\mathbb{R})^d,\ p,\ q\in\mathbb{N},\ \widetilde{W}_k\in\mathcal{F}_c(\mathbb{R}),\ v_{jk},\ \theta_j\in\mathbb{R},\ \mathbf{u}(j)\in\mathbb{R}^d\Big\}. \tag{4.51}$$

Remark 4.4 By Proposition 4.3 easily we have, if $\widetilde{F}_N\in\widetilde{\mathcal{H}}[\sigma]$, then $\widetilde{F}_N$ is a closure fuzzy mapping.

We proceed to discuss the universal approximation. The equivalent conditions of continuous fuzzy functions which can be represented with the sense "$\approx_\varepsilon$" by the functions in $\widetilde{\mathcal{H}}[\sigma]$ or $\widetilde{\mathcal{H}}_0[\sigma]$, can be established.

Theorem 4.17 *Let σ be a Tauber-Wiener function, and $F:\mathcal{F}_c(\mathbb{R})^d\longrightarrow\mathcal{F}_c(\mathbb{R})$ be a continuously closure fuzzy mapping. Then $\widetilde{\mathcal{H}}_0[\sigma]$ is the universal approximator of F.*

Proof. For arbitrarily $\varepsilon>0$, and compact set $\mathcal{U}\subset\mathcal{F}(\mathbb{R})^d$, suppose $\mathbf{U}=U_1\times\cdots\times U_d\subset\mathbb{R}^d$ is a compact set corresponding to $\mathcal{U}$. Obviously, F is continuous if it is restricted to $\mathbb{R}^d$. By Theorem 4.11, there are $p,\ q\in\mathbb{N},\ v_{jk},\ \theta_j\in\mathbb{R},\ \widetilde{W}_k\in\mathcal{F}_c(\mathbb{R})$ and $\mathbf{u}(j)\in\mathbb{R}^d$ $(k=1,...,q,\ j=1,...,p)$, such that

$$\forall\mathbf{x}\in\mathbf{U},\ D\big(F(\mathbf{x}),\ \sum_{k=1}^{q}\widetilde{W}_k\cdot\sum_{j=1}^{p}v_{jk}\cdot\sigma(\langle\mathbf{x},\ \mathbf{u}(j)\rangle+\theta_j)\big)<\frac{\varepsilon}{2}.$$

Let $\widetilde{F}_N(\widetilde{\mathbf{X}})=\sum_{k=1}^{q}\widetilde{W}_k\cdot\sum_{j=1}^{p}v_{jk}\cdot\sigma(\langle\widetilde{\mathbf{X}},\ \mathbf{u}(j)\rangle+\theta_j)$ $(\widetilde{\mathbf{X}}\in\mathcal{F}_c(\mathbb{R})^d)$. Then we have, $\widetilde{F}_N\in\widetilde{\mathcal{H}}_0[\sigma]$. And $\widetilde{F}_N$ is a closure fuzzy mapping, by assumption and Theorem 4.15 it follows that $\forall\ \widetilde{\mathbf{X}}\in\mathcal{U},\ D\big(F(\widetilde{\mathbf{X}}),\ G(\widetilde{\mathbf{X}})\big)\leq\varepsilon/2<\varepsilon$, i.e. $\widetilde{\mathcal{H}}_0[\sigma]$ is the universal approximator of F. □

Similarly with Theorem 4.13, we can obtain the equivalent conditions about universal approximators.

Corollary 4.4 *Let σ be a Tauber-Wiener function, and $F : \mathcal{F}_c(\mathbb{R})^d \longrightarrow \mathcal{F}_c(\mathbb{R})$ be a continuous fuzzy function. Then the following facts are equivalent:*

(i) $\widetilde{\mathcal{H}}_0[\sigma]$ is the universal approximator of F;

(ii) $\widetilde{\mathcal{H}}[\sigma]$ is the universal approximator of F;

(iii) F is a closure fuzzy mapping;

(iv) $\mathcal{P}$ is the universal approximator of F.

Proof. Using Theorem 4.17, Remark 4.4, Theorem 4.14 and Theorem 4.15 and with the order: (i)$\Longrightarrow$(ii) $\Longrightarrow$(iii)$\Longrightarrow$(i), (i)$\Longleftrightarrow$(iv), we can show the conclusion. □

4.5.2 Learning algorithm

In this subsection we applying the approach in §4.2 to develop a learning algorithm of the regular FNN's in $\widetilde{\mathcal{H}}_0[\sigma]$. Also we discuss our subjects in subspace $\mathcal{F}_{0c}(\mathbb{R})$. Here we assume that $\sigma(\cdot)$ is a continuously differentiable and nonnegatively increasing function. Choose $\gamma \in \mathbb{N}$, let $\alpha_{k'} = k'/\gamma$ $(k' = 0, 1, ..., \gamma)$. If $\widetilde{F}_N \in \widetilde{\mathcal{H}}_0[\sigma]$, for $i = 1, ..., d$; $k = 1, ..., q$, let

$$\big(\widetilde{W}_k\big)_{\alpha_{k'}} = \big[w^1_{k(k')},\ w^2_{k(k')}\big];\ \big(\widetilde{X}_i\big)_{\alpha_{k'}} = \big[x^1_{i(k')},\ x^2_{i(k')}\big].$$

If $\widetilde{\mathbf{X}} = (\widetilde{X}_1, ..., \widetilde{X}_d) \in \mathcal{F}_{0c}(\mathbb{R})^d$, we introduce the following notations:

$$X^1_{j(k')} = \sum_{i=1}^{d}\big(u_{ij}x^1_{i(k')} \wedge u_{ij}x^2_{i(k')}\big)+\theta_j,\ X^2_{j(k')} = \sum_{i=1}^{d}\big(u_{ij}x^1_{i(k')} \vee u_{ij}x^2_{i(k')}\big)+\theta_j,$$

$$Y^1_{k(k')} = \sum_{j=1}^{p}\big(v_{jk}\sigma(X^1_{j(k')}) \wedge v_{jk}\sigma(X^2_{j(k')})\big),$$

$$Y^2_{k(k')} = \sum_{j=1}^{p}\big(v_{jk}\sigma(X^1_{j(k')}) \vee v_{jk}\sigma(X^2_{j(k')})\big),$$

Then the $\alpha_{k'}$−cut of $\widetilde{F}_N(\widetilde{\mathbf{X}})$ can be represented as follows:

$$\begin{aligned}\widetilde{F}_N(\widetilde{\mathbf{X}})_{\alpha_{k'}} &= \sum_{k=1}^{q}\big[w^1_{k(k')},\ w^2_{k(k')}\big]\cdot\Big(\sum_{j=1}^{p} v_{jk}\cdot\sigma\Big(\sum_{i=1}^{d} u_{ij}\cdot\big[x^1_{i(k')},\ x^2_{i(k')}\big]+\theta_j\Big)\Big)\\ &= \sum_{k=1}^{q}\big[w^1_{k(k')},\ w^2_{k(k')}\big]\cdot\Big(\sum_{j=1}^{p} v_{jk}\cdot\big[\sigma\big(X^1_{j(k')}\big),\ \sigma\big(X^2_{j(k')}\big)\big]\Big),\\ &= \Big[\sum_{k=1}^{q}\big(w^1_{k(k')}Y^1_{k(k')} \wedge w^2_{k(k')}Y^1_{k(k')} \wedge w^1_{k(k')}Y^2_{k(k')} \wedge w^2_{k(k')}Y^2_{k(k')}\big),\\ &\qquad \sum_{k=1}^{q}\big(w^1_{k(k')}Y^1_{k(k')} \vee w^2_{k(k')}Y^1_{k(k')} \vee w^1_{k(k')}Y^2_{k(k')} \vee w^2_{k(k')}Y^2_{k(k')}\big)\Big].\end{aligned} \tag{4.52}$$

Let $\widetilde{F}_N \in \widetilde{\mathcal{H}}_0[\sigma]$. Choose fuzzy pattern pairs $\big((\widetilde{X}_1(l),...,\widetilde{X}_d(l)); \widetilde{O}(l)\big)$ $(l = 1,...,L)$, by which we can train the fuzzy connection weights and thresholds of $\widetilde{F}_N$, where $\big(\widetilde{X}_1(l),...,\widetilde{X}_d(l)\big)$ is the input of $\widetilde{F}_N$, and $\widetilde{O}(l)$ is the corresponding output. Let $\widetilde{O}(l)_{\alpha_{k'}} = \big[o^1_{k'}(l),\ o^2_{k'}(l)\big]$, $\big(\widetilde{X}_i(l)\big)_{\alpha_{k'}} = \big[x^1_{i(k')}(l),\ x^2_{i(k')}(l)\big]$. Similarly with (4.24), we define the following error function:

$$\begin{aligned} E &= \frac{1}{2}\sum_{l=1}^{L}\Big(\sum_{k'=0}^{\gamma} d_E\Big(\widetilde{F}_N\ \big(\widetilde{X}_1(l),...,\widetilde{X}_d(l)\big)_{\alpha_{k'}},\ \big(\widetilde{O}(l)\big)_{\alpha_{k'}}\Big)^2\Big) \\ &= \frac{1}{2}\sum_{l=1}^{L}\sum_{k'=0}^{\gamma}\Big(\Big[o^1_{k'}(l) - \sum_{k=1}^{q} Z^1_{k(k')}(l)\Big]^2 + \Big[o^2_{k'}(l) - \sum_{k=1}^{q} Z^2_{k(k')}(l)\Big]^2\Big). \end{aligned} \tag{4.53}$$

where $Z^1_{k(k')}(l)$, $Z^2_{k(k')}(l)$ are defined respectively as follows:

$$Z^1_{k(k')}(l) = w^1_{k(k')}Y^1_{k(k')}(l) \wedge w^2_{k(k')}Y^1_{k(k')}(l) \wedge w^1_{k(k')}Y^2_{k(k')}(l) \wedge w^2_{k(k')}Y^2_{k(k')}(l),$$
$$Z^2_{k(k')}(l) = w^1_{k(k')}Y^1_{k(k')}(l) \vee w^2_{k(k')}Y^1_{k(k')}(l) \vee w^1_{k(k')}Y^2_{k(k')}(l) \vee w^2_{k(k')}Y^2_{k(k')}(l),$$

and $Y^1_{k(k')}(l)$, $Y^2_{k(k')}(l)$ is the representation of $Y^1_{k(k')}$, $Y^2_{k(k')}$, respectively, by letting $x^1_{i(k')} = x^1_{i(k')}(l)$, $x^2_{i(k')} = x^2_{i(k')}(l)$. We write all adjustable parameters related to $\widetilde{F}_N$ as a vector $\mathbf{w} = (w_1,...,w_N)$, that is

$$\mathbf{w} = \big(u_{11},...,u_{dp}, v_{11},...,v_{pq}, \theta_1,...,\theta_p, w^1_{1(0)},...,w^1_{q(\gamma)}, w^2_{q(\gamma)},...,w^2_{1(0)}\big).$$

Similarly with Theorem 4.6, we can prove

Theorem 4.18 *Let the transfer function $\sigma : \mathbb{R} \longrightarrow \mathbb{R}$ be continuously differentiable and non-negatively increasing. Then the error function defined by (4.53) is differentiable a.e. with respect to $\mathbf{w}$ in $\mathbb{R}^N$. Further, for $r = 1,2$; $k = 1,...,q$; $k' = 0,1,...,\gamma$; $j = 1,...,p$; $i = 1,...,d$, if let*

$$\Delta^1_{k'}(l) = \sum_{k=1}^{q} Z^1_{k(k')}(l) - o^1_{k'}(l),\quad \Delta^2_{k'}(l) = \sum_{k=1}^{q} Z^2_{k(k')}(l) - o^2_{k'}(l);$$
$$D^1_{k(k')}(l) = w^1_{k(k')}Y^1_{k(k')}(l) \wedge w^1_{k(k')}Y^2_{k(k')}(l) - w^2_{k(k')}Y^1_{k(k')}(l) \wedge w^2_{k(k')}Y^2_{k(k')}(l);$$
$$D^2_{k(k')}(l) = w^1_{k(k')}Y^1_{k(k')}(l) \vee w^1_{k(k')}Y^2_{k(k')}(l) - w^2_{k(k')}Y^1_{k(k')}(l) \vee w^2_{k(k')}Y^2_{k(k')}(l);$$
$$A^r_{k(k')}(l) = \mathrm{lor}\big((-1)^r D^1_{k(k')}(l)\big)\big(Y^1_{k(k')}(l)\mathrm{lor}\big(w^r_{k(k')}\big) + Y^2_{k(k')}(l)\mathrm{lor}\big(-w^r_{k(k')}(l)\big)\big);$$
$$B^r_{k(k')}(l) = \mathrm{lor}\big((-1)^{r+1} D^2_{k(k')}(l)\big)\big(Y^1_{k(k')}(l)\mathrm{lor}\big(-w^r_{k(k')}\big) + Y^2_{k(k')}(l)\mathrm{lor}\big(w^r_{k(k')}(l)\big)\big);$$
$$U^r_{k(k')}(l) = \mathrm{lor}\big((-1)^r w^2_{k(k')}\big)\mathrm{lor}\big(D^1_{k(k')}(l)\big) + \mathrm{lor}\big((-1)^r w^1_{k(k')}\big)\mathrm{lor}\big(-D^1_{k(k')}(l)\big);$$
$$V^r_{k(k')}(l) = \mathrm{lor}\big((-1)^{r+1} w^1_{k(k')}\big)\mathrm{lor}\big(D^2_{k(k')}(l)\big) + \mathrm{lor}\big((-1)^{r+1} w^2_{k(k')}\big)\mathrm{lor}\big(-D^2_{k(k')}(l)\big);$$

$$C^r_{ijk(k')}(l)=v_{jk}\Big(\mathrm{lor}\big((-1)^r v_{jk}\big)\sigma'\big(X^2_{j(k')}(l)\big)\big(\sum_{t=1}^{2}x^{3-t}_{i(k')}(l)\mathrm{lor}((-1)^{t+1}u_{ij})\big)+$$
$$+\mathrm{lor}\big((-1)^{r+1}v_{jk}\big)\sigma'\big(X^1_{j(k')}(l)\big)\big(\sum_{t=1}^{2}x^{3-t}_{i(k')}(l)\mathrm{lor}((-1)^{t}u_{ij})\big)\Big);$$

$$H^1_{k(k')}(l)=\Delta^1_{k'}(l)U^1_{k(k')}(l)+\Delta^2_{k'}(l)V^1_{k(k')}(l);$$

$$H^2_{k(k')}(l)=\Delta^1_{k'}(l)U^2_{k(k')}(l)+\Delta^2_{k'}(l)V^2_{k(k')}(l).$$

we obtain the following partial derivative formulas:

$$(i)\ \frac{\partial E}{\partial\theta_j}=\sum_{l=1}^{L}\sum_{k'=0}^{\gamma}\sum_{k=1}^{q}\Big\{H^1_{k(k')}(l)\big(\sum_{t=1}^{2}\sigma'(X^{3-t}_{j(k')}(l))\mathrm{lor}((-1)^{t+1}v_{jk})\big)+$$
$$+H^2_{k(k')}(l)\big(\sum_{t=1}^{2}\sigma'(X^{3-t}_{j(k')}(l))\mathrm{lor}((-1)^{t}v_{jk})\big)\Big\}.$$

$$(ii)\ \frac{\partial E}{\partial u_{ij}}=\sum_{l=1}^{L}\sum_{k'=0}^{\gamma}\sum_{k=1}^{q}\Big\{\Delta^1_{k'}(l)\Big(U^1_{k(k')}(l)C^2_{ijk(k')}(l)+U^2_{k(k')}(l)C^1_{ijk(k')}(l)\Big)+$$
$$+\Delta^2_{k'}(l)\Big(V^1_{k(k')}(l)C^2_{ijk(k')}(l)+V^2_{k(k')}(l)C^1_{ijk(k')}(l)\Big)\Big\};$$

$$(iii)\ \frac{\partial E}{\partial v_{jk}}=\sum_{l=1}^{L}\sum_{k'=0}^{\gamma}\Big\{H^1_{k(k')}(l)\big(\sum_{t=1}^{2}\sigma(X^{3-t}_{j(k')}(l))\mathrm{lor}((-1)^{t}v_{jk})\big)$$
$$+H^2_{k(k')}(l)\big(\sum_{t=1}^{2}\sigma(X^{3-t}_{j(k')}(l))\mathrm{lor}((-1)^{t+1}v_{jk})\big)\Big\};$$

$$(iv)\ \frac{\partial E}{\partial w^r_{k(k')}}=\sum_{l=1}^{L}\big\{A^r_{k(k')}(l)\Delta^1_{k'}(l)+B^r_{k(k')}(l)\Delta^2_{k'}(l)\big\}\ (r=1,2).$$

Proof. Similarly with Theorem 4.6 we can directly show (iv). Considering the following facts we can prove (i)–(iii), respectively:

$$\frac{\partial Z^r_{k(k')}(l)}{\partial u_{ij}}=\frac{\partial Z^r_{k(k')}(l)}{\partial Y^1_{k(k')}}\cdot\frac{\partial Y^1_{k(k')}(l)}{\partial u_{ij}}+\frac{\partial Z^r_{k(k')}(l)}{\partial Y^2_{k(k')}}\cdot\frac{\partial Y^2_{k(k')}(l)}{\partial u_{ij}};$$

$$\frac{\partial Z^r_{k(k')}(l)}{\partial v_{jk}}=\frac{\partial Z^r_{k(k')}(l)}{\partial Y^1_{k(k')}}\cdot\frac{\partial Y^1_{k(k')}(l)}{\partial v_{jk}}+\frac{\partial Z^r_{k(k')}(l)}{\partial Y^2_{k(k')}}\cdot\frac{\partial Y^2_{k(k')}(l)}{\partial v_{jk}};$$

$$\frac{\partial Z^r_{k(k')}(l)}{\partial \theta_j}=\frac{\partial Z^r_{k(k')}(l)}{\partial Y^1_{k(k')}}\cdot\frac{\partial Y^1_{k(k')}(l)}{\partial \theta_j}+\frac{\partial Z^r_{k(k')}(l)}{\partial Y^2_{k(k')}}\cdot\frac{\partial Y^2_{k(k')}(l)}{\partial \theta_j}.$$

where $r=1,2$. Similarly with Theorem 4.6, we can show that E is differentiable a.e. with respect to $\mathbf{w}$ in $\mathbb{R}^N$. The theorem is proved. □

By the partial derivatives determined by Theorem 4.18, we can construct an improved fuzzy BP algorithm, that is, the accelerated convergence fuzzy BP algorithm. Similarly with Algorithm 4.2, we let the learning constant η vary as the time step iterates, i.e. suppose $\eta=\eta[t]=\rho(E[t])$, η is defined by the error function $E[t]$ in each iteration: $\rho(E[t])=\rho_0E[t]/\big\|\nabla E(\mathbf{w})\big\|^2$, where

$\nabla E(\mathbf{w}) = \big(\partial E/\partial w_1, ..., \partial E/\partial w_N\big)$ is a gradient vector, ρ_0 is a given constant. we obtain the following iteration scheme:

$$\begin{cases} u_{ij}[t+1] = u_{ij}[t] - \rho_0 \cdot E[t] \cdot \dfrac{\partial E[t]}{\partial u_{ij}[t]} \Big/ \big\|\nabla E(\mathbf{w})\big\|^2; \\ v_{jk}[t+1] = v_{jk}[t] - \rho_0 \cdot E[t] \cdot \dfrac{\partial E[t]}{\partial v_{jk}[t]} \Big/ \big\|\nabla E(\mathbf{w})\big\|^2; \\ \theta_j[t+1] = \theta_j[t] - \rho_0 \cdot E[t] \cdot \dfrac{\partial E[t]}{\partial \theta_j[t]} \Big/ \big\|\nabla E(\mathbf{w})\big\|^2; \\ w^r_{k(k')}[t+1] = w^r_{k(k')}[t] - \rho_0 \cdot E[t] \cdot \dfrac{\partial E[t]}{\partial w^r_{k(k')}[t]} \Big/ \big\|\nabla E(\mathbf{w})\big\|^2 \ (r = 1, 2). \end{cases} \tag{4.54}$$

Let ρ_0 ba a small positive number, such as, $\rho_0 = 0.01$. We can construct the learning algorithm for $\widetilde{F}_N \in \widetilde{\mathcal{H}}_0[\sigma]$:

Algorithm 4.3 Accelerated convergence fuzzy BP algorithm.

Step 1. Randomly choose initial values: $u_{ij}[0]$, $v_{jk}[0]$, $\theta_j[0]$ and $w^r_{k(k')}[0]$ $(r = 1, 2)$, and let $t = 0$;

Step 2. Calculate the following partial derivatives:

$$\frac{\partial E[t]}{\partial u_{ij}[t]}, \ \frac{\partial E[t]}{\partial v_{jk}[t]}, \ \frac{\partial E[t]}{\partial \theta_j[t]}, \ \frac{\partial E[t]}{\partial w^r_{k(k')}[t]};$$

Step 3. According to iteration scheme (4.54) update the parameters u_{ij}, v_{jk}, θ_j and $w^r_{k(k')}$;

Step 4. For each $k \in \{1, ..., q\}$, with the increasing order, we re-array the set $\{w^r_{k(k')} | r = 1, 2;\ k' = 0, 1, .., \gamma\}$, that is, from small to large we have

$$w^1_{k(0)} \le w^1_{k(1)} \le \cdots \le w^1_{k(\gamma)} \le w^2_{k(\gamma)} \le \cdots \le w^2_{k(0)}.$$

Step 5. Discriminate whether $\big|E[t]\big| < \varepsilon$? If yes go to Step 6, otherwise let $t = t + 1$ go to Step 2;

Step 6. Output all parameters.

Using Algorithm 4.3, we can employ the four layer regular feedforward FNN's in $\widetilde{\mathcal{H}}_0[\sigma]$ to represent a class of fuzzy functions, approximately, that is, we can realize the result of Theorem 4.16 by this algorithm.

Example 4.2 Define fuzzy function $F : \mathcal{F}_{0c}(\mathbb{R})^2 \longrightarrow \mathcal{F}_{0c}(\mathbb{R})$ as follows:

$$\forall\, \widetilde{\mathbf{X}} = (\widetilde{X}_1, \widetilde{X}_2) \in \mathcal{F}_{0c}(\mathbb{R})^2, \ F(\widetilde{\mathbf{X}}) = F(\widetilde{X}_1, \widetilde{X}_2) = \max\big(\widetilde{X}_1, \widetilde{X}_2\big), \tag{4.55}$$

where $\forall x \in \mathbb{R}$, define $\max\big(\widetilde{X}_1, \widetilde{X}_2\big)(x) = \widetilde{X}_1(x) \vee \widetilde{X}_2(x)$. Then easily we can imply, the fuzzy function F defined by (4.55) is continuously increasing closure fuzzy mapping. Choose fuzzy set family $\mathcal{U} = \{(\widetilde{X}(l_1),\ \widetilde{X}(l_2))\,|\,l_1, l_2 \in \mathbb{N}\bigcup\{0\}\}$, where $\widetilde{X}(l)$ $(l \in \mathbb{N})$ is defined as follows:

$$\forall x \in \mathbb{R},\ \widetilde{X}(l)(x) = \begin{cases} \dfrac{2x(5l^2+1)+5l^2+3}{5l^2+3}, & -\dfrac{1}{5l^2+1} - \dfrac{1}{2} \leq x \leq 0, \\ \dfrac{-2x(5l^2+1)+5l^2+3}{5l^2+3}, & 0 < x \leq \dfrac{1}{5l^2+1} + \dfrac{1}{2}, \\ 0, & \text{otherwise;} \end{cases}$$

$$\forall x \in \mathbb{R},\quad \widetilde{X}(0)(x) = \begin{cases} 1+2x, & -\dfrac{1}{2} \leq x \leq 0, \\ 1-2x, & 0 < x \leq \dfrac{1}{2}, \\ 0, & \text{otherwise.} \end{cases}$$

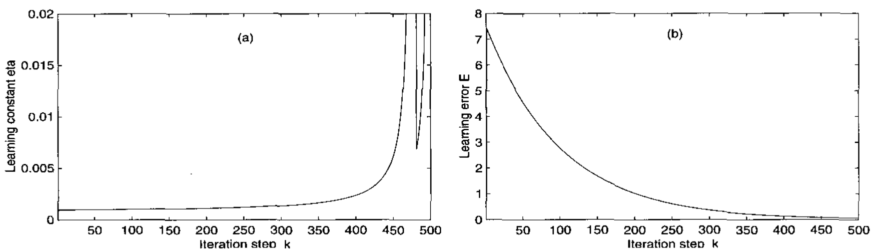


Figure 4.19 (a): Changing law of η; (b): Changing curve of E

We can show that $\mathcal{U} \subset \mathcal{F}_{0c}(\mathbb{R})^2$ is a compact set. For given $\varepsilon = 0.1$. To apply a four layer regular feedforward FNN to represent F under the approximate sense "$\approx_\varepsilon$" , we choose the fuzzy pattern family

$$\mathcal{U}_0 = \{(\widetilde{X}(l_1),\ \widetilde{X}(l_2))\,|\,l_1, l_2 = 0, 1, 2\}.$$

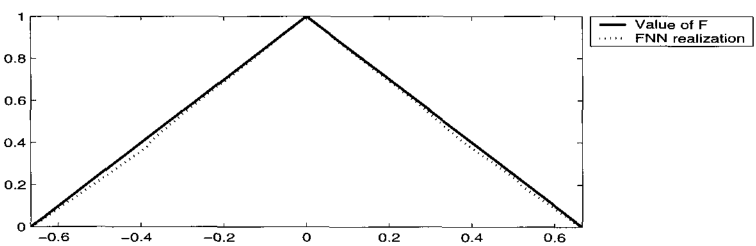


Figure 4.20 Fuzzy set $F(\widetilde{X}(0), \widetilde{X}(1))$ and fuzzy set $\widetilde{F}_N(\widetilde{X}(0), \widetilde{X}(1))$

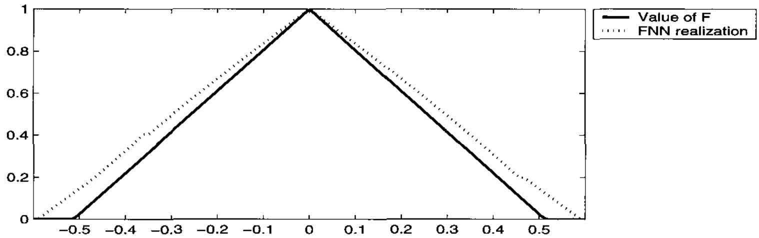


Figure 4.21 Fuzzy set $F(\widetilde{X}(4), \widetilde{X}(4))$ and fuzzy set $\widetilde{F}_N(\widetilde{X}(4), \widetilde{X}(4))$

We can easily show, $\lim_{l\to+\infty} D\big(\widetilde{X}(l),\ \widetilde{X}(0)\big)= 0$, furthermore, $\forall\ \widetilde{\mathbf{X}}\in \mathcal{U}$, there is $\widetilde{\mathbf{Y}}\in \mathcal{U}_0$, such that $H\big(\widetilde{\mathbf{X}},\ \widetilde{\mathbf{Y}}\big)< \varepsilon/2$, that is, $\mathcal{U}_0$ is a $\varepsilon/2$-net of $\mathcal{U}$. Since $\widetilde{X}(0) \subset\widetilde{X}(2) \subset\widetilde{X}(1)$, we obtain under the sense "$\approx_\varepsilon$" the learning patterns for realizing fuzzy function F as follows:

$$\begin{aligned}&\big((\widetilde{X}(0),\widetilde{X}(0));\widetilde{X}(0)\big),\quad \big((\widetilde{X}(0),\widetilde{X}(1));\widetilde{X}(1)\big),\quad \big((\widetilde{X}(0),\widetilde{X}(2));\widetilde{X}(2)\big),\\ &\big((\widetilde{X}(1),\widetilde{X}(0));\widetilde{X}(1)\big),\quad \big((\widetilde{X}(1),\widetilde{X}(1));\widetilde{X}(1)\big),\quad \big((\widetilde{X}(1),\widetilde{X}(2));\widetilde{X}(1)\big),\\ &\big((\widetilde{X}(2),\widetilde{X}(0));\widetilde{X}(2)\big),\quad \big((\widetilde{X}(1),\widetilde{X}(1));\widetilde{X}(1)\big),\quad \big((\widetilde{X}(2),\widetilde{X}(2));\widetilde{X}(2)\big).\end{aligned}$$

So let $L = 9$, and choose $\gamma = 1$. And two hidden layers have five hidden neurons, respectively, that is, $p = q = 5$. In Algorithm 4.3, after 500 iterations we can obtain the changing curve of the error function E with respect to the iteration step t, as shown in Figure 4.19(b). As t increasing, $E[t]$ converges to 0. The convergence speed of $\{E[t]\,|t \in \mathbb{N}\}$ is much quicker than that of Algorithm 4.1. Figure 4.19(a) illustrates the changing law of the learning constant η in Algorithm 4.3 with respect to the iteration step t.

Randomly we choose $\big(\widetilde{X}(0),\widetilde{X}(1)\big)$, $\big(\widetilde{X}(4),\widetilde{X}(4)\big)\in \mathcal{U}$, Calculate the values of fuzzy function F and FNN $\widetilde{F}_N$ at the given two points, respectively, as shown in Figure 4.20 and Figure 4.21. Thus, it can be seen that with the given accuracy ε, the fuzzy function F can be realized by a four layer regular feedforward FNN, approximately.

Similarly with studying the BP algorithm, we can show the convergence of the accelerated convergence fuzzy BP Algorithm 4.3. Also similar with Theorem 4.7 we define the nonzero weight space $\mathcal{W}_{nn}$ of $\widetilde{\mathcal{H}}_0[\sigma]$. In $\mathcal{W}_{nn}$, By (4.56) we imply that the error function E is differentiable. Also we can show

$$E[t+1]-E[t] = \sum_{i=1}^{d}\Big(\frac{\partial E}{\partial w_i[t]}\Big)\cdot\big(w_i[t+1]-w_i[t]\big)\leq 0.$$

Considering $E[t] \geq 0$, we have, $\big\{E[t]\,\big|t \in \mathbb{N}\big\}$ is a convergent sequence. By the illustration, we can also see that the convergence speed of $\big\{E[t]\,\big|t \in \mathbb{N}\big\}$ is much quicker than that of the fuzzy BP Algorithm 4.1.

§4.6 Approximation of regular FNN with integral norm

In preceding two sections we study the universal approximation of regular FNN's with Hausdroff metric for fuzzy sets, systematically. That is, the regular FNN's can with arbitrary accuracy approximate a class of continuous fuzzy functions on each compact set of $f_c(\mathbb{R}^d)$. Such facts can provide us with the theoretic basis for applying FNN's to many real fields, such as system modeling, system identification and so on. However, if a real I/O system is not continuous, but a general integrable system, how can we use FNN's to solve the similar problems? That is, whether can we study the universal approximation of FNN's with general integral norm? In the section we focus on fuzzy valued integrable functions, and present the approximation capability of regular FNN's to a large class of fuzzy valued functions.

4.6.1 Integrable bounded fuzzy valued functions

For a given set $T \subset \mathbb{R}^d$, and $(T, \mathcal{B}_T, \mu)$ is a Lebesgue measure space, where $\mathcal{B}_T = \mathcal{B} \cap T \triangleq \{C \cap T : C \in \mathcal{B}\}$, and $\mathcal{B}$ is a Lebesgue measurable set family in $\mathbb{R}^d$. Let $\mathcal{F}_0^c(\mathbb{R}) \subset \mathcal{F}_0(\mathbb{R})$, so that $\big(\mathcal{F}_0^c(\mathbb{R}), D\big)$ is a completely separable metric space. By [64] we imply, $\mathcal{F}_0^c(\mathbb{R})$ can be the triangular fuzzy number space, or trapezoidal fuzzy number space, and so on. denote $L^1(T) \triangleq L^1(T, \mathcal{B}_T, \mu) = \big\{f : T \longrightarrow \mathbb{R} \,\big|\, \int_T |f(x)| \mathrm{d}\mu < +\infty\big\}$.

Definition 4.4 The fuzzy valued function $F : T \longrightarrow \mathcal{F}_0^c(\mathbb{R})$ is called to be integrable and bounded on T, if there is $\rho \in L^1(T)$, satisfying $\forall \mathbf{x} \in T$, $\forall \alpha \in [0,1]$, $\forall y \in F(\mathbf{x})_\alpha$, $|y| \le \rho(\mathbf{x})$.

As an example of integrable and bounded fuzzy valued functions, easily we can show the following conclusion.

Remark 4.5 (i) If the set $T \subset \mathbb{R}^d$ is bounded, and $F : \mathbb{R}^d \longrightarrow \mathcal{F}_0^c(\mathbb{R})$ is continuous, then we can choose $\rho(\mathbf{x}) = |F(\mathbf{x})|\,(\mathbf{x} \in T)$. Therefore, F is integrable bounded on T;

(ii) Let $F : T \longrightarrow \mathcal{F}_0^c(\mathbb{R})$ be integrable and bounded on T. Then $|F(\cdot)| \in L^1(T)$.

Let $\mathcal{L}^1(T, \mathcal{B}_T, \mu)$ be a collection of all integrable bounded fuzzy valued functions that $T \longrightarrow \mathcal{F}_0^c(\mathbb{R})$. For F_1, $F_2 \in \mathcal{L}^1(T, \mathcal{B}_T, \mu)$, define

$$\Delta(F_1, F_2) = \int_T D(F_1(x), F_2(x)) \mathrm{d}\mu, \tag{4.56}$$

Then by Remark 4.5, easily we have, for F_1, $F_2 \in \mathcal{L}^1(T, \mathcal{B}_T, \mu)$, $\Delta(F_1,\, F_2) < +\infty$. Considering $(\mathcal{F}_0^c(\mathbb{R}), D)$ is a completely separable metric space, we show that the metric space $\big(\mathcal{L}^1(T, \mathcal{B}_T, \mu), \Delta\big)$ is also complete and separable.

Definition 4.5 Let $S : T \longrightarrow \mathcal{F}_0^c(\mathbb{R})$, and $\{T_i : 1 \le i \le K\}$ be a finite partition of T, i.e. $\bigcup_{k=1}^{K} T_k = T$, $T_i \cap T_j = \emptyset\ (i \ne j)$, and $T_i \in \mathcal{B}_T\ (i = 1, ..., K)$.

If there exist $\widetilde{A}_1, ..., \widetilde{A}_K \in \mathcal{F}_0^c(\mathbb{R})$, such that

$$\forall \mathbf{x} \in T, S(\mathbf{x}) = \sum_{k=1}^{K} \chi_{T_k}(\mathbf{x}) \cdot \widetilde{A}_k,$$

we call S a fuzzy valued simple function on T, where $\chi_{T_k}(\cdot)$ is a characteristic function of T_k.

Similarly with the conventional measure theory, we at first prove that the collection of all fuzzy valued simple functions on T is dense in $\mathcal{L}^1(T, \mathcal{B}_T, \mu)$.

Theorem 4.19 *Assume that $F : T \longrightarrow \mathcal{F}_0^c(\mathbb{R})$ is an integrable bounded function, and $\mu(T) < +\infty$. Given arbitrarily $\varepsilon > 0$, then there is a fuzzy valued simple function $S : T \longrightarrow \mathcal{F}_0^c(\mathbb{R})$, such that $\Delta(F, S) < \varepsilon$.*

Proof. Since F is integrable and bounded, by Remark 4.5 we obtain, there is $\delta > 0$, satisfying, for each $P \subset T$, $\mu(P) < \delta$, such that $\int_P |F(\mathbf{x})| \mathrm{d}\mu < \varepsilon/2$. It is no harm to assume $\mu(T) = 1$. Since $(\mathcal{F}_0^c(\mathbb{R}), D)$ is a separable space, we let $\mathcal{A}_0 = \{\widetilde{A}_i, i \in \mathbb{N}\}$ be a dense subset of $\mathcal{F}_0^c(\mathbb{R})$. Then $\forall \widetilde{X} \in \mathcal{F}_0^c(\mathbb{R})$, there is $i \in \mathbb{N}$, such that $D(\widetilde{X}, \widetilde{A}_i) < \varepsilon/2$. Let

$$T_1 = \{\mathbf{x} \in T | D(F(\mathbf{x}), \widetilde{A}_1) < \frac{\varepsilon}{2}\},$$

$$T_2 = \{\mathbf{x} \in T | D(F(\mathbf{x}), \widetilde{A}_1) \geq \frac{\varepsilon}{2}, D(F(\mathbf{x}), \widetilde{A}_2) < \frac{\varepsilon}{2}\},$$

....

$$T_n = \{\mathbf{x} \in T | D(F(\mathbf{x}), \widetilde{A}_i) \geq \frac{\varepsilon}{2} \ (i = 1, ..., n-1), \ D(F(\mathbf{x}), \widetilde{A}_n) < \frac{\varepsilon}{2}\},$$

....

Easily we have, $T_i \cap T_j = \emptyset \ (i \neq j)$. For each $\mathbf{x} \in T$, if $\mathbf{x} \notin T_1$, there is $i_0 \in \mathbb{N}$, so that $D(F(\mathbf{x}), \widetilde{A}_{i_0}) < \varepsilon/2$, and $D(F(\mathbf{x}), \widetilde{A}_i) \geq \varepsilon/2 \ (i < i_0)$. Thus, $\mathbf{x} \in T_{i_0}$. Therefore, $T = \bigcup_{i \in \mathbb{N}} T_i$. Considering the following fact:

$$\mu(T) = \mu\Big(\bigcup_{i \in \mathbb{N}} T_i\Big) = \sum_{i \in \mathbb{N}} \mu(T_i) < +\infty,$$

we imply, there is $d_0 \in \mathbb{N}$, satisfying $\mu\big(\bigcup_{i > d_0} T_i\big) < \delta$. Let $S(\mathbf{x}) = \sum_{i=1}^{d_0} \chi_{T_i}(\mathbf{x}) \cdot \widetilde{A}_i$ $(\mathbf{x} \in T)$. And write $T_0 = \bigcup_{i > d_0} T_i$, then $\{T_0, T_1, ..., T_{d_0}\}$ is a finite partition of T. Let $\widetilde{A}_0 \equiv 0$, easily it follows that S is a fuzzy valued simple function on T, furthermore

$$\forall i : 1 \leq i \leq d_0, \forall \mathbf{x} \in T_i, \ D(F(\mathbf{x}), S(\mathbf{x})) = D(F(\mathbf{x}), \widetilde{A}_i) < \frac{\varepsilon}{2};$$

$$\int_{T_0} D(F(\mathbf{x}), S(\mathbf{x})) \mathrm{d}\mu = \int_{T_0} D(F(\mathbf{x}), \{0\}) \mathrm{d}\mu = \int_{T_0} |F(\mathbf{x})| \mathrm{d}\mu < \frac{\varepsilon}{2},$$

Also we have

$$\Delta(F,S) = \int_T D\big(F(\mathbf{x}),\, S(\mathbf{x})\big)\mathrm{d}\mu = \int_{\bigcup\limits_{i=0}^{d_0} T_i} D\big(F(\mathbf{x}),\, S(\mathbf{x})\big)\mathrm{d}\mu$$

$$= \sum_{i=1}^{d_0}\int_{T_i} D\big(F(\mathbf{x}),\, S(\mathbf{x})\big)\mathrm{d}\mu + \int_{T_0} D\big(F(\mathbf{x}),\, S(\mathbf{x})\big)\mathrm{d}\mu$$

$$< \sum_{i=1}^{d_0}\int_{T_i} D\big(F(\mathbf{x}),\, \widetilde{A}_i\big)\mathrm{d}\mu + \frac{\varepsilon}{2} \le \frac{\varepsilon}{2}\sum_{i=1}^{d_0}\int_{T_i}\mathrm{d}\mu + \frac{\varepsilon}{2} \le \frac{\varepsilon}{2}\mu(T) + \frac{\varepsilon}{2} = \varepsilon.$$

Thus, the theorem is proved. □

By $\mathcal{S}$ we denote the collection of all fuzzy valued simple functions thst $\mathbb{R}^d \longrightarrow \mathcal{F}_0^c(\mathbb{R})$. For each $T \subset \mathbb{R}^d$, $S \in \mathcal{S}$, then the restriction of S on T is also a fuzzy valued simple function on T, which is also written as S.

4.6.2 Universal approximation with integral norm

By Theorem 4.19, if $(T, \mathcal{B}_T, \mu)$ is a bounded measure space, then $\mathcal{S}$ is the universal approximator of each fuzzy valued integrable bounded function with integral norm, that is, $\forall \varepsilon > 0$, $\forall F \in \mathcal{L}^1(T, \mathcal{B}_T, \mu)$, there is $S \in \mathcal{S}$, so that $\Delta(F,\, S) < \varepsilon$. Also we call S the $L_1(\mu)-$norm approximation of F.

Theorem 4.20 *Let $T \subset \mathbb{R}^d$ be a bounded set, so $(T, \mathcal{B}_T, \mu)$ be a bounded measure space. Then $\big(\mathcal{L}^1(T, \mathcal{B}_T, \mu),\, \Delta\big)$, the integrable bounded fuzzy valued function space is a completely separable metric space.*

Proof. It is no harm to assume that $T = [0,1]^d$ is a compact set. Then $\mu(T) = 1$. Easily we have, Δ is a metric of the space $\mathcal{L}^1(T, \mathcal{B}_T, \mu)$. The completeness can be demonstrated as that of $L_1(T)$. So it suffices to prove sufficiency. Let $\mathcal{A}_0 = \big\{\widetilde{A}_1, \widetilde{A}_2, ...\big\}$ is a dense subset of $\big(\mathcal{F}_0^c(\mathbb{R}),\, D\big)$. Write

$$\mathcal{C}(T) = \big\{F : T \longrightarrow \mathcal{F}_0^c(\mathbb{R})\big|F \,\text{is continuous on}\, T\big\},$$

In the following we show, $\mathcal{C}(T)$ is dense in $\mathcal{S}$. Choose a closed set $B \subset T$. For $x \in T$, define $L_B(\mathbf{x}) \in \mathbb{R}_+ :\; L_B(\mathbf{x}) = \inf\big\{\|\mathbf{x} - \mathbf{x}_1\|\,\big|\mathbf{x}_1 \in B\big\}$. Let

$$Q_n(\mathbf{x}) = \frac{1}{1 + n\cdot L_B(\mathbf{x})}\;(n \in \mathbb{N}), \Longrightarrow \begin{cases} Q_n(\mathbf{x}) \equiv 1, & (\mathbf{x} \in B), \\ \lim\limits_{n\to+\infty} Q_n(\mathbf{x}) = 0, & (\mathbf{x} \notin B). \end{cases}$$

For each $\widetilde{A} \in \mathcal{F}_0^c(\mathbb{R})$, if let $F(\mathbf{x}) = \widetilde{A}\cdot\chi_B(\mathbf{x})$, $F_n(\mathbf{x}) = \widetilde{A}\cdot Q_n(\mathbf{x})$ $(\mathbf{x} \in T)$, by Lebesgue's control convergent theorem we obtain

$$\Delta(F, F_n) = \int_T D\big(F(\mathbf{x}), F_n(\mathbf{x})\big)\mathrm{d}\mu \le |\,\widetilde{A}\,|\cdot\int_T |\chi_B(\mathbf{x}) - Q_n(\mathbf{x})|\mathrm{d}\mu \longrightarrow 0\;(n\to+\infty).$$

By the definition of Lebesgue measurable set, $\forall B \in \mathcal{B}_T$, $\forall\, \widetilde{A} \in \mathcal{F}_0^c(\mathbb{R})$, it follows that $\{Q_n\} \subset \mathcal{C}(T)$, satisfying $\lim\limits_{n\to+\infty}\Delta\big(\widetilde{A}\cdot\chi_B,\; Q_n\big) = 0$. Thus, $\forall \varepsilon > 0$, $\forall S \in \mathcal{S}$, By Lemma 4.7 it is easy to show, there is $Q \in \mathcal{C}(T)$, such that $\Delta(S, Q_n) < \varepsilon$, that is, $\mathcal{C}(T)$ is dense in $\mathcal{S}$. Let

$$\mathcal{C}_P(T)=\Big\{F : T \longrightarrow \mathcal{F}_0^c(\mathbb{R})\big|F(\mathbf{x})= \sum_{i_1,\ldots,i_d=0}^{m} \widetilde{A}_{i_1\ldots i_d} \cdot K_{m;i_1\ldots i_d}(\mathbf{x}),\ \widetilde{A}_{i_1\ldots i_d}\in \mathcal{A}_0\Big\},$$

Then $\mathcal{C}_P(T)$ is a countable set. Theorem 4.11 implies, $\forall \varepsilon > 0$, $\forall F \in \mathcal{C}(T)$, there is a fuzzy valued Bernstein polynomial $B_m(F)$ which is defined as follows: $B_m(F;\mathbf{x}) = \sum\limits_{i_1,\ldots,i_d=0}^{m} \widetilde{B}_{i_1\ldots i_d} \cdot K_{m;i_1\ldots i_d}(\mathbf{x})$, where $\widetilde{B}_{i_1\ldots i_d}\in \mathcal{F}_0^c(\mathbb{R})$, satisfying $\forall \mathbf{x} \in T$, $D(F(\mathbf{x}),\ B_m(F;\mathbf{x})) < \varepsilon/2$. Since $\mathcal{A}_0$ is dense in $(\mathcal{F}_0^c(\mathbb{R}), D)$, there is $\{\widetilde{A}_{i_1\ldots i_d}\} \subset \mathcal{A}_0$, so that $\forall i_1, ..., i_d \in \{0, 1, ..., m\}$, $D(\widetilde{B}_{i_1\ldots i_d},\ \widetilde{A}_{i_1\ldots i_d}) < \varepsilon/2$. So using Lemma 4.7 and (4.38) we obtain

$$\begin{aligned} D\Big(B_m(F;\mathbf{x}),\ &\sum_{i_1,\ldots,i_d=0}^{m} \widetilde{A}_{i_1\ldots i_d} \cdot K_{m;i_1\ldots i_d}(\mathbf{x})\Big) \\ &\le \sum_{i_1,\ldots,i_d=0}^{m} K_{m;i_1\ldots i_d}(\mathbf{x}) \cdot D(\widetilde{A}_{i_1\ldots i_d},\ \widetilde{B}_{i_1\ldots i_d}) < \frac{\varepsilon}{2}, \end{aligned}$$

where $\mathbf{x} \in T$. Let $P_m(\mathbf{x}) = \sum\limits_{i_1,\ldots,i_d=0}^{m} \widetilde{A}_{i_1\ldots i_d} \cdot K_{m;i_1\ldots i_d}(\mathbf{x})$. Then $P_m \in \mathcal{C}_P(T)$, furthermore

$$\begin{aligned} \Delta(F,\ P_m) &\le \Delta(F,\ B_m(F))+\Delta(B_m(F),\ P_m) \\ &= \int_T D(F(\mathbf{x}),\ B_m(F;\mathbf{x}))\mathrm{d}\mu + \int_T D(B_m(F;\mathbf{x}),\ P_m(\mathbf{x}))\mathrm{d}\mu \le \varepsilon. \end{aligned}$$

Hence $\mathcal{C}_P(T)$ is dense in $\mathcal{C}(T)$, and also dense in $\mathcal{S}$. By Theorem 4.19, we imply, $\mathcal{C}_P(T)$ is dense in $(\mathcal{L}^1(T, \mathcal{B}_T, \mu), \Delta)$. Thus, $(\mathcal{L}^1(T, \mathcal{B}_T, \mu), \Delta)$ is separable. Therefore, $(\mathcal{L}^1(T, \mathcal{B}_T, \mu), \Delta)$ is a completely separable metric space. □

We call the transfer function σ to be $L^1(T)-$universal [9], if $\forall f \in L^1(T)$, and $\forall \varepsilon > 0$, there exist $K \in \mathbb{N}$, and a three layer feedforward neural network defined as $g(\mathbf{x}) = \sum\limits_{k=1}^{K} a_i \cdot \sigma(\langle \mathbf{w}_k,\ \mathbf{x}\rangle+\theta_k)$, which there are K hidden neurons, so that $\{\int_T |f(\mathbf{x}) - g(\mathbf{x})|\mathrm{d}\mu\} < \varepsilon$. By [9], if $\sigma : \mathbb{R} \longrightarrow \mathbb{R}$ is not a polynomial a.e., and $\forall a, b \in \mathbb{R} :\ a \le b$, $\int_a^b |\sigma(x)|\mathrm{d}x < +\infty$, then for each compact set $T \subset \mathbb{R}$, σ is $L^1(T)-$universal.

Theorem 4.21 *Let $(T, \mathcal{B}_T, \mu)$ be a finite measure space, and $\sigma : \mathbb{R} \longrightarrow \mathbb{R}$ be $L^1(T)-$universal. Then the following conclusions hold:*

(i) With integral norm $\mathcal{H}_0[\sigma]$ and consequently $\mathcal{H}[\sigma]$ is the universal approximator of $\mathcal{S}$;

(ii) Assume that $F : T \longrightarrow \mathcal{F}_0^c(\mathbb{R})$ is integrable bounded. Then with integral norm $\mathcal{H}[\sigma]$ can approximate F to arbitrary degree of accuracy.

Proof. (i) Let $S(\mathbf{x}) = \sum_{k=1}^{q} \widetilde{A}_k \cdot \chi_{T_k}(\mathbf{x})$ $(\mathbf{x} \in T)$. Without losing generality we let not all $|\widetilde{A}_1|, ..., |\widetilde{A}_q|$ are zero. For arbitrary $\varepsilon > 0$, since $\chi_{T_k} \in L^1(T)$ $(k = 1, ..., q)$, and σ is $L^1(T)-$universal, there are $p_k \in \mathbb{N}$, $v'_{1k}, ..., v'_{p_k k}$, $\theta'_{1k}, ..., \theta'_{p_k k} \in \mathbb{R}$, $\mathbf{u}'_k(1), ..., \mathbf{u}'_k(p_k) \in \mathbb{R}^d$, satisfying

$$\int_T \Big|\chi_{T_k}(\mathbf{x}) - \sum_{j=1}^{p_k} v'_{jk} \cdot \sigma\big(\langle \mathbf{x}, \mathbf{u}'_k(j)\rangle + \theta'_{jk}\big)\Big| \mathrm{d}\mu < \frac{\varepsilon}{2q \cdot \bigvee_{1\le k\le q} \{|\widetilde{A}_k|\}}.$$

Let $p = \sum_{k=1}^{q} p_k$. Then for $k = 2, ..., q$, write $\beta_k = \sum_{r=1}^{k-1} p_r$, $\beta_1 = 0$. For $k = 1, ..., q$, $j = 1, ..., p$, similarly with the proof of Theorem 4.12 we define

$$v_{jk} = \begin{cases} v'_{(j-\beta_k)k}, & \beta_k < j \le \beta_{k+1}, \\ 0, & \text{otherwise}; \end{cases} \qquad \theta_j = \begin{cases} \theta'_{(j-\beta_k)k}, & \beta_k < j \le \beta_{k+1}, \\ 0, & \text{otherwise}; \end{cases}$$

$$\mathbf{u}(j) = \begin{cases} \mathbf{u}'_k(j-\beta_k), & \beta_k < j \le \beta_{k+1}, \\ 0, & \text{otherwise}. \end{cases}$$

Let $F_N(\mathbf{x}) = \sum_{k=1}^{q} \widetilde{A}_k \cdot \sum_{j=1}^{p} v_{jk} \cdot \sigma\big(\langle \mathbf{x}, \mathbf{u}(j)\rangle + \theta_j\big)$. Easily we can show, $F_N \in \mathcal{H}_0[\sigma]$, furthermore

$$\forall k \in \{1, ..., q\}, \sum_{j=1}^{p} v_{jk} \cdot \sigma\big(\langle \mathbf{x}, \mathbf{u}(j)\rangle + \theta_j\big) = \sum_{j=1}^{p_k} v'_{jk} \cdot \sigma\big(\langle \mathbf{x}, \mathbf{u}'_k(j)\rangle + \theta'_{jk}\big).$$

By Lemma 4.7 it follows that

$$\begin{aligned} \Delta(F_N, S) &= \int_T D\Big(\sum_{k=1}^{q} \widetilde{A}_k \cdot \sum_{j=1}^{p} v_{jk} \cdot \sigma\big(\langle \mathbf{x}, \mathbf{u}(j)\rangle + \theta_j\big), \sum_{k=1}^{q} \widetilde{A}_k \cdot \chi_{T_k}(\mathbf{x})\Big) \mathrm{d}\mu \\ &\le \int_T \sum_{k=1}^{q} |\widetilde{A}_k| \cdot D\big(\chi_{T_k}(\mathbf{x}), \sum_{j=1}^{p} v_{jk} \cdot \sigma(\langle \mathbf{x}, \mathbf{u}(j)\rangle + \theta_j)\big) \mathrm{d}\mu \\ &\le q \cdot \bigvee_{1\le k\le q} |\widetilde{A}_k| \cdot \int_T \Big|\chi_{T_k}(\mathbf{x}) - \sum_{j=1}^{p_k} v'_{jk} \cdot \sigma\big(\langle \mathbf{x}, \mathbf{u}'_k(j)\rangle + \theta'_{jk}\big)\Big| \mathrm{d}\mu \\ &< \varepsilon. \end{aligned}$$

So with integral norm $\mathcal{H}_0[\sigma]$ is the universal approximator of $\mathcal{S}$.

(ii) It is the direct result of Theorem 4.19 and (i). Thus, the theorem is proved. □

By the crisp measure we introduce the integral norm of fuzzy valued functions, and study the universal approximation of regular FNN's to integrable bounded fuzzy valued functions in this metric. By Remark 4.5, the results presented here are general comparing with that of §4.4. Since the fuzzy valued measures and fuzzy function integrals are manifold [57], [64], undoubtedly, choosing different fuzzy integrals result in different research results. This section is only a beginning in the subjects related. There are a number of problems and questions to be solved. The integrable systems exist extensively in real and theoretic fields, so the research on universal approximations of FNN's with integral norm is very important and challenging, theoretically and practically.

References

[1] Abdukhalikov K. S. & Kim C., Fuzzy neurons and fuzzy multilinear mappings, *Proc. IEEE Internat. Conf. on Neural Networks,*1997: 543–547.

[2] Alefeld G. & Herzberger J., *Introduction to Interval Computations,* New York: Academic Press, 1983.

[3] Aliev R. A., Genetic algorithm-based learning of fuzzy neural network. Part 1: feed-forward fuzzy neural networks, *Fuzzy Sets and Systems,* 2001, **118**: 351–358.

[4] Bardossy A. Bogardi I. & Duckstein L., Fuzzy nonlinear regression analysis of dose-response relationships, *European J. Oper. Res.,* 1993, **66**(1): 46–51.

[5] Buckley J. J. & Hayashi Y., Can fuzzy neural nets approximate continuous fuzzy functions, *Fuzzy Sets and Systems,* 1994, **61**(1): 43–51.

[6] Buckley J. J. & Hayashi Y., Fuzzy neural networks: a survey, *Fuzzy Sets and Systems,* 1994, **66**: 1–13.

[7] Buckley J. J. & Hayashi Y., Direct fuzzification of neural networks, in *Proc. of 1st Asian Fuzzy Syst. Symp.,* 1993, **1**: 560–567.

[8] Buckley J. J. & Hayashi Y., Fuzzy neural nets and applications, *Fuzzy Sets and AI,* 1992, **1**(1): 11–41.

[9] Burton R. M., & Dehling H. G., Universal approximation in p-mean by neural networks, *Neural Networks,* 1998, **11**: 661–667.

[10] Carozza M. & Rampone S., Function approximation from noise data by an incremental RBF network, *Pattern Recognition,* 1999, *32*(12): 2081–2083.

[11] Charalambous C., Conjugate gradient algorithm for efficient training of artificial neural networks, *IEE Proc. -G,* 1992, **139**: 301-310.

[12] Chen D. S. & Jain R. C., A robust back propagation learning algorithm for function approximation, *IEEE Trans. on Neural Networks,* 1996, *7*(2): 467–479.

[13] Chen S. H., Operations of fuzzy numbers with step form membership function using function principle, *Information Sciences,* 1998, **108**: 149–155.

[14] Chen T. P., Approximation problem in system identification by neural network, *Science in China, Series A*(in Chinese), 1994, **37**(4): 414–422.

[15] Chen T. P., Chen H. & Liu R. W., Approximation capability in $C(\overline{\mathbb{R}}^n)$ by multilayer feedforward networks and related problems, *IEEE Trans. on Neural Networks,* 1995, **6**(1): 25–30.

[16] Chen T. P. & Liang Y., Neural networks calculate output capability of dynamical systems defined on whole space, *Science in China, Series E*(in Chinese), 1998,**28** (4): 338–349.

[17] Chung F. -L. & Lee T., Fuzzy competitive learning, *Neural Networks,* 1994, **7**(3): 539–552.

[18] Dai Y., Han J. & Liu G., et al, Convergence properties of nonlinear conjugate gradient algorithms, *SIAM J. on Optimization,* 1999, **10**: 345-358.

[19] Diamond P. & Kloeden P., Characterization of compact subsets of fuzzy sets, *Fuzzy Sets and Systems,* 1989, **29**: 341–348.

[20] Diamond P. & Kloeden P., *Metric Spaces of Fuzzy Sets,* Singapore: World Scientific Publishing, 1994.

[21] Dunyak J. & Wunsch D., A training technique for fuzzy number neural networks, *Proc. IEEE Internat. Conf. on Neural Network,* 1997: 533–536.

[22] Dunyak J. & Wunsch D., Training fuzzy number neural networks with alpha-cut refinements, *Proc. IEEE Internat. Conf. on Systems, Man, Cybernet.,* 1997: 189–194.

[23] Dunyak J. & Wunsch D., Fuzzy number neural networks, *Fuzzy Sets and Systems,* 1999, **108**(1): 49–58.

[24] Feuring T., Learning in fuzzy neural networks, *Proc. IEEE Internat. Conf. on Neural Networks,* 1996: 1061–1066.

[25] Feuring T., Buckley J. J. & Hayashi Y., Fuzzy neural nets can solve the overfitting problem, in *Proc. of Internat. Joint Conf. on Neural Networks,* 1999, **4**: 4197–4201.

[26] Feuring T., Buckley J. J. & Hayashi Y., A gradient descent learning algorithm for fuzzy neural networks, in *Proc. of IEEE Internat. Conf. on Fuzzy Systems,* 1998, **2**: 1136–1141.

[27] Feuring T. & Lippe W. M., The fuzzy neural network approximation lemma, *Fuzzy Sets and Systems,* 1999, **102**(2): 227–237.

[28] Feuring T. & Lippe W. M., A goodness-function for trained fuzzy neural networks, *Proc. IEEE Internat. Conf. on Neural Networks,* 1995: 2264–2269.

[29] Feuring T. & Lippe W. M., Error prediction for neural networks by fuzzification, *Internat. J. Intelligence Systems,* 1998, **13**(3): 495–502.

[30] Feuring T. & Lippe W. M., Output estimation of fuzzy neural networks, in *Proc. of 4th Internat. Conf. on Soft Computing,* 1996, **1**: 626–629.

[31] Goldberg D. E., *Genetic algorithms in search, optimization, and machine learning,* Addison-Wesley, Reading, MA, 1989.

[32] Gupta M. M. & Rao D. H., On the principle of fuzzy neural networks, *Fuzzy Sets and Systems,* 1994, **61**: 1–18.

[33] Haykin S., *Neural Networks, A Comprehensive Foundation,* New York: IEEE Press, 1993.

[34] Hayashi Y., Buckley J. J. & Czogala E., Fuzzy neural network with fuzzy signals and weights, *Internat. J. Intell. Systems,* 1993, **8**: 527–537.

[35] Hayashi Y., Buckley J. J. & Czogala E., Direct fuzzification of neural network fuzzified delta rule, in *Proc. of 2nd Internat. Conf. on Fuzzy Logic and Neural Networks,* 1992, **1**: 73–76.

[36] Hayashi Y., Buckley J. J. & Czogala E., Fuzzy neural network with fuzzy signals and weights, in *Proc. of Internat. Joint Conf. on Neural Networks,* 1992, **2**: 696–701.

[37] Ishibuchi H., Fujioka R. & Tanaka H., Neural networks that learn from fuzzy if–then rules, *IEEE Trans. on Fuzzy Systems,* 1993, **1**: 85–97.

[38] Ishibuchi H., Kwon K. & Tanaka H. A., Learning algorithm of fuzzy neural networks with triangular fuzzy weights, *Fuzzy sets and Systems*, 1995, **71**: 277–293.

[39] Ishibuchi H., Kwon K. & Tanaka H. A., Learning of fuzzy neural networks from fuzzy inputs and fuzzy targets, in *Proc. of 5th IFSA World Congress,* 1993, **1**: 147–150.

[40] Ishibuchi H., Morioka K. & Turksen I. B., Learning by fuzzified neural networks, *Internat. J. Approx.Reasoning,* 1995, **13**(4): 327–358.

[41] Ishibuchi H. & Nii M., Numerical analysis of the learning of fuzzified neural networks from fuzzy if–then rules, *Fuzzy Sets and Systems,* 2001, **120**: 281–307.

[42] Ishibuchi H. & Nii M., Fuzzy regression using asymmetric fuzzy coefficients and fuzzified neural networks, *Fuzzy Sets and Systems,* 2001, **119**: 273–290.

[43] Ishibuchi H. & Nii M., Fuzzification of input vectors for improving the generalization capability of neural networks, in *Proc. of IEEE Internat. Conf. on Fuzzy Systems,* 1998, **3**: 373–379.

[44] Ishibuchi H., Nii M. & Oh C. -H., Approximate realization of fuzzy mapping by regression models,neural networks and rule-based systems, *Proc. IEEE Internat. Conf. on Fuzzy Systems,* 1999, **2**: 939–944.

[45] Ishibuchi H., Nii M. & Tanaka K., Subdivision methods for decreasing excess fuzziness of fuzzy arithmetic in fuzzified neural networks, in *Proc. of NAFIPS'99,* 1999, **1**: 448–452.

[46] Jang J. S. R., Sun C. T. & Mizutani E. *Neuro-fuzzy and soft computing — a computational approach to learning and machine intelligence,* Englewood Cliffs, NJ: Prentice-Hall, 1997.

[47] Jiang Chuanhai, Approximation problems of neural networks, *Ann. of Math., Series A,*(in Chinese) 1998, **17**(3): 295–300.

[48] Krishnamraju P. V., Buckley J. J. & Reilly K. D., et al, Genetic learning algorithms for fuzzy neural nets, *Proc. IEEE Internat. Conf. on Fuzzy Systems,* 1994, **3**: 1969–1974.

[49] Lee K. -M., Kwak D. -H. & Wang H. -L., Tuning of fuzzy models by fuzzy neural networks, *Fuzzy Sets and Systems,* 1995, **76**: 47–61.

[50] Liu Puyin, Feedforward fuzzy neural networks can be the universal approximators for fuzzy valued functions, *Proc. 7th European Congress on Intelligent Techniques and Soft Computing,* Aachen, Germany, Sep. 13-16, 1999.

[51] Liu Puyin, Universal approximations of continuous fuzzy valued functions by multi-layer regular fuzzy neural networks, *Fuzzy Sets and Systems,* 2001, **119**(2): 313–320.

[52] Liu Puyin, Analysis of approximation of continuous fuzzy function by multivariate fuzzy polynomial, *Fuzzy Sets and Systems,* 2002, **127**(3): 299–313.

[53] Liu Puyin, On the approximation realization of fuzzy closure mapping by multilayer regular fuzzy neural network, *Multiple Valued Logic,* 2000, **5**: 463–480.

[54] Liu Puyin, Fuzzy valued Markov processes and their properties, *Fuzzy Sets and Systems,* 1997, **91**: 87–96.

[55] Liu Puyin, Analyses of regular fuzzy neural networks for approximation capability, *Fuzzy Sets and Systems,* 2000, **114**(3): 329–338.

[56] Liu Puyin, Fuzzy valued functions approximated arbitrarily close by a class of regular fuzzy neural networks, *J. Fuzzy Mathematics,* 1999, **7**(4): 826–837.

[57] Liu Puyin, Approximation analyses for fuzzy valued functions in $L_1(\mu)$ by regular fuzzy neural networks, *J. Electron. Sci.*, 2000, **17**(2): 132–138.

[58] Liu Puyin, Approximation capabilities of multilayer feedforward regular fuzzy neural network, *Applied Math.,A Journal of Chinese University,* 2001, **16**(1): 23-35.

[59] Liu Puyin, Regular fuzzy neural network as universal approximator of fuzzy valued function, *Control and Decision*(in Chinese), 2003, **18**(1): 8–13.

[60] Liu Puyin & Li Hongxing, Efficient learning algorithms for three-layer feedforward regular fuzzy neural networks, *IEEE Trans. on Neural Networks.*, 2004, **15**(2).

[61] Liu Puyin & Li Hongxing, Approximation analysis for feedforward regular fuzzy neural networks with two hidden layers, *Fuzzy Sets and Systems.,* 2004 (in Press).

[62] Liu Puyin & Wang Hao, Research on approximation capability of regular fuzzy neural network to continuous fuzzy function, *Science in China, Series E,* 1999, **41**(2): 143–151.

[63] Liu Puyin & Wang Huaxing, Universal approximation of regular fuzzy networks to a class of continuous fuzzy valued functions, *Acta Electron. Sinica,* 1997, **25**(11): 48–52.

[64] Liu Puyin & Wu Mengda, *Fuzzy theory and applications,* Changsha: Press of National University if Defence Tech., 1998.

[65] Liu Puyin & Zhang Hanjiang, Research on fuzzy neural network theory—a survey, *Fuzzy Systems and Math.*, 1998, **10**(1): 77-87.

[66] Liu Puyin & Zhang Weiming, Mathematical problems in research on reliability of communication networks, *J. China Institute of Comm.*(in Chinese), 2000, **21** (10): 50-57.

[67] Luo F. -L. & Unbehauen R., *Applied Neural Networks for Signal Processing,* Cambridge, New York, Melbourne: Cambridge University Press, 1998.

[68] Maniezzo V., Genetic evolution of the topology and weight distribution of neural networks, *IEEE Trans. on Neural Networks,* 1994, **5**: 39–53.

[69] Mendel K. & Liang Q., Type-2 fuzzy logic systems, *IEEE Trans. on Fuzzy Systems,* 1999 **7**(6): 788–800.

[70] Mitchell M., *An Introduction to Genetic Algorithms,* Massachusetts•London: The MIT Press, 1996.

[71] Miyazaki A., Kwan K. & Ishibuchi H., et al., Fuzzy regression analysis by fuzzy neural networks and its application, *IEEE Trans. on Fuzzy Systems,* 1994, **2**:

[72] Narendra K. S. & Parthasarathy K., Identification and control of dynamical systems using neural networks, *IEEE Trans. on Neural Networks,* 1990, **1**: 4–27.

[73] Nguyen H. T., A note on the extension principle for fuzzy sets, *J. Math. Anal. Appl.,* 1978, **64**: 369–380.

[74] Pal S. K. & Mitra S., Multilayer perceptron, fuzzy sets, and classification, *IEEE Trans. on Neural Networks,* 1992, **3**(3): 683–697.

[75] Park S. & Han T., Iterative inversion of fuzzified neural networks, *IEEE Trans. on Fuzzy Systems,* 2000, **8**(3): 268–280.

[76] Parlos A. G., Fernandez B. & Atiya A. F., et al, An accelerated learning algorithm for multilayer perceptron networks, *IEEE Trans. on Neural Networks,* 1994, **5**: 493–497.

[77] Powell M. J. D., Some convergence properties of the conjugate gradient method, *Math. Prog.,* 1976, **11**(1): 42–49.

[78] Rudolph G., Convergence analysis of canonical genetic algorithms, *IEEE Trans.on Neural Networks,* 1994, **5**: 96–101.

[79] Scarselli F. & Tosi A. C., Universal approximation using feedforward neural networks: a survey of some existing methods,and some new results, *Neural Networks,* 1998, **11**(1): 15–37.

[80] Yuan Y. X., *Numerical method for nonlinear programming,* Shanghai: Press of Shanghai Sci. & Tech., 1993.

[81] Zhang X. H. Tan S. H. & Huang C. C., et al, An efficient computational algorithm for min-max operations,*Fuzzy Sets and Systems*, 1999, **104**: 297–304.

[82] Zhang X. H., Huang C. C. & Tan S. H., et al, The min-max function differentiation and training of fuzzy neural networks,*IEEE Trans. on Neural Networks,* 1996, **7**: 1139–1150.

[83] Zadeh L. A. & Yager R. R., *Fuzzy sets, neural networks and soft computing,* New York: Van Nostrand Reinhold, 1994.

[84] A special issue on fuzzy arithmetic, *Fuzzy Sets and Systems,* 1997, **91**(3).

CHAPTER V

Polygonal Fuzzy Neural Networks

Using the results obtained in Chapter IV, we enunciate that, to ensure the regular FNN's to constitute universal approximators to the continuous fuzzy function class $\mathcal{C}_F$, each fuzzy function in $\mathcal{C}_F$ must be a closure fuzzy mapping besides being monotone and increasing. However, in practice it is difficult to discriminate a given fuzzy function whether is a closure fuzzy mapping, which results in much inconvenience, undoubtedly for application of FNN's. To overcome the drawback this chapter will introduce a novel FNN—polygonal FNN, whose topological architecture inherits from one of a regular FNN, and internal operations are based on a family of simplified extension principle [31, 37, 38].

As a generalization of triangular or trapezoidal fuzzy numbers, the polygonal fuzzy numbers presented in this chapter can approximate a class of bounded fuzzy numbers, with arbitrary degree of accuracy. So compared with regular FNN's, polygonal FNN's have the following advantages: First, the I/O relationship structures are more succinct, and hence it is easier to analysis the basic properties related, such as approximation capability, learning algorithm and so on; Second, more strong approximation capability can be ensured, for polygonal FNN's can be universal approximators to monotone increasing and continuous fuzzy functions, and thus wider application area for polygonal FNN's can also be guaranteed; Finally, we can express the I/O relationship of a polygonal FNN as the combination of I/O relationships corresponding to some crisp neural networks.

To realize above objectives let us now show some uniformity results about universal approximation related to crisp feedforward neural networks. Then we focus on the topics related to polygonal FNN's, including system structure analysis, representation of I/O relationships, and universal approximation and so on. By designing a fast convergent learning algorithm and illustrating some simulation examples, it is demonstrated that polygonal FNN's are easy to train and apply.

§5.1 Uniformity analysis of feedforward networks

As a topic closely related to approximation of neural networks, the uniformity analysis for system universal approximation has recently attracted much attention [8, 27, 50, 51, 54]. The uniformity of universal approximation of neural networks means that for a given accuracy $\varepsilon > 0$, the neural networks can provide with a uniform representations under the sense '$\approx_\varepsilon$', to a family of continuous functions. That is, if σ is a transfer function, $I \subset \mathbb{R}^d$ is a compact set, for each compact family $V \subset C(I)$, of continuous functions, there is $q \in \mathbb{N}$, so that $\forall f \in V$, there exist $\phi_1(f), ..., \phi_q(f) \in \mathbb{R}$, satisfying $\left|f(\mathbf{x}) - \sum_{k=1}^{q} \phi_k(f) \cdot N_k^m(\mathbf{x})\right| < \varepsilon$ $(\mathbf{x} \in I)$, where $N_k^m(\cdot)$ is a function determined by a m layer feedforward neural network, which is only dependent of V, and independent of each function f in V. For example, if $m = 3$, there exist $p \in \mathbb{N}$, $v_{kj} \in \mathbb{R}$, $\theta_j \in \mathbb{R}$, and $\mathbf{u}(j) \in \mathbb{R}^d$ $(k = 1, ..., q;\ j = 1, ..., p)$, being only dependent of V, so that

$$\forall \mathbf{x} \in I,\ \forall f \in V,\ \left|f(\mathbf{x}) - \sum_{k=1}^{q} \phi_k(f) \cdot \sum_{j=1}^{p} v_{kj}\sigma\big(\langle \mathbf{u}(j), \mathbf{x}\rangle + \theta_j\big)\right| < \varepsilon,$$

where, $\phi_k(\cdot)$ $(k = 1, ..., q)$ is a continuous functional. In [9, 27], the systematic achievements related to universal approximation of neural networks with the integral norm $L^p(I)$ are presented. We can utilize these results to deal with uniformity analysis for universal approximation, consequently to study the approximation capability of neural networks within a general framework. Thus, we may make an essential exposition to neural system approximation, which is of very significance in theory, as the approximate representations of nonlinear operators by neural networks [8, 27], and application as system identification [9, 52], system modelling [52], signal processing and so on. This section will mainly aims at the following important problems:

(i) Whether can we choose the family $\{\phi_k(\cdot) | k = 1, ..., q\}$ of continuous functionals, so that $\{\phi_1(f), ..., \phi_q(f)\}$ or at least a part of this set is independent of f, and $\phi_k(f)$ can be calculated with a unifying computation framework?

(ii) Besides continuity of the functionals, whether are some prescribed conditions for $\phi_k(f)$, such as $\phi_k(f)$ being increasing with respect to f, i.e. $f \le g, \Longrightarrow \phi_k(f) \le \phi_k(g)$, be ensured?

In this section we take the four layer, i.e. $m = 3$ and three layer, i.e. $m = 2$, feedforward neural networks, respectively, as our research objects to deal with above problems, systematically. For given compact set $I \subset \mathbb{R}^d$, define the maximum norm $\|\cdot\|_{C(I)}$ as follows: $\|f\|_{C(I)} = \bigvee_{\mathbf{x} \in I} \{|f(\mathbf{x})|\}$.

5.1.1 Universal approximation of four-layer network

At first let us aim at the uniformity analysis for universal approximation of four layer neural networks. To this end we introduce a novel function — uniform Tauber-Wiener function.

Definition 5.1 Let $\sigma : \mathbb{R} \longrightarrow \mathbb{R}$ be a transfer function, we call σ a uniform Tauber-Wiener function, if for each compact set $V \subset C(I)$, $\forall \varepsilon > 0$, there are

$p, q \in \mathbb{N}$ and $v_{jk}, \theta_j \in \mathbb{R}$, $\mathbf{u}(j) \in \mathbb{R}^d$ ($k = 1, ..., q$; $j = 1, ..., p$), so that $\forall f \in V$, it follows that there exist q real numbers, $\phi_1(f), ..., \phi_q(f)$ satisfying

$$\Big| f(x) - \sum_{k=1}^{q} \phi_k(f) \cdot \sum_{j=1}^{p} v_{jk}\sigma\big(\langle \mathbf{u}(j), \mathbf{x}\rangle + \theta_j\big)\Big| < \varepsilon \ (\mathbf{x} \in I).$$

If $f \in C(I)$, $\delta > 0$, by $U(f, \delta)$ we denote a $\delta-$neighborhood of f in $C(I)$, that is

$$U(f, \delta) = \big\{ g \in C(I) \big| \ \|g - f\|_{C(I)} < \delta \big\}.$$

In the following we develop a constructive proof to establish sufficient conditions for connection weights of a four layer feedforward neural networks, so that the given approximating accuracy is ensured. Moreover, a learning algorithm easy to operate is designed to realize the according approximating procedure. To this end, we firstly establish an approximate expression for $f \equiv 1$ by a three layer neural network as, $\sum\limits_{i=1}^{n} c_i \cdot \sigma(u_i x + \theta_i)$ $(x \in \mathbb{R})$.

Theorem 5.1 *Let the transfer function σ be a generalized sigmoidal function, i.e.* $\lim\limits_{x \to +\infty} \sigma(x) = 1$, $\lim\limits_{x \to -\infty} \sigma(x) = 0$. *Moreover, σ is bounded. For arbitrary $\varepsilon > 0$, and $M > 1$, $n \in \mathbb{N}$, the following conditions hold:*

(i) $\forall x > M$, $\big|\sigma(x) - 1\big| < \varepsilon/3$, $\forall x < -M$, $\big|\sigma(x)\big| < \varepsilon/3$;

(ii) If the constant K satisfying: $\forall x \in \mathbb{R}$, $\big|\sigma(x)\big| \le K$, *we have,* $(2K+1)/n < \varepsilon/3$.

Then there exist $\beta > 0$, and $\lambda_1, ..., \lambda_n \in \mathbb{R}$, so that

$$\forall x \in \mathbb{R}, \ \Big| \frac{1}{n} \sum_{i=1}^{n} \big(\sigma(\beta x + \lambda_i) + \sigma(-\beta x - \lambda_i) \big) - 1 \Big| < \varepsilon \tag{5.1}$$

Proof. Partition the interval $[-M, M]$ into n equal parts: $-M = x_0 < x_1 < \cdots < x_{n-1} < x_n = M$, and $x_i = -M + (2Mi)/n$ for $i = 0, 1, ..., n$. Choose $\beta > 0$, satisfying: $\beta/n > 1$. Let $t_i = (x_i + x_{i-1})/2$ $(i = 1, ..., n)$, and set

$$\forall x \in \mathbb{R}, \ S(x) = \frac{1}{n} \sum_{i=1}^{n} \big[\sigma\big(\beta(x - t_i)\big) + \sigma\big(-\beta(x - t_i)\big) \big].$$

Let us show $\forall x \in \mathbb{R}$, $|S(x) - 1| < \varepsilon$. In fact, if $x < -M$, $\forall i = 1, ..., n$, $|x - t_i| \ge |x_0 - t_1| = M/n$, we have, $\beta(x - t_i) \le \beta M/n < -M$, and $-\beta(x - t_i) > M$. By assumption (i) we imply, $\big|\sigma\big(-\beta(x - t_i)\big) - 1\big| < \varepsilon/3$, $\big|\sigma\big(\beta(x - t_i)\big)\big| < \varepsilon/3$. Therefore

$$\begin{aligned} |S(x) - 1| &= \Big| \frac{1}{n} \sum_{i=1}^{n} \big[\big(\sigma(-\beta(x - t_i)) - 1 \big) + \sigma(\beta(x - t_i)) \big] \Big| \\ &\le \frac{1}{n} \sum_{i=1}^{n} \big[\big| \sigma\big(-\beta(x - t_i)\big) - 1 \big| + \big| \sigma\big(\beta(x - t_i)\big) \big| \big] < \frac{1}{n} \sum_{i=1}^{n} \big(\frac{\varepsilon}{3} + \frac{\varepsilon}{3} \big) < \varepsilon. \end{aligned}$$

With the same reason, $\forall x > M,\ |S(x)-1| < \varepsilon$. If $x \in [-M, M]$, let $k \in \{1,...,n\}$, $x \in [x_{k-1}, x_k]$. For $i \in \{0,1,...,k-1\}$, it follows that $x - t_i \geq x_{k-1} - t_i \geq (x_i - x_{i-1})/2 = M/n$. So $\beta(x-t_i) \geq \beta M/n > M$, $-\beta(x-t_i) < -M$. Similarly, for $i \in \{k+1,...,n\}$, $x - t_i \leq -M/n$. Hence $\beta(x-t_i) < -M$, $-\beta(x-t_i) > M$. By conditions (i) (ii) we obtain

$$
\begin{aligned}
|S(x)\ -1| &= \Big|\frac{1}{n}\sum_{i=1}^{n}\big[\sigma\big(-\beta(x-t_i)\big)-1+\sigma\big(\beta(x-t_i)\big)\big]\Big| \\
&\leq \frac{1}{n}\sum_{i=1}^{k-1}\big[\big|\sigma\big(\beta(x-t_i)\big)-1\big|+\big|\sigma\big(-\beta(x-t_i)\big)\big|\big]+\frac{1}{n}\big[\big|\sigma\big(-\beta(x-t_k)\big)\big|+ \\
&+\big|\sigma\big(\beta(x-t_k)\big)\big|+1\big]+\frac{1}{n}\sum_{i=k+1}^{n}\big[\big|\sigma\big(\beta(x-t_i)\big)\big|+\big|\sigma\big(-\beta(x-t_i)\big)-1\big|\big] \\
&< \frac{1}{n}\sum_{i=1}^{k-1}\Big(\frac{\varepsilon}{3}+\frac{\varepsilon}{3}\Big)+\frac{1}{n}\sum_{i=k+1}^{n}\Big(\frac{\varepsilon}{3}+\frac{\varepsilon}{3}\Big)+\frac{2K+1}{n} < \frac{\varepsilon}{3}+\frac{\varepsilon}{3}+\frac{\varepsilon}{3}=\varepsilon.
\end{aligned}
\tag{5.2}
$$

In summary, $\forall x \in \mathbb{R}$, $|S(x)-1| < \varepsilon$. By letting $\lambda_i = -\beta t_i$ $(i = 1,...,n)$, we imply the theorem. □

Let us now proceed to study the approximation capability of a four layer feedforward neural network by using the Bernstein polynomial as a bridge. Also some order conditions for connection weights are established. Give $m \in \mathbb{N}$, and a multivariate function $f : [0,1]^d \longrightarrow \mathbb{R}$. The multivariate Bernstein $B_m(\cdot)$ defined in (4.37) is firstly utilized to show the following lemma.

Lemma 5.1 *Let $V \subset C([0,1]^d)$ be a compact set, and arbitrarily give $\varepsilon > 0$. Then there is $n \in \mathbb{N}$, so that $\forall m \in \mathbb{N}: m > n$, if partition $[0, 1]$ into m equal parts: $0 < 1/m < \cdots < (m-1)/m < 1$, for arbitrary $f \in V$, we have*

$$\forall \mathbf{x} = (x_1, ..., x_d) \in [0,1]^d,\ \Big|f(\mathbf{x}) - \sum_{i_1,...,i_d=0}^{m} f\Big(\frac{i_1}{m},...,\frac{i_d}{m}\Big)K_{m;i_1...i_d}(\mathbf{x})\Big| < \varepsilon.$$

Proof. Since $V \subset C([0,1]^d)$ is a compact set, there exist $\gamma \in \mathbb{N}$, and $f_1,...,f_\gamma \in V$, so that $V \subset \bigcup_{j=1}^{\gamma} U\big(f_j,\ \varepsilon/3\big)$, moreover, $f_1,...,f_\gamma$ are uniformly continuous on $[0, 1]^d$. So there is $\delta > 0$, satisfying: $\forall \mathbf{x}^1 = (x_1^1,...,x_d^1)$, $\mathbf{x}^2 = (x_1^2,...,x_d^2) \in [0,1]^d$, $|x_i^1 - x_i^2| < \delta$ $(i = 1,...,d)$, $\Longrightarrow |f_j(\mathbf{x}^1) - f_j(\mathbf{x}^2)| < \varepsilon/4$ $(j = 1,...,\gamma)$. Moreover, there is $M > 0$, so that $\forall j \in \{1,...,\gamma\}$, $\forall \mathbf{x} \in [0,1]^d$, $|f_j(\mathbf{x})| \leq M$. Give $m \in \mathbb{N}: 1/m < \delta$. Partition $[0, 1]$ into m parts: $0 < 1/m < \cdots < (m-1)/m < 1$. Then for each $j = 1,...,\gamma$ and $\forall i_1,...,i_d \in \{1,...,m\}$, we have

$$\forall \mathbf{x} \in \Big[\frac{i_1-1}{m},\ \frac{i_1}{m}\Big] \times \cdots \times \Big[\frac{i_d-1}{m},\ \frac{i_d}{m}\Big],\ \Big|f_j(\mathbf{x}) - f_j\Big(\frac{i_1}{m},...,\frac{i_d}{m}\Big)\Big| < \frac{\varepsilon}{3}. \tag{5.3}$$

By (4.38) (4.40) and (5.3), $\forall j = 1, ..., \gamma$, $\forall \mathbf{x} = (x_1, ..., x_d) \in [0, 1]^d$, we can show the following facts:

$$\begin{aligned}
&\Big|f_j(\mathbf{x}) - \sum_{i_1,...,i_d=0}^{m} f_j\big(\frac{i_1}{m}, ..., \frac{i_d}{m}\big) K_{m;i_1...i_d}(\mathbf{x})\Big| \\
&= \Big|\sum_{i_1,...,i_d=0}^{m} \Big(f(\mathbf{x}) - f_j\Big(\frac{i_1}{m}, ..., \frac{i_d}{m}\Big)\Big) K_{m;i_1...i_d}(\mathbf{x})\Big| \\
&\leq \sum_{i_1,...,i_d:\|\mathbf{x}-(\frac{i_1}{m},...,\frac{i_d}{m})\|<\delta} K_{m;i_1,...,i_d}(\mathbf{x}) \Big|f_j\Big(\frac{i_1}{m}, ..., \frac{i_d}{m}\Big) - f_j(\mathbf{x})\Big| + \\
&\quad + \sum_{i_1,...,i_d:\|\mathbf{x}-(\frac{i_1}{m},...,\frac{i_d}{m})\|\geq\delta} K_{m;i_1,...,i_d}(\mathbf{x}) \Big|f_j\Big(\frac{i_1}{m}, ..., \frac{i_d}{m}\Big) - f_j(\mathbf{x})\Big| \qquad (5.4) \\
&< \frac{\varepsilon}{4} + M \cdot \sum_{i_1,...,i_d:\|\mathbf{x}-(\frac{i_1}{m},...,\frac{i_d}{m})\|\geq\delta} K_{m;i_1,...,i_d}(\mathbf{x}) \Big[\frac{\|\mathbf{x} - (\frac{i_1}{m}, ..., \frac{i_d}{m})\|}{\delta}\Big]^2 \\
&\leq \frac{\varepsilon}{4} + \frac{2M}{m^2\delta^2} \cdot \sum_{k=1}^{d} \Big(\sum_{i_k=0}^{m} \binom{m}{i_k} x_k^{i_k} (1 - x_k)^{m-i_k} (mx_k - i_k)^2\Big) \\
&\leq \frac{\varepsilon}{4} + \frac{2Md}{m^2\delta^2} \cdot \frac{m}{4} = \frac{\varepsilon}{4} + \frac{Md}{2m\delta^2}.
\end{aligned}$$

Choose $n \in \mathbb{N}: n \geq \max\big(1/\delta, (6Md)/(\varepsilon\delta^2)\big)$. Thus, $\forall m > n$, $(Md)/(2m\delta^2) < \varepsilon/12$. Therefore

$$\Big|f_j(\mathbf{x}) - \sum_{i_1,...,i_d=0}^{m} f_j\big(\frac{i_1}{m}, ..., \frac{i_d}{m}\big) K_{m;i_1...i_d}(\mathbf{x})\Big| < \frac{\varepsilon}{3}.$$

For arbitrary $f \in V$, let $j_0 \in \{1, ..., q\}$, such that $f \in U\big(f_{j_0}, \varepsilon/3\big)$. So $\forall \mathbf{x} \in [0, 1]^d$, $|f(\mathbf{x}) - f_{j_0}(\mathbf{x})| < \varepsilon/3$. By (5.2) (5.3) and (5.4), $\forall \mathbf{x} = (x_1, ..., x_d) \in [0, 1]^d$, it follows that

$$\begin{aligned}
&\Big|f(\mathbf{x}) - \sum_{i_1,...,i_d=0}^{m} f\big(\frac{i_1}{m}, ..., \frac{i_d}{m}\big) K_{m;i_1,...,i_m}(\mathbf{x})\Big| \\
&\leq |f(\mathbf{x}) - f_{j_0}(\mathbf{x})| + \Big|f_{j_0}(\mathbf{x}) - \sum_{i_1,...,i_d=0}^{m} f_{j_0}\big(\frac{i_1}{m}, ..., \frac{i_d}{m}\big) K_{m;i_1,...,i_m}(\mathbf{x})\Big| + \\
&\quad + \sum_{i_1,...,i_d=0}^{m} \Big|f_{j_0}\big(\frac{i_1}{m}, ..., \frac{i_d}{m}\big) - f\big(\frac{i_1}{m}, ..., \frac{i_d}{m}\big)\Big| K_{m;i_1...i_d}(\mathbf{x}) \\
&< \frac{\varepsilon}{3} + \frac{\varepsilon}{3} + \frac{\varepsilon}{3} = \varepsilon.
\end{aligned}$$

Thus the lemma is proved. □

Let $I \subset \mathbb{R}^d$ be a compact set, we call the functional $\phi : C(I) \longrightarrow \mathbb{R}$ to be order–preserved (also see [30, 33]), if

$$\forall f,\ g \in C(I),\ f \le g \Longrightarrow \phi(f) \le \phi(g),$$

where $f \le g$ means that $\forall \mathbf{x} \in I,\ f(\mathbf{x}) \le g(\mathbf{x})$.

Theorem 5.2 *Let σ be a Tauber-Wiener function. Then σ is a uniform Tauber-Wiener function, that is, for arbitrary compact set $V \subset C([0,1]^d)$, and $\forall\, \varepsilon > 0$, there are $p,\ q \in \mathbb{N}$, $\mathbf{u}(j) \in \mathbb{R}^d$, $v_{jk},\ \theta_j \in \mathbb{R}$ $(k = 1, ..., q;\ j = 1, ..., p)$, and continuous functionals $\phi_1, ..., \phi_q :\ C([0,1]^d) \longrightarrow \mathbb{R}$, such that for each $f \in V$, we have, $\|f - H_f\|_{C([0,1]^d)} < \varepsilon$, where*

$$H_f(\mathbf{x}) = \sum_{k=1}^{q} \phi_k(f)\Big(\sum_{j=1}^{p} v_{jk}\sigma\big(\langle \mathbf{u}(j), \mathbf{x}\rangle + \theta_j\big)\Big).$$

Moreover, all functionals related are order–preserved, i.e. $\forall f,\ g \in V,\ f \ge g, \Longrightarrow \phi_k(f) \ge \phi_k(g)$ $(k = 1, ..., q)$.

Proof. Since $V \subset C([0,1]^d)$ is a compact set, for arbitrary $\varepsilon > 0$, by Lemma 5.1, there is $m \in \mathbb{N}$, so that $\forall\, \mathbf{x} = (x_1, ..., x_d) \in [0,\ 1]^d$, if let

$$B_m(f;\mathbf{x}) = \sum_{i_1,...,i_d=0}^{m} f\Big(\frac{i_1}{m}, ..., \frac{i_d}{m}\Big) K_{m;i_1,...,i_d}(x_1, ..., x_d), \tag{5.5}$$

we can obtain the following facts:

$$\forall \mathbf{x} \in [0,\ 1]^d,\ \forall f \in V,\ \big|B_m(f;\mathbf{x}) - f(\mathbf{x})\big| < \frac{\varepsilon}{3}. \tag{5.6}$$

It is no harm to assume $M_f \triangleq \bigvee_{i_1,...,i_d=0}^{m} \big\{\big|f\big(i_1/m, ..., i_d/m\big)\big|\big\} > 0$. For arbitrary $i_1, ..., i_d \in \{0, 1, ..., m\}$, $K_{m;i_1...i_d}(\mathbf{x})$ is a multi-variate polynomial defined on $[0,1]^d$, so it is continuous on $[0,1]^d$. By the assumption, σ is a Tauber-Wiener function, there are $q(i_1, ..., i_d) \in \mathbb{N}$, and $\mathbf{u}_i(i_1, ..., i_d) \in \mathbb{R}^d$, $\theta_i(i_1, ..., i_d) \in \mathbb{R}$, $v_i(i_1, ..., i_d) \in \mathbb{R}$ $(i = 1, ..., q(i_1, ..., i_d))$, so that $\forall\, \mathbf{x} = (x_1, ..., x_d) \in [0,1]^d$, we have $\big|K_{m;i_1...i_d}(\mathbf{x}) - N_{i_1...i_d}(\mathbf{x})\big| < \varepsilon/(3M(m+1)^d)$, where

$$N_{i_1...i_d}(\mathbf{x}) = \sum_{i=1}^{q(i_1,...,i_d)} v_i(i_1, ..., i_d)\sigma\big(\langle \mathbf{u}_i(i_1, ..., i_d),\ \mathbf{x}\rangle + \theta_i(i_1, ..., i_d)\big).$$

Therefore using (5.5), we can show

$$\begin{aligned}
&\Big|B_m(f;\mathbf{x}) - \sum_{i_1,...,i_d=0}^{m} f\Big(\frac{i_1}{m}, ..., \frac{i_d}{m}\Big)\cdot N_{i_1...i_d}(\mathbf{x})\Big| \\
&\le \sum_{i_1,...,i_d=0}^{m} \Big|f\Big(\frac{i_1}{m}, ..., \frac{i_d}{m}\Big)\Big| \cdot \Big|K_{m;i_1...i_d}(x_1, ..., x_d) \cdot N_{i_1...i_d}(\mathbf{x})\Big| \qquad (5.7)\\
&< \sum_{i_1,...,i_d=0}^{m} M \cdot \frac{\varepsilon}{3M(m+1)^d} = \frac{\varepsilon}{3}.
\end{aligned}$$

Let $\phi_{i_1,...,i_d}: C([0,1]^d) \longrightarrow \mathbb{R}$, so that for each $f \in C([0,1]^d)$, $\phi_{i_1,...,i_d}(f) = f(i_1/m,...,i_d/m)(i_1,...,i_d = 0,1,...,m)$. Obviously $\phi_{i_1,...,i_d}$ is a order-preserved and continuous functional. The index set $\{(i_1,...,i_d)|i_1,...,i_d = 0,1,...,m\}$ is re-arrayed with the following order:

$$\big\{(0,...,0),...,(0,...,m),\ (1,0,...,0),...,(1,0,...,0,m),...,\\ (m,...,m,0),...,(m,....,m)\big\}.$$

Denote $q \triangleq (m+1)^d$, and $p \triangleq \sum_{i_1,...,i_d=0}^{m} q(i_1,...,i_d)$. If $j \in \{1,...,p\}, k \in \{1,...,q\}$, define

$$v_{jk} = \begin{cases} v_j(0,...,0), & 1 \le j \le q(0,...,0),\ k \leftrightarrow (0,...,0), \\ \cdots\cdots\cdots, & \\ v_j(i_1,...,i_d), & \sum_{i_1'=0}^{i_1}\cdots\sum_{i_{d-1}'=0}^{i_{d-1}}\sum_{i_d'=0}^{i_d-1} q(i_1',...,i_d') \le j \le \sum_{i_1'=0}^{i_1}\cdots\sum_{i_d'=0}^{i_d} q(i_1',...,i_d'), \\ & k \leftrightarrow (i_1,...,i_d), \\ \cdots\cdots\cdots, & \\ v_j(m,...,m), & \sum_{i_1,...,i_{d-1}=0}^{m}\sum_{i_d=0}^{m-1} q(i_1,...,i_d) \le j \le \sum_{i_1,...,i_d=0}^{m} q(i_1,...,i_d), \\ & k \leftrightarrow (m,...,m), \end{cases} \tag{5.8}$$

where $k \leftrightarrow (i_1,...,i_d)$ means the ordinal number of $(i_1,...,i_d)$ being k after the re-array procedure. Using the sets

$$\{\mathbf{u}_i(i_1,...,i_d)|i = 1,...,q(i_1,...,i_d)\},\ \{\theta_i(i_1,...,i_d)|i = 1,...,q(i_1,...,i_d)\},$$

Similarly with (5.8) we can define the vector family $\{\mathbf{u}(j)|j = 1,...,p\} \subset \mathbb{R}^d$ and the real number family $\{\theta_j|j = 1,...,p\} \subset \mathbb{R}$. Furthermore, we can prove that, $\{\phi_{i_1,...,i_d}|i_1,...,i_d = 0,1,...,m\} = \{\phi_k|k = 1,...,q\}$, and

$$\sum_{i_1,...,i_d=0}^{m} \phi_{i_1,...,i_d}(f) \cdot \sum_{i=1}^{q(i_1,...,i_d)} v_i(i_1,...,i_d)\sigma\big(\langle \mathbf{u}_i(i_1,...,i_d), \mathbf{x}\rangle + \theta_i(i_1,...,i_d)\big)$$
$$= \sum_{k=1}^{q} \phi_k(f) \cdot \sum_{j=1}^{p} v_{jk}\sigma\big(\langle \mathbf{u}(j), \mathbf{x}\rangle + \theta_j\big) \triangleq H_f(\mathbf{x}).$$

Thus, using (5.6) (5.7) $\forall \mathbf{x} \in [0,1]^d$, we can obtain

$$|f(\mathbf{x}) - H_f(\mathbf{x})| \le |f(\mathbf{x}) - B_m(f;\mathbf{x})| + |B_m(f;\mathbf{x}) - H_f(\mathbf{x})| < \frac{\varepsilon}{3} + \frac{\varepsilon}{3} = \frac{2}{3}\varepsilon.$$

That is, $\|f - H_f\|_{C([0,1]^d)} \le 2\varepsilon/3 < \varepsilon$. Moreover, $\phi_1,...,\phi_q$ are order-preserved and continuous functionals, which proves the theorem. □

Let $V \subset C(\mathbb{R}^d)$ be a given compact set. For each compact set $I \subset \mathbb{R}^d$, we restrict each function in V to define on I, then $V \subset C(I)$ is a compact set.

Similarly, when substituting I for $[0,\ 1]^d$, we can obtain the conclusion similar with Theorem 5.2. In the following, we utilize an example to demonstrate the realization procedure of Theorem 5.2, and develop a learning algorithm for the functional $\phi_k(k=1,...,q)$, connection weights $v_{jk},\ \theta_j \in \mathbb{R}$, and weight vector $\mathbf{u}(j) \in \mathbb{R}^d$.

Example 5.1 Let $d=1,\ I=[0,\ 1]$, and σ be bounded and increasing, $\lim_{x\to+\infty} \sigma(x)=1,\ \lim_{x\to-\infty} \sigma(x)=0$. Then σ is Tauber-Wiener function [31]. Choose $V=\{\sin(x/n)+(n-1)/n | n\in\mathbb{N}\}\cup\{1\} \subset C([0,1])$. And let $g_n(x)=\sin(x/n)+(n-1)/n\ (x\in[0,1])$, then we can show

(i) $g_n(x)\le g_{n+1}(x)\ (x\in[0,\ 1])$; (ii) g_n converges uniformly to $1\ (n\to+\infty)$.

It is easy to show that V is a compact set. Give arbitrarily $\varepsilon>0$. Considering

$$\forall n\in\mathbb{N},\ |g_n(x_1)-g_n(x_2)| = \left|\sin\frac{x_1}{n}-\sin\frac{x_2}{n}\right| \le \left|\frac{x_1-x_2}{n}\right|\ (x_1,x_2\in[0,1]).$$

we choose $m=\mathrm{Int}(3/\varepsilon)+1$, and partition $[0,\ 1]$ into m equal parts: $0<1/m<\cdots<(m-1)/m<1$. Define one-dimensional Bernstein polynomial as follows:

$$B_m(g_n;x)=\sum_{k=0}^{m} g_n\Big(\frac{k}{m}\Big)\binom{m}{k}x^k(1-x)^{m-k} \triangleq \sum_{k=0}^{m} g_n\Big(\frac{k}{m}\Big)K_{m;k}(x). \tag{5.9}$$

Choose $M>0$, so that $|\sigma(t)-1|<\varepsilon/3\ (t>M)$, $|\sigma(t)|<\varepsilon/3\ (t<-M)$. And let $\beta>0:\ \beta/(2m)>M$. By Theorem 5.1 and Proposition 4.2, it follows that if letting

$$N_k(x)=\sum_{j=1}^{m}\Big[\frac{K_{m;k}(0)}{m}+\Big(K_{m;k}\Big(\frac{j}{m}\Big)-K_{m;k}\Big(\frac{j-1}{m}\Big)\Big)\Big]\sigma\Big(\beta\Big(x-\frac{2j-1}{2m}\Big)\Big)+ \\ +\sum_{j=1}^{m}\frac{K_{m;k}(0)}{m}\sigma\Big(-\beta\Big(x-\frac{2j-1}{2m}\Big)\Big). \tag{5.10}$$

we have, $\|K_{m;k}(\cdot)-N_k(\cdot)\|_{C([0,\,1])}<\varepsilon/3$. By Theorem 5.2, we define the following single input four layer feedforward neural network:

$$H_{g_n}(x)=\sum_{k=0}^{m} g_n\Big(\frac{k}{m}\Big)\Big[\sum_{j=1}^{m}\frac{K_{m;k}(0)}{m}\sigma\Big(-\beta\Big(x-\frac{2j-1}{2m}\Big)\Big) \\ +\sum_{j=1}^{m}\Big(\frac{K_{m;k}(0)}{m}+\Big(K_{m;k}\Big(\frac{j}{m}\Big)-K_{m;k}\Big(\frac{j-1}{m}\Big)\Big)\Big)\sigma\Big(\beta\Big(x-\frac{2j-1}{2m}\Big)\Big)\Big]. \tag{5.11}$$

Then, $\|g_n-H_{g_n}\|_{C([0,1])}<\varepsilon\ (n\in\mathbb{N})$. By above discussion we obtain the following learning algorithm for connection weights:

Step 1. Using (5.7) we choose $m\in\mathbb{N}$, and define $M>0:\ |\sigma(t)-1|<\varepsilon/3\ (t>M)$, $|\sigma(t)|<\varepsilon/3\ (t<-M)$;

Step 2. Define the functionals $\phi_0, \phi_1, ..., \phi_m : \phi_k(g) = g(k/m)$ $(g \in V)$;

Step 3. By (5.10) define v_{jk}, $\mathbf{u}(j) = u_j \in \mathbb{R}$, θ_j $(k = 0, 1, ..., m;\ j = 1, ..., 2m)$:

$$v_{jk} = \begin{cases} \dfrac{K_{m;k}(0)}{m} + K_{m;k}\Big(\dfrac{j}{m}\Big) - K_{m;k}\Big(\dfrac{j-1}{m}\Big), & j = 1, ..., m, \\ \dfrac{K_{m;k}(0)}{m}, & j = m+1, ..., 2m; \end{cases}$$

$$u_j = \begin{cases} \beta, & j = 1, ..., m, \\ -\beta, & j = m+1, ..., 2m; \end{cases} \qquad \theta_j = \begin{cases} -\dfrac{\beta(2j-1)}{2m}, & j = 1, ..., m, \\ \dfrac{\beta(2(j-m)-1)}{2m}, & j = m+1, ..., 2m. \end{cases}$$

Let us now Illustrate the realization procedure of uniform approximation of four layer feedforward neural network within the following general framework.

Algorithm 5.1 Realizing algorithm for uniform approximation of neural network. Let compact set $V \subset C([0,\ 1]^d)$. Then

Step 1. By the given accuracy $\varepsilon > 0$, we get $m \in \mathbb{N}$, and $d-$ dimensional Bernstein polynomial $B_m(\cdot)$;

Step 2. With the accuracy $\varepsilon/3$ construct the three layer feedforward neural network $N_{i_1,...,i_d}$ corresponding to the approximating polynomial $K_{m;i_1...i_d}(\cdot)$, and get the weights $v_i(i_1, ..., i_d)$, $\theta_i(i_1, ..., i_d) \in \mathbb{R}$, and $\mathbf{u}_i(i_1, ..., i_d) \in \mathbb{R}^d$;

Step 3. Define the functional $\phi_{i_1,...,i_d}(f) = f\big(i_1/m, ..., i_d/m\big)$ $(f \in V)$, and by (5.8) calculate the connection weights: v_{jk}, $\theta_j \in \mathbb{R}$, $\mathbf{u}(j) \in \mathbb{R}^d$ $(k = 1, ..., q;\ j = 1, ..., p)$;

Step 4. Output the four layer feedforward neural network requested.

5.1.2 Uniformity analysis of three layer neural network

Let us proceed to study the uniformity of universal approximation for the three layer feedforward neural network, i.e. $m = 2$ in $N_k^m(\cdot)$. To this end, we focus on one-dimensional case, i.e. $d = 1$. Define a compact subset of $C(\mathbb{R}_+)$, which is called quasi-difference order-preserved set [40].

Definition 5.2 A function set $\mathcal{C}_{or}(\mathbb{R}_+) \subset C(\mathbb{R}_+)$ is called quasi-difference order-preserved set, if the following conditions hold:

(i) $\mathcal{C}_{or}(\mathbb{R}_+)$ is a compact set, moreover, the limit $\lim\limits_{x \to +\infty} f(x)$ exists for each $f \in \mathcal{C}_{or}(\mathbb{R}_+)$, ;

(ii) For arbitrary $A > 0$, there are $m_0 \in \mathbb{N}$, and $y_1, ..., y_{m_0} \in [0,\ A]$, so that

$$\begin{aligned} \Sigma_A &\triangleq \big\{y \in [0,\ A] \big| \text{there are } f,\ g \in \mathcal{C}_{or}(\mathbb{R}_+),\ f \not\equiv g,\ f(y) = g(y)\big\} \\ &= \{y_1, ..., y_{m_0}\}; \end{aligned} \tag{5.12}$$

(iii) $\forall a_1,\ a_2 \in [0,\ A] :\ a_1 < a_2,\ [a_1,\ a_2] \cap \{y_1, ..., y_{m_0}\} = \emptyset$, then

$$\forall f,\ g \in \mathcal{C}_{or}(\mathbb{R}_+),\ f \le g,\ \Longrightarrow\ f(a_2) - f(a_1) \le g(a_2) - g(a_1).$$

To deal with uniform approximation of three-layer neural networks, we at first show the following conclusion.

Lemma 5.2 *Let* $\mathcal{C}_0 \subset C(\mathbb{R}_+)$ *be a compact set, moreover* $\forall f \in \mathcal{C}_0$, *the limit* $\lim\limits_{x \to +\infty} f(x)$ *exists. Then for arbitrary* $\varepsilon > 0$, *there are* $A > 0$, *and* $\delta > 0$, *so that*

$$\forall x_1,\, x_2 \geq A,\; \forall f \in \mathcal{C}_0,\; \big|f(x_1) - f(x_2)\big| < \varepsilon,$$
$$\forall x_1,\, x_2 \in [0,\, A],\; \forall f \in \mathcal{C}_0,\; |x_1 - x_2| < \delta,\; \Longrightarrow\; \big|f(x_1) - f(x_2)\big| < \varepsilon.$$

Proof. By assumption we have, $\mathcal{C}_0 \subset C(\mathbb{R}_+)$ is a compact set. Then there are $q \in \mathbb{N}$, and $\{f_1, ..., f_q\} \subset \mathcal{C}_0$, such that $\forall f \in \mathcal{C}_0$, there is $q_1 \in \{1, ..., q\}$, satisfying $\big\|f - f_{q_1}\big\|_{C(\mathbb{R}_+)} < \varepsilon/4$. Moreover, $\forall q' \in \{1, ..., q\}$, the limit $\lim\limits_{x \to +\infty} f_{q'}(x)$ exists. Then there is $A > 0$, so that $\forall x_1,\, x_2 \geq A$, $\forall q' \in \{1, ..., q\}$, we have, $\big|f_{q'}(x_1) - f_{q'}(x_2)\big| < \varepsilon/4$. Arbitrarily given $f \in \mathcal{C}_0$, there is $q_0 \in \{1, ..., q\}$, such that $\big\|f - f_{q_0}\big\|_{C(\mathbb{R}_+)} < \varepsilon/4$, $\Longrightarrow$ $\forall x \in \mathbb{R}_+$, $\big|f(x) - f_{q_0}(x)\big| < \varepsilon/4$. So $\forall x_1,\, x_2 > A$, it follows that

$$\big|f(x_1) - f(x_2)\big| \leq \big|f(x_1) - f_{q_0}(x_1)\big| + \big|f_{q_0}(x_1) - f_{q_0}(x_2)\big| + \big|f_{q_0}(x_2) - f(x_2)\big| < \varepsilon.$$

Considering that $f_1, ..., f_q$ are uniformly continuous on $[0,\, A]$, with the similar reason we can show that there is $\delta > 0$, $\forall x_1,\, x_2 \in [0,\, A]$, $|x_1 - x_2| < \delta$, $\Longrightarrow$ $\forall\, f \in \mathcal{C}_0$, $\big|f(x_1) - f(x_2)\big| < \varepsilon$. The lemma is proved. □

Write $x_0 = 0$. Choose $m \in \mathbb{N}$. For $A > 0$, we partition $[0,\, A]$ into m identical length sub-intervals: $0 < A/m < \cdots < (m-1)A/m < A$. And let $x_k = kA/m$ for $k = 1, ..., m$. Denote $t_k = (x_k + x_{k-1})/2$. If $f \in C(\mathbb{R}_+)$, let

$$S_f(x) = f(0) + \sum_{k=1}^{m} \big(f(x_k) - f(x_{k-1})\big)\sigma\big(\beta(x - t_k)\big) \;\; (x \in \mathbb{R}_+), \tag{5.13}$$

where β is a constant. We shall give the uniform representation of a quasi-difference order-preserved set by three-layer feedforward neural networks.

Theorem 5.3 *Let* $\sigma : \mathbb{R} \longrightarrow [0, 1]$ *be a generalized Sigmoidal function, that is,* $\sigma(x) \to 1$ *when* $x \to +\infty$, *and* $\sigma(x) \to 0$ *when* $x \to -\infty$. *Also let* $\sigma(0) > 0$, *and* $\mathcal{C}_{or}(\mathbb{R}_+) \subset C(\mathbb{R}_+)$ *is a quasi-difference order-preserved set. Then for arbitrary* $\varepsilon > 0$, *there are* $p \in \mathbb{N}$, *and order-preserved functional* $\phi_j : \mathcal{C}_{or}(\mathbb{R}_+) \longrightarrow \mathbb{R}_+$, $\beta \in \mathbb{R}_+$, $\theta_j \in \mathbb{R}$ $(j = 1, ..., p)$, *so that for each* $f \in \mathcal{C}_{or}(\mathbb{R}_+)$, *if let* $F_f(x) = \sum\limits_{j=1}^{p} \phi_j(f) \cdot \sigma\big(\beta x - \theta_j\big)$ $(x \in \mathbb{R}_+)$, *we have,* $\big\|f - F_f\big\|_{C(\mathbb{R}_+)} < \varepsilon$.

Proof. Given arbitrarily $\varepsilon > 0$. By assumption, $\mathcal{C}_{or}(\mathbb{R}_+) \subset C(\mathbb{R}_+)$ is a quasi-difference order-preserved set. Lemma 5.2 and Definition 5.2 imply that there are $A > 0$, $m_0 \in \mathbb{N}$, $y_1, ..., y_{m_0} \in [0,\, A]$, and $m \in \mathbb{N}$, so that $A/m <$

$\max\{|y_{k_1}-y_{k_2}|\ |y_{k_1}\neq y_{k_2}, k_1, k_2=1,...,m_0\}$, moreover

$$\begin{cases} \forall x_1,\, x_2\geq A,\ \forall f\in \mathcal{C}_{or}(\mathbb{R}_+),\ \big|f(x_1)-f(x_2)\big|<\dfrac{\varepsilon}{4}, \\ \forall x_1,\, x_2\in[0,A],\ |x_1-x_2|\leq \dfrac{A}{m},\ \forall f\in \mathcal{C}_{or}(\mathbb{R}_+),\ \big|f(x_1)-f(x_2)\big|<\dfrac{\varepsilon}{4m_0}. \end{cases} \tag{5.14}$$

Partition the interval $[0,\ A]$ into m identical length sub-intervals: $0<A/m<\cdots<A(m-1)/m<A$, and let $x_k=kA/m\ (k=0,1,...,m)$, moreover let $t_k=(x_k+x_{k-1})/2\ (k=1,...,m)$. Then by definition of m, A in (5.12), it follows that $\forall k\in\{1,...,m\}$, at most there is one $x\in[x_{k-1},\ x_k]$, such that $x\in\Sigma_A$. define set K as follows:

$$K=\big\{k\in\{1,...,m\}\big|\ [x_{k-1},\ x_k]\big\}\cap\Sigma_A\neq\emptyset, \Longrightarrow \text{Card}(K)\leq m_0.$$

By assumption, $\lim\limits_{x\to+\infty}\sigma(x)=1,\ \lim\limits_{x\to-\infty}\sigma(x)=0$. Choose $\beta>0$, so that $x>\beta,\ \Longrightarrow\ |\sigma(x)-1|<1/m,\ x<-\beta,\ \Longrightarrow\ |\sigma(x)|<1/m$. Moreover, $\beta/(2m)>1$. Arbitrarily given $f\in\mathcal{C}_{or}(\mathbb{R}_+)$, defining $S_f(x)$ as (5.13) we shall prove

$$\forall x\in\mathbb{R}_+,\ \big|f(x)-S_f(x)\big|<\frac{\varepsilon}{2}, \tag{5.15}$$

In fact, $\forall x\in\mathbb{R}_+$, if $x>A$, it follows that $\forall k\in\{1,...,m\},\ \beta(x-t_k)\geq\beta A/(2m)>A,\Longrightarrow\big|\sigma\big(\beta(x-t_k)\big)-1\big|<1/m$. By (5.14) we obtain

$$\big|f(x)-f(A)\big|<\frac{\varepsilon}{4},\quad \big|f(A)-S_f(x)\big|\leq\sum_{k=1}^{m}\big|f(x_k)-f(x_{k-1})\big|\cdot\big|\sigma\big(\beta(x-t_k)\big)-1\big|<\frac{\varepsilon}{4},$$

Therefore, $\big|f(x)-S_f(x)\big|<\varepsilon/2$, that is, (5.15) holds; If $x\in[0,A]$, let $k_0\in\{1,...,m\}$, $x\in[x_{k_0-1},x_{k_0}]$. Write

$$h(x)=f(0)+\big(f(x_{k_0})-f(x_{k_0-1})\big)\sigma\big(\beta(x-t_{k_0})\big)+\sum_{k=1}^{k_0-1}\big(f(x_k)-f(x_{k-1})\big).$$

If $k<k_0$, then $\beta(x-t_k)>\beta A/(2m)$; If $k>k_0$, then $\beta(x-t_k)<-\beta A/(2m)$. Thus

$$\begin{aligned} \big|f(x)-h(x)\big| &\leq \sum_{k=1}^{k_0-1}\big|f(x_k)-f(x_{k-1})\big|\cdot\big|\sigma\big(\beta(x-t_k)\big)-1\big| \\ &\quad+\sum_{k=k_0+1}^{m}\big|f(x_k)-f(x_{k-1})\big|\cdot\big|\sigma\big(\beta(x-t_k)\big)\big| \qquad (5.16)\\ &\leq \sum_{k=1}^{k_0-1}\frac{\varepsilon}{4m}+\sum_{k=k_0+1}^{m}\frac{\varepsilon}{4m}<\frac{\varepsilon}{4}. \end{aligned}$$

Moreover, it is easy to prove $h(x) = f(x_{k_0-1})+\big(f(x_{k_0})-f(x_{k_0-1})\big)\sigma\big(\beta(x-t_{k_0})\big)$. So by (5.14) (5.16) it follows that

$$\begin{aligned}&\big|f(x)-S_f(x)\big|\leq \big|f(x)-h(x)\big|+\big|h(x)-S_f(x)\big|\\&\big|f(x)-f(x_{k_0-1})\big|+\big|f(x_{k_0})-f(x_{k_0-1})\big|\sigma\big(\beta(x-t_{k_0})\big)+\frac{\varepsilon}{4}<\frac{\varepsilon}{8}+\frac{\varepsilon}{8}+\frac{\varepsilon}{4}=\frac{\varepsilon}{2},\end{aligned}$$

Also (5.15) holds. In summary (5.15) holds. For $f \in \mathcal{C}_{or}(\mathbb{R}_+)$, define the function $F_f(\cdot)$ as follows:

$$\forall x\in\mathbb{R}_+,\ F_f(x)=f(0)+\sum_{k\in\{1,...,m\}\setminus K}\big(f(x_k)-f(x_{k-1})\big)\sigma\big(\beta(x-t_k)\big).$$

For $k \in \{1,...,m\}\setminus K$, define functional $\phi_k : \ \mathcal{C}_{or}(\mathbb{R}_+) \longrightarrow \mathbb{R}_+$, satisfying $\phi_k(f) = f(x_k)-f(x_{k-1})$, and let $u_k = \beta, \theta_k = -\beta t_k$. Re-arrange the order of set $\{1,...,m\}\setminus K$ as $\{2,...,p\}$ $(p\in\mathbb{N})$, and define $\phi_1(f) = f(0)/\sigma(0)$, $u_1 = \theta_1 = 0$. Then

$$\forall f\in\mathcal{C}_{or}(\mathbb{R}_+),\ F_f(x)=\sum_{j=1}^{p}\phi_j(f)\sigma\big(u_jx+\theta_j\big)\big).$$

By condition (iii) of Definition 5.2 and the definition of set K we may easily prove that $\phi_1,...,\phi_p : \ \mathcal{C}_{or}(\mathbb{R}_+) \longrightarrow \mathbb{R}_+$ are order-preserved functionals. By (5.14), $\big|S_f(x)-F_f(x)\big|\leq \sum_{k\in K}\big|f(x_k)-f(x_{k-1})\big|\cdot\big|\sigma\big(\beta(x-t_k)\big)\big|< m_0\cdot\varepsilon/(4m_0) = \varepsilon/4$ for each $x \in \mathbb{R}_+$. Thus, if $f \in \mathcal{C}_{or}(\mathbb{R}_+)$, by (5.15) we can conclude for $x \in \mathbb{R}_+$ that

$$\big|f(x)-F_f(x)\big)\big|\leq \big|f(x)-S_f(x)\big|+\big|S_f(x)-F_f(x)\big|<\frac{\varepsilon}{4}+\frac{\varepsilon}{4}=\frac{\varepsilon}{2}.$$

Thus, $\big\|f-F_f\big\|_{C(\mathbb{R}_+)}\leq \varepsilon/2<\varepsilon$. The theorem is proved. □

Let us now develop a learning algorithm by an example to realize the approximation shown by Theorem 5.4. A concrete constructing procedure of a three-layer feedforward neural network is also presented.

Example 5.2 Let the function $f_k(x) = k/\big((k+1)(1+\exp(-x))\big)$ for $k \in \mathbb{N}$. Then $\{f_k\} \subset C(\mathbb{R}_+)$. Further it is easy to prove that $\{f_k\}$ is a quasi-difference order-preserved set, here for arbitrary $A > 0$, we may choose $\Sigma_A = \emptyset$. Give arbitrarily $\varepsilon > 0$.

Step 1. Let $A = \max\big(1,\ 1+\ln(4/\varepsilon)\big)$. Choose $m \in \mathbb{N}:\ A/m < \varepsilon/4$. Then

$$\begin{aligned}&\forall x\geq A,\ \forall k\in\mathbb{N},\ \Big|f_k(x)-\frac{k}{k+1}\Big|<\exp(-x)<\frac{\varepsilon}{4},\\&\forall x_1,\,x_2\in[0,\,A],\ |x_1-x_2|\leq\frac{A}{m},\Longrightarrow\big|f_k(x_1)-f_k(x_2)\big|<|x_1-x_2|<\frac{\varepsilon}{4}.\end{aligned}$$

Step 2. Choose $t_j = (2j-1)A/2m$, and let $\beta > 0: \ |\sigma(x) - 1| < 1/m \ (x > \beta)$, $|\sigma(x)| < 1/m \ (x < -\beta)$, $\beta/(2m) > 1$. Denote

$$F_{f_k}(x) = \sum_{j=1}^{m+1} \phi_j(f_k) \cdot \sigma\big(\beta x + \theta_j\big).$$

Then $F_f(\cdot)$ is the three-layer feedforward neural network that satisfies the given conditions, where $\phi_1(f_k) = f_k(0)/\sigma(0)$, $\theta_1 = 0$, $\phi_j(f_k) = f_k(x_{j-1}) - f_k(x_{j-2})$, $\theta_j = -\beta t_{j-1}$ $(j = 2, ..., m+1)$. Easily it follows that $\phi_j : \{f_k\} \longrightarrow \mathbb{R}_+$ is an order-preserved functional.

In above discussion, we establish a family of continuous functions, each of which can be approximated by three-layer feedforward neural networks with identical base function, to arbitrary degree of accuracy. The conclusion generalizes the approximation results related feedforward neural networks [8, 9, 27]. Moreover, it can be applied to construct some fuzzy neural networks, which can provide the approximations with arbitrary accuracy to a class of fuzzy functions. That is our research subject in §5.4.

§5.2 Symmetric polygonal fuzzy number

The research on fuzzy numbers has received considerable attention, and many theoretical achievements related have emerged (see [6, 10–12, 18, 29, 43, 45, 62] etc). Fuzzy numbers have found useful in many research fields, such as fuzzy control [2, 3, 26], fuzzy neural networks [13, 14, 19–22, 34–36, 48], fuzzy reliability [4], fuzzy system analysis [45], signal process [5], expert system [1, 7, 17], regression analysis [16, 23–25, 56–58], programming problem [49, 59] and so on.

Recent years, many scholars take general fuzzy numbers as basic tools to deal with their respective problems with vagueness and uncertainty. For instance, Chen et al [5] use generic LR fuzzy numbers to develop a noise filter based on a family of fuzzy inference rules, and corresponding capability for noise removal is improved, significantly. And Kuo et al [28] present a primitive LR fuzzy cell structure for hardware realization of fuzzy computations. In [6] Chen employs some step form fuzzy numbers to simplify the fuzzy arithmetic operations, such as addition, multiplication, subtraction and division and so on. However, we still encounter many drawbacks when the LR or step form fuzzy numbers are applied in real, such as: When representing LR fuzzy numbers through a few of adjustable parameters, we can not control the approximating accuracy, efficiently. It is not convenient to take LR fuzzy numbers as fuzzy weights of fuzzy neural networks; Although a step form fuzzy number can be determined, uniquely by a few of parameters, its form is too complicated for calculating fuzzy relational structures to be very useful for solving real problems. Also it is difficult to develop some learning algorithms for fuzzy neural networks with step form fuzzy number weights. Furthermore, these fuzzy numbers are not closed under the fuzzy arithmetic operations based on Zadeh's extension principle [15], [47].

Therefore it is urgent that the following problems are studied and solved. First, the fuzzy arithmetic based on Zadeh's principle should appropriately be modified, so that the common fuzzy numbers are closed under some non-linear operations, for example multiplication and division. Second, whether can the triangular and trapezoidal fuzzy numbers be generalized as general ones, so that the novel fuzzy numbers have simple forms and can represent a large of class of fuzzy sets, approximately. Third, The new fuzzy number space possesses similar properties with ones of triangular or trapezoidal fuzzy number space. Finally, a few of adjustable parameters can determine such a fuzzy number, so we can use them to deal with some complicated fuzzy computations. In the section our aim is to introduce polygonal fuzzy numbers and polygonal line operators and to present the systematic studies to above problems.

As in §4.2, we deal with our questions in the fuzzy number space $\mathcal{F}_{0c}(\mathbb{R})$. Let $\widetilde{A}\in \mathcal{F}_{0c}(\mathbb{R})$, and denote $\widetilde{A}_0= [a_0^1,\, a_0^2]$, $\mathrm{Ker}(\widetilde{A}) = [e_0^1,\, e_0^2]$. Then $\widetilde{A}(\cdot)$ is strictly increasing and right continuous on $[a_0^1,\, e_0^1)$; and strictly decreasing and left continuous on $(e_0^2,\, a_0^2]$; on $[e_0^1,\, e_0^2]$ we have $\widetilde{A}(x) \equiv 1$.

5.2.1 Symmetric polygonal fuzzy number space

As the generalizations of triangular and trapezoidal fuzzy numbers, we define n-symmetric polygonal fuzzy numbers for $n \in \mathbb{N}$ in the section. They are similar with triangular and trapezoidal fuzzy numbers in their structures and basic properties (also see [32, 37, 38]).

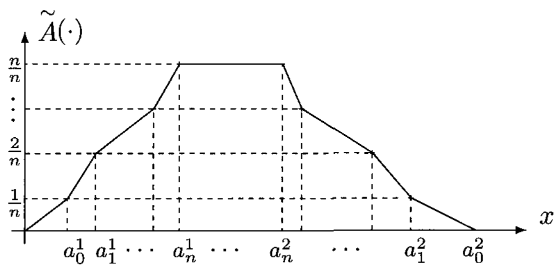


Figure 5.1 The symmetric polygonal fuzzy number $\widetilde{A}$

Definition 5.3 Let $\widetilde{A}\in \mathcal{F}_{0c}(\mathbb{R})$. If there exist $n \in \mathbb{N}$, and $a_0^1, a_1^1, ..., a_n^1,\, a_n^2,\, a_{n-1}^2, ..., a_0^2 \in \mathbb{R}$: $a_0^1 \le \cdots \le a_n^1 \le a_n^2 \le \cdots \le a_0^2$, so that the following conditions hold:

(i) $\mathrm{Supp}(\widetilde{A}) = [a_0^1,\, a_0^2]$, $\mathrm{Ker}(\widetilde{A}) = [a_n^1,\, a_n^2]$;

(ii) Let $k \in \{1, ..., n\}$. Then $a_{k-1}^1 < a_k^1 \Longrightarrow \widetilde{A}(a_{k-1}^1) = (k-1)/n$, $\widetilde{A}(a_k^1 - 0) = k/n$, and $a_k^2 < a_{k-1}^2, \Longrightarrow \widetilde{A}(a_{k-1}^2) = (k-1)/n$, $\widetilde{A}(a_k^2 + 0) = k/n$;

(iii) $\forall k \in \{1, ..., n\}$, $\widetilde{A}(\cdot)$ is linear on $[a_{k-1}^1,\, a_k^1]$ and $[a_k^2,\, a_{k-1}^2]$, respectively.

Here $\widetilde{A}(a_k^1-0)$ is the left limit $\lim\limits_{t\to 0+}\widetilde{A}(a_k^1-t)$, and $\widetilde{A}(a_k^2+0)$ is the right limit $\lim\limits_{t\to 0+}\widetilde{A}(a_k^2+t)$. $\widetilde{A}$ is called a $n-$symmetric polygonal fuzzy number, and we denote $\widetilde{A}$ by $\widetilde{A}=([a_n^1,\ a_n^2];\ a_0^1,...,a_{n-1}^1,\ a_{n-1}^2,...,a_0^2)$. The collection of all $n-$symmetric polygonal fuzzy numbers is denoted by $\mathcal{F}_{0c}^{tn}(\mathbb{R})$.

In particular, let $n=1$, then an 1−symmetric polygonal fuzzy number is triangular or trapezoidal. If $\widetilde{A}=([a_n^1,\ a_n^2];\ a_0^1,...,a_{n-1}^1,a_{n-1}^2,...,a_0^2)\in\mathcal{F}_{0c}^{tn}(\mathbb{R})$, then we have, $\mathrm{Ker}(\widetilde{A})=[a_n^1,\ a_n^2]$, $\mathrm{Supp}(\widetilde{A})=[a_0^1,\ a_0^2]$, and

$$\begin{cases} a_0^1\le a_1^1\le\cdots\le a_n^1\le a_n^2\le a_{n-1}^2\le\cdots\le a_0^2, \\ \forall i=0,1,...,n,\ \widetilde{A}_{i/n}=\left[a_i^1,\ a_i^2\right]. \end{cases} \tag{5.17}$$

Moreover

$$\begin{cases} 0\le\widetilde{A}(a_0^1)\le\widetilde{A}(a_1^1)\le\cdots\le\widetilde{A}(a_{n-1}^1)<\widetilde{A}(a_n^1)=1, \\ 0\le\widetilde{A}(a_0^2)\le\widetilde{A}(a_1^2)\le\cdots\le\widetilde{A}(a_{n-1}^2)<\widetilde{A}(a_n^2)=1. \end{cases} \tag{5.18}$$

The fuzzy numbers defined here are simpler and more applicable than ones with step forms in [6]. By $\mathcal{F}_{0c}^{tn}(\mathbb{R}_+)$ we denote the collection of all nonnegative fuzzy numbers in $\mathcal{F}_{0c}^{tn}(\mathbb{R})$, that is, $\forall\ \widetilde{A}\in\mathcal{F}_{0c}^{tn}(\mathbb{R}_+)$, it follows $\forall x<0,\ \widetilde{A}(x)=0$.

Theorem 5.4 *Let* $\widetilde{A},\ \widetilde{B}\in\mathcal{F}_{0c}^{tn}(\mathbb{R})$. *Moreover let*

$$\widetilde{A}=([a_n^1,\ a_n^2];\ a_0^1,...,a_{n-1}^1,\ a_{n-1}^2,...,a_0^2),$$
$$\widetilde{B}=([b_n^1,\ b_n^2];\ b_0^1,...,b_{n-1}^1,\ b_{n-1}^2,...,b_0^2).$$

Then the following conclusions hold:

(i) $D(\widetilde{A},\ \widetilde{B})=\bigvee\limits_{i=0}^{n}\left\{d_H\left([a_i^1,\ a_i^2],\ \ [b_i^1,\ b_i^2]\right)\right\}$;

(ii) $\widetilde{A}\subset\widetilde{B}\Longleftrightarrow\forall i\in\{0,1,...,n\},\ b_i^1\le a_i^1\le a_i^2\le b_i^2$.

Proof. (i) Give arbitrarily $\alpha\in[0,\ 1]$. Let $\widetilde{A}_\alpha=[a_\alpha^1,\ a_\alpha^2]$, $\widetilde{B}_\alpha=[b_\alpha^1,\ b_\alpha^2]$, $\alpha\in\left[(i-1)/n,\ i/n\right]$ for some $i\in\{1,...,n\}$. Let us now prove, $|a_\alpha^1-b_\alpha^1|\le|a_i^1-b_i^1|\vee|a_{i-1}^1-b_{i-1}^1|$ in the following cases, respectively.

I. $a_{i-1}^1<a_i^1,\ b_{i-1}^1<b_i^1$; II. $a_{i-1}^1=a_i^1,\ b_{i-1}^1<b_i^1$;

III. $a_{i-1}^1<a_i^1,\ b_{i-1}^1=b_i^1$; IV. $a_{i-1}^1=a_i^1,\ b_{i-1}^1=b_i^1$.

It is no harm to assume $(i-1)/n<\alpha<i/n$. Using Definition 5.3, we get in case I that, $\widetilde{A}(a_{i-1}^1)=\widetilde{A}(a_{i-1}^2)=(i-1)/n$; $\widetilde{A}(a_i^1-0)=\widetilde{A}(a_i^2+0)=i/n$, moreover, the following facts hold:

$$\begin{cases} a_\alpha^1=a_{i-1}^1+(n\alpha-(i-1))(a_i^1-a_{i-1}^1), \\ b_\alpha^1=b_{i-1}^1+(n\alpha-(i-1))(b_i^1-b_{i-1}^1). \end{cases} \tag{5.19}$$

Therefore, it follows that

$$\begin{aligned} |a_\alpha^1 - b_\alpha^1| &= |(n\alpha - (i-1))(a_i^1 - b_i^1) + (i - n\alpha)(a_{i-1}^1 - b_{i-1}^1)| \\ &\le |a_i^1 - b_i^1| \vee |a_{i-1}^1 - b_{i-1}^1|. \end{aligned} \tag{5.20}$$

Easily we can show, (5.19) holds in case II, and so we also get (5.20). Similarly we can prove (5.20) in case III. And in case IV, $a_\alpha^1 - b_\alpha^1 = 0 = |a_i^1 - b_i^1| = |a_{i-1}^1 - b_{i-1}^1|$, which also ensures (5.20) to hold. In summary, we have $|a_\alpha^1 - b_\alpha^1| \le |a_i^1 - b_i^1| \vee |a_{i-1}^1 - b_{i-1}^1|$. With the same reason, we can show, $|a_\alpha^2 - b_\alpha^2| \le |a_i^2 - b_i^2| \vee |a_{i-1}^2 - b_{i-1}^2|$. Consequently

$$\begin{cases} d_H\big([a_\alpha^1,\ a_\alpha^2],\ [b_\alpha^1,\ b_\alpha^2]\big) \le \max\{K_{i-1},\ K_i\}, \\ K_{i-1} = d_H\big([a_{i-1}^1,\ a_{i-1}^2],\ [b_{i-1}^1,\ b_{i-1}^2]\big),\ \ K_i = d_H\big([a_i^1,\ a_i^2],\ [b_i^1,\ b_i^2]\big). \end{cases}$$

So $D(\widetilde{A},\ \widetilde{B}) = \bigvee_{0\le\alpha\le1} \{d_H(\widetilde{A}_\alpha,\ \widetilde{B}_\alpha)\} = \bigvee_{i=0}^{n} \{d_H(\widetilde{A}_{i/n},\ \widetilde{B}_{i/n})\}$, which implies (i) by (5.17).

(ii) $\widetilde{A} \subset \widetilde{B} \Longrightarrow \forall i \in \{0, 1, ..., n\},\ \widetilde{A}_{i/n} \subset \widetilde{B}_{i/n}$. So $\forall i \in \{0, 1, ..., n\},\ [a_i^1,\ a_i^2] \subset [b_i^1,\ b_i^2]$, i.e. $b_i^1 \le a_i^1 \le a_i^2 \le b_i^2$. Conversely, $\forall \alpha \in [0,1]:\ \alpha \in \big[(i-1)/n,\ i/n\big]$. Using (5.19) we obtain, $[a_\alpha^1,\ a_\alpha^2] \subset [b_\alpha^1,\ b_\alpha^2]$ holds, that is, $\widetilde{A}_\alpha \subset \widetilde{B}_\alpha$. So $\widetilde{A} \subset \widetilde{B}$. □

By the definition of $\mathcal{F}_{0c}^{tn}(\mathbb{R})$ and (5.19) we can get the following conclusion.

Remark 5.1 Let $\widetilde{A} \in \mathcal{F}_{0c}^{tn}(\mathbb{R})$. Define the interval valued function I_f : $I_f(\alpha) = \widetilde{A}_\alpha$ ($\alpha \in [0,\ 1]$). Then I_f is uniformly continuous on $[0,\ 1]$, that is, $\forall \varepsilon > 0$, there is $\delta > 0$, so that $\forall \alpha_1,\ \alpha_2 \in [0,1],\ |\alpha_1 - \alpha_2| < \delta \Longrightarrow d_H(\widetilde{A}_{\alpha_1},\ \widetilde{A}_{\alpha_2}) < \varepsilon$.

In fact, for each $\widetilde{A} = ([a_n^1,\ a_n^2];\ a_0^1, ..., a_{n-1}^1,\ a_{n-1}^2, ..., a_0^2) \in \mathcal{F}_{0c}^{tn}(\mathbb{R})$, it is no harm to assume that $a_0^1 < a_n^1$, $a_n^2 < a_0^2$. For arbitrary $\varepsilon > 0$, choose $\delta = \min\{1/(2n),\ \varepsilon/(2na_n^1 - 2na_0^1),\ \varepsilon/(2na_0^2 - 2na_n^2)\} > 0$. $\forall \alpha_1,\ \alpha_2 \in [0,1]$: $|\alpha_1 - \alpha_2| < \delta$. There is $i \in \{1, ..., n-1\}$, so that $\alpha_1,\ \alpha_2 \in \big[(i-1)/n,\ (i+1)/n\big]$. By (5.19) it follows that

$$\big|a_{\alpha_1}^1 - a_{\alpha_2}^1\big| = (a_{i+1}^1 - a_{i-1}^1) \cdot \big|n\alpha_1 - n\alpha_2\big| \le n(a_n^1 - a_0^1) \cdot \big|\alpha_1 - \alpha_2\big| \le \frac{\varepsilon}{2}.$$

Similarly, $\big|a_{\alpha_1}^2 - a_{\alpha_2}^2\big| \le \varepsilon/2$. So $d_H(\widetilde{A}_{\alpha_1},\ \widetilde{A}_{\alpha_2}) = |a_{\alpha_1}^1 - a_{\alpha_2}^1| \vee |a_{\alpha_1}^2 - a_{\alpha_2}^2| < \varepsilon$.

Theorem 5.5 *For a given $n \in \mathbb{N}$, $(\mathcal{F}_{0c}^{tn}(\mathbb{R}),\ D)$ is a completely separable metric space.*

Proof. It is trivial that $(\mathcal{F}_{0c}^{tn}(\mathbb{R}),\ D)$ is metric space. At first we prove the completeness of the space. Let $\{\widetilde{A}[k]\,|k \in \mathbb{N}\} \subset \mathcal{F}_{0c}^{tn}(\mathbb{R})$ be a basic sequence, that is, $\forall \varepsilon > 0$, there is $K \in \mathbb{N}$, so that $\forall k_1,\ k_2 > K,\ D(\widetilde{A}[k_1],\ \widetilde{A}[k_2]) <$

$\varepsilon/2$, where $\widetilde{A}[k] = \big([a_n^1(k),\, a_n^2(k)];\ a_0^1(k), ..., a_{n-1}^1(k),\, a_{n-1}^2(k), ..., a_0^2(k)\big)$. By Theorem 5.4, for $k_1,\, k_2 > K,\ i = 0, 1, ..., n$, it follows that

$$d_H\big([a_i^1(k_1),\, a_i^2(k_1)],\ [a_i^1(k_2),\, a_i^2(k_2)]\big) < \frac{\varepsilon}{2}.$$

So $\forall i \in \{0, 1, ..., n\}$, we get

$$\big|a_i^1(k_1) - a_i^1(k_2)\big| < \frac{\varepsilon}{2},\quad \big|a_i^2(k_1) - a_i^2(k_2)\big| < \frac{\varepsilon}{2}.$$

Therefore for each $i = 0, 1, ..., n$, both $\{a_i^1(k)|k \in \mathbb{N}\},\ \{a_i^2(k)|k \in \mathbb{N}\} \subset \mathbb{R}$ are basic sequences. There are $a_i^1,\, a_i^2 \in \mathbb{R}$, so that $\lim\limits_{k\to+\infty} a_i^1(k) = a_i^1,\ \lim\limits_{k\to+\infty} a_i^2(k) = a_i^2$. Moreover, $\forall\, k \in \mathbb{N},\ a_{i-1}^1(k) \le a_i^1(k) \le a_i^2(k) \le a_{i-1}^2(k) \Longrightarrow a_{i-1}^1 \le a_i^1 \le a_i^2 \le a_{i-1}^2$. Let $\widetilde{A} = ([a_n^1,\, a_n^2];\, a_0^1, ..., a_{n-1}^1,\, a_{n-1}^2, ..., a_0^2)$. Then $\widetilde{A} \in \mathcal{F}_{0c}^{tn}(\mathbb{R})$. So, there is $m \in \mathbb{N}:\ \forall\, k > m,\ \forall i \in \{0, 1, ..., n\}$, we have

$$\big|a_i^1(k) - a_i^1\big| < \frac{\varepsilon}{2},\ \big|a_i^2(k) - a_i^2\big| < \frac{\varepsilon}{2},\ \Longrightarrow\ d_H\big([a_i^1(k),\, a_i^2(k)],\ [a_i^1,\, a_i^2]\big) < \frac{\varepsilon}{2}.$$

By Theorem 5.4, when $k > m$, we have

$$D\big(\widetilde{A}[k],\ \widetilde{A}\,\big) = \bigvee_{i=0}^{n}\big\{d_H\big([a_i^1(k),\, a_i^2(k)],\ [a_i^1,\, a_i^2]\big)\big\} < \varepsilon,$$

that is, $\lim\limits_{k\to+\infty} \widetilde{A}[k] = \widetilde{A}$. Thus, $(\mathcal{F}_{0c}^{tn}(\mathbb{R}),\ D)$ is complete.

Next let us prove the separability. Define the set $\mathcal{C}$ as follows:

$$\mathcal{C} = \big\{\widetilde{L} = ([l_n^1,\, l_n^2];\ l_0^1, ..., l_{n-1}^1,\, l_{n-1}^2, ..., l_0^2) \in \mathcal{F}_{0c}^{tn}(\mathbb{R})\big|\ l_0^1,\, l_1^1, ..., l_n^1,\, l_n^2, ..., l_0^2 \text{ are rational numbers}\big\}.$$

Obviously, $\mathcal{C} \subset \mathcal{F}_{0c}^{tn}(\mathbb{R})$ is a countable set. And given arbitrarily $\widetilde{A} \in \mathcal{F}_{0c}^{tn}(\mathbb{R})$, let $\widetilde{A} = ([a_n^1,\, a_n^2];\, a_0^1, ..., a_{n-1}^1,\, a_{n-1}^2, ..., a_0^2)$. $\forall\, \varepsilon > 0$, choose rational numbers $l_0^1, ..., l_n^1,\, l_n^2, ..., l_0^2$, satisfying $l_0^1 \le \cdots \le l_n^1 \le l_n^2 \le l_{n-1}^2 \le \cdots \le l_0^2$, and $\forall i \in \{0, 1, ..., n\},\ k = 1, 2,\ |a_i^k - l_i^k| < \varepsilon$. Let $\widetilde{L} = ([l_n^1,\, l_n^2];\, l_0^1, ..., l_{n-1}^1,\, l_{n-1}^2, ..., l_0^2)$. By Theorem 5.4, $\widetilde{L} \in \mathcal{F}_{0c}^{tn}(\mathbb{R})$, and $d_H\big([a_i^1,\, a_i^2],\ [l_i^1,\, l_i^2]\big) = |a_i^1 - l_i^1| \vee |a_i^2 - l_i^2| < \varepsilon$. Then

$$D(\widetilde{A},\ \widetilde{L}) = \bigvee_{i=0}^{n}\{d_H([a_i^1,\, a_i^2],\ [l_i^1,\, l_i^2])\} < \varepsilon,$$

that is, $\mathcal{C}$ is dense in $\mathcal{F}_{0c}^{tn}(\mathbb{R})$. So $\mathcal{F}_{0c}^{tn}(\mathbb{R})$ is separable. Hence $(\mathcal{F}_{0c}^{tn}(\mathbb{R}),\ D)$ is a completely separable metric space. □

Theorem 5.6 *Suppose $\mathcal{U} \subset \mathcal{F}_{0c}(\mathbb{R})$. Then the following conclusions hold:*

(i) $\mathcal{U}$ is bounded if and only if there is a compact set $U \subset \mathbb{R}$, so that $\forall\, \widetilde{X} \in \mathcal{U}, \mathrm{Supp}(\widetilde{X}) \subset U$;

(ii) $\mathcal{U} \subset \mathcal{F}_{0c}^{tn}(\mathbb{R})$ is a compact set if and only if $\mathcal{U}$ is bounded and closed.

Proof. (i) Suppose that $\mathcal{U} \subset \mathcal{F}_{0c}(\mathbb{R})$ is bounded. And choosing $\widetilde{A} \in \mathcal{U}$, we imply, there is a $M > 0$, such that $\forall\, \widetilde{X} \in \mathcal{U},\ D(\widetilde{A}, \widetilde{X}) \le M$. Assume that $\mathrm{Supp}(\widetilde{A}) = [a_0^1,\, a_0^2]$, and set $U = [a_0^1 - M,\, a_0^2 + M]$, then $U \subset \mathbb{R}$ is a compact set. Moreover, $\forall\, \widetilde{X} \in \mathcal{U}$, if let $\mathrm{Supp}(\widetilde{X}) = [x_0^1,\, x_0^2]$, we have $D(\widetilde{A},\ \widetilde{X}) \le M$, i.e. $d_H\big([a_0^1,\, a_0^2],\ [x_0^1,\, x_0^2]\big) = |a_0^1 - x_0^1| \vee |a_0^2 - x_0^2| \le M$. So $a_0^1 - M \le x_0^1 \le x_0^2 \le a_0^2 + M$. Thus $\mathrm{Supp}(\widetilde{X}) \subset [a_0^1 - M,\, a_0^2 + M] = U$.

Conversely, it is no harm to assume $U = [-M,\, M] \subset \mathbb{R}$ is a symmetrically bounded and closed interval, so that $\forall\, \widetilde{X} \in \mathcal{U},\ \mathrm{Supp}(\widetilde{X}) \subset [-M, M]$. Give $\widetilde{A} \in \mathcal{U}$. Then $\forall\, \widetilde{X} \in \mathcal{U}$, if let $\widetilde{A}_\alpha = [a_\alpha^1,\, a_\alpha^2]$, $\widetilde{X}_\alpha = [x_\alpha^1,\, x_\alpha^2]$ for each $\alpha \in [0, 1]$, we have $\widetilde{A}_\alpha,\ \widetilde{X}_\alpha \subset [-M,\, M]$. So

$$d_H(\widetilde{A}_\alpha, \widetilde{X}_\alpha) = |a_\alpha^1 - x_\alpha^1| \vee |a_\alpha^2 - x_\alpha^2| \le (|a_\alpha^1| + |x_\alpha^1|) \vee (|a_\alpha^2| + |x_\alpha^2|) \le 2M.$$

Therefore, $D(\widetilde{A},\ \widetilde{X}) = \bigvee_{\alpha \in [0,1]} \{d_H(\widetilde{A}_\alpha,\ \widetilde{X}_\alpha)\} \le 2M$ for each $\widetilde{X} \in \mathcal{U}$, that is, $\mathcal{U}$ is bounded.

(ii) Assume that $\mathcal{U} \subset \mathcal{F}_{0c}^{tn}(\mathbb{R})$ is bounded and closed. It suffices to prove $\mathcal{U}$ is sequentially compact. Let $\{\widetilde{A}[k]\,|k \in \mathbb{N}\} \subset \mathcal{U}$, and write

$$\widetilde{A}[k] = \big([a_n^1(k),\, a_n^2(k)];\ a_0^1(k), ..., a_{n-1}^1(k),\ a_{n-1}^2(k), ..., a_0^2(k)\big)\ (k = 1, 2, ...). \tag{5.21}$$

By (i), there is a compact set $U \subset \mathbb{R}$, such that $\{a_i^j(k)|k \in \mathbb{N}\} \subset U$ $(i = 0, 1, ..., n;\ j = 1, 2)$. So there is convergent subsequence of $\{a_i^j(k)|k \in \mathbb{N}\}$ for each $i = 0, 1, ..., n;\ j = 1, 2$. By the method of choosing subsequences, it is no harm to assume $\lim_{k \to +\infty} a_i^j(k) = a_i^j$. Using (5.21) we obtain $a_{i-1}^1 \le a_i^1 \le a_i^2 \le a_{i-1}^1$ for $i = 1, ..., n$. So $\widetilde{A} \triangleq \big([a_n^1,\, a_n^2];\ a_0^1, ..., a_{n-1}^1,\ a_{n-1}^2, ..., a_0^2\big) \in \mathcal{F}_{0c}^{tn}(\mathbb{R})$. Since $\mathcal{U}$ is closed, $\widetilde{A} \in \mathcal{U}$. Thus $\mathcal{U}$ is sequentially compact.

The necessity is obvious. □

By Theorem 5.6 $(\mathcal{F}_{0c}^{tn}(\mathbb{R}),\ D)$ is a locally compact space. Thus, by Theorem 5.5 and Theorem 5.6 the properties of symmetric polygonal fuzzy numbers are also similar with ones of triangular or trapezoidal fuzzy numbers. And the conclusions in [55] are generalized to general cases.

Theorem 5.7 *Let $\mathcal{U} \subset \mathcal{F}_{0c}^{tn}(\mathbb{R})$ be bounded and closed. Then $\{\widetilde{A}_\alpha \mid \widetilde{A} \in \mathcal{U}\}$ is equicontinuous with respect to α, that is, $\forall\, \varepsilon > 0$, there is $\delta > 0$, so that $\forall \alpha_1,\ \alpha_2 \in [0, 1]$, and for each $\widetilde{A} \in \mathcal{U}$, $|\alpha_1 - \alpha_2| < \delta, \Longrightarrow d_H(\widetilde{A}_{\alpha_1},\ \widetilde{A}_{\alpha_2}) < \varepsilon$.*

Proof. For any $\varepsilon > 0$, by $\mathcal{V}_{\varepsilon/3}(\widetilde{A}) \subset \mathcal{F}_{0c}^{tn}(\mathbb{R})$ we denote $\varepsilon/3-$neighborhood of given $\widetilde{A}$. By Theorem 5.6, $\mathcal{U}$ is compact, consequently by the fact that $\bigcup_{\widetilde{A}\in\mathcal{U}} \mathcal{V}_{\varepsilon/3}(\widetilde{A}) \supset \mathcal{U}$ we show that there are $m \in \mathbb{N}$ and $\widetilde{A}_1, ..., \widetilde{A}_m \in \mathcal{U}$, so that $\bigcup_{i=1}^{m} \mathcal{V}_{\varepsilon/3}(\widetilde{A}_i) \supset \mathcal{U}$. By Remark 5.1, there is $\delta > 0$, such that $\forall i \in \{1, ..., m\}$, and $\forall \alpha_1, \alpha_2 \in [0,1]$, $|\alpha_1 - \alpha_2| < \delta$, $\Longrightarrow d_H\big((\widetilde{A}_i)_{\alpha_1}, (\widetilde{A}_i)_{\alpha_2}\big) < \varepsilon/3$. Therefore, $\forall \widetilde{A} \in \mathcal{U}$, let $i_0 \in \{1, ..., m\}$, $\widetilde{A} \in \mathcal{V}_{\varepsilon/3}(\widetilde{A}_{i_0})$. Thus

$$\begin{aligned} d_H\big(\widetilde{A}_{\alpha_1}, \widetilde{A}_{\alpha_2}\big) &\leq d_H\big(\widetilde{A}_{\alpha_1}, (\widetilde{A}_{i_0})_{\alpha_1}\big) + d_H\big((\widetilde{A}_{i_0})_{\alpha_1}, (\widetilde{A}_{i_0})_{\alpha_2}\big) \\ &\quad + d_H\big((\widetilde{A}_{i_0})_{\alpha_2}, \widetilde{A}_{\alpha_2}\big) < \frac{\varepsilon}{3} + \frac{\varepsilon}{3} + \frac{\varepsilon}{3} = \varepsilon. \end{aligned}$$

So $\{\widetilde{A}_\alpha \mid \widetilde{A} \in \mathcal{U}\}$ is equicontinuous with respect to α. □

5.2.2 Polygonal line operator

Let us discuss now the approximation capability of $\mathcal{F}_{0c}^{tn}(\mathbb{R})$ to fuzzy numbers in $\mathcal{F}_{0c}(\mathbb{R})$. Choose $\widetilde{A} \in \mathcal{F}_{0c}(\mathbb{R})$, and $n \in \mathbb{N}$. Partition [0, 1] into n equal parts: $0 < 1/n < \cdots < (n-1)/n < 1$. Let $\widetilde{A}_{i/n} = [a_i^1, a_i^2]$ for $i \in \{0, 1, ..., n\}$. Link the points $\big(a_0^1, \widetilde{A}(a_0^1)\big), ..., \big(a_n^1, \widetilde{A}(a_n^1)\big)$, $\big(a_n^2, \widetilde{A}(a_n^2)\big), ..., \big(a_0^2, \widetilde{A}(a_0^2)\big)$, by line successively. And a polygonal line denoted by $t\widetilde{A}_n\ (\cdot)$ is established. Obviously, fuzzy number $t\widetilde{A}_n \in \mathcal{F}_{0c}^{tn}(\mathbb{R})$. $t\widetilde{A}_n$ is called $n-$symmetric polygonal fuzzy number with respect to $\widetilde{A}$. The membership curve of such a $n-$polygonal fuzzy number is shown in Figure 5.2. We can show, $\text{Ker}(\widetilde{A}) = \text{Ker}(t\widetilde{A}_n) = [a_n^1, a_n^2]$, $\text{Supp}(\widetilde{A}) = \text{Supp}(t\widetilde{A}_n) = [a_0^1, a_0^2]$. Moreover

$$\begin{cases} a_0^1 \leq a_1^1 \leq \cdots \leq a_n^1 \leq a_n^2 \leq a_{n-1}^2 \leq \cdots \leq a_0^2; \\ \forall i \in \{0, 1, ..., n\}, \widetilde{A}_{i/n} = (t\widetilde{A}_n)_{i/n}. \end{cases} \tag{5.22}$$

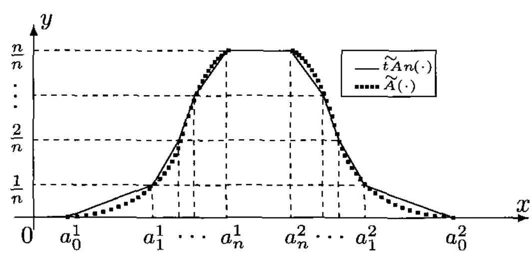


Figure 5.2 Illustration of polygonal fuzzy operator

Let $n \in \mathbb{N}$, define the operator $Z_n : \mathcal{F}_{0c}(\mathbb{R}) \longrightarrow \mathcal{F}_{0c}^{tn}(\mathbb{R})$ as follows:

$$\forall \widetilde{A} \in \mathcal{F}_{0c}(\mathbb{R}),\ Z_n(\widetilde{A}) = t\widetilde{A}_n\ .$$

$Z_n(\cdot)$ is called a $n-$polygonal line operator.

Theorem 5.8 *Let* $\widetilde{A},\ \widetilde{B}\in \mathcal{F}_{0c}(\mathbb{R})$, *and* $n\in\mathbb{N}$. *Write*

$$Z_n(\widetilde{A})=([a_n^1,\ a_n^2];\ a_0^1,...,a_{n-1}^1,\ a_{n-1}^2,...,a_0^2),$$

$$Z_n(\widetilde{B})=([b_n^1,\ b_n^2];\ b_0^1,...,b_{n-1}^1,\ b_{n-1}^2,...,b_0^2).$$

Then the following conclusions hold:

(i) $Z_n(\widetilde{A})\subset Z_n(\widetilde{B})\Longleftrightarrow \forall i\in\{0,1,...,n\},\ [a_i^1,\ a_i^2]\subset[b_i^1,\ b_i^2]$;

(ii) $\widetilde{A}\subset\widetilde{B}$ *if and only if* $\forall n\in\mathbb{N},\ Z_n(\widetilde{A})\subset Z_n(\widetilde{B})$;

(iii) If $m\in\mathbb{N}$, *then* $D(Z_m(\widetilde{A}),\ Z_m(\widetilde{B}))\le D(\widetilde{A},\ \widetilde{B})$. *Moreover, we have,* $\lim\limits_{m\to+\infty} D(\widetilde{A},\ Z_m(\widetilde{A}))=0$.

Proof. (i) Since $Z_n(\widetilde{A}),\ Z_n(\widetilde{B})\in\mathcal{F}_{0c}^{tn}(\mathbb{R})$, (i) is directly obtained by Theorem 5.4.

(ii) Let $\widetilde{A}\subset\widetilde{B}$. Then $\forall\alpha\in[0,\ 1],\ \widetilde{A}_\alpha\subset\widetilde{B}_\alpha$. If $i=0,1,...,n,\ \widetilde{A}_{i/n}\subset\widetilde{B}_{i/n}$, i.e. $[a_i^1,\ a_i^2]\subset[b_i^1,\ b_i^2]$. (i) implies $Z_n(\widetilde{A})\subset Z_n(\widetilde{B})$.

Conversely, for each $\beta\in[0,\ 1]$, let

$$\widetilde{A}_\beta=[a^1(\beta),\ a^2(\beta)],\ \widetilde{B}_\beta=[b^1(\beta),\ b^2(\beta)].$$

If $\widetilde{A}\not\subset\widetilde{B}$, there is $\alpha\in[0,\ 1]:\ \widetilde{A}_\alpha\not\subset\widetilde{B}_\alpha$. Then by $\widetilde{A}_\alpha=[a^1(\alpha),\ a^2(\alpha)],\ \widetilde{B}_\alpha=[b^1(\alpha),\ b^2(\alpha)]$, we have, either $a^1(\alpha)<b^1(\alpha)$ or $b^2(\alpha)<a^2(\alpha)$. It is no harm to assume $a^1(\alpha)<b^1(\alpha)$, and there are $x_0,\ y_0\in\mathbb{R}:\ \alpha=\widetilde{A}(x_0)=\widetilde{B}(y_0)$. Then α is a cluster point of the ranges of $\widetilde{A}(\cdot),\ \widetilde{B}(\cdot)$, respectively. And $a^1(\cdot),\ b^1(\cdot)$ are increasing and right continuous. Hence there are $m\in\mathbb{N}$, and $j\in\{1,...,m\}$, such that $\alpha\in\big[(j-1)/m,\ j/m\big]$, furthermore, $a^1\big(j/m\big)<b^1\big(j/m\big)$. Thus, $Z_m(\widetilde{A})\not\subset Z_m(\widetilde{B})$, which contradicts the assumptions. So $\widetilde{A}\subset\widetilde{B}$.

(iii) By Theorem 5.4, it follows that

$$\begin{aligned}D(Z_m(\widetilde{A}),\ Z_m(\widetilde{B})) &= \bigvee_{j=0}^{m}\{d_H(\widetilde{A}_{j/m},\ \widetilde{B}_{j/m})\}\\ &\le \bigvee_{\alpha\in[0,1]}\{d_H(\widetilde{A}_\alpha,\ \widetilde{B}_\alpha)\}=D(\widetilde{A},\ \widetilde{B}).\end{aligned}$$

Since $\widetilde{A}\in\mathcal{F}_{0c}(\mathbb{R})$, and let $\widetilde{A}_0=[c,\ d]$, $\mathrm{Ker}(\widetilde{A})=[a,\ b]$, we can easily show, $\widetilde{A}(\cdot)$ is strictly increasing on $[c,\ a]$, and strictly decreasing on $[b,\ d]$. For each $\alpha\in[0,\ 1]$, let $\widetilde{A}_\alpha=[a^1(\alpha),\ a^2(\alpha)]$, and $Z_n(\widetilde{A})_\alpha=[a_n^1(\alpha),\ a_n^2(\alpha)]$. There is $N\in\mathbb{N}$, so that $\forall n>N$, there exists $l_n\in\mathbb{N}:\ \alpha\in\big[(l_n-1)/n,\ l_n/n\big]$. Thus, by the facts that

$$\widetilde{A}_{(l_n-1)/n}=Z_n(\widetilde{A})_{(l_n-1)/n}\supset\widetilde{A}_\alpha\supset\widetilde{A}_{l_n/n}=Z_n(\widetilde{A})_{l_n/n},$$

and $Z_n(\widetilde{A})_{l_n/n} \subset Z_n(\widetilde{A})_\alpha \subset Z_n(\widetilde{A})_{(l_n-1)/n}$, we can get

$$\begin{aligned} & d_H\big(\widetilde{A}_\alpha,\ Z_n(\widetilde{A})_\alpha\big) \\ \le\ & d_H\big(\widetilde{A}_\alpha,\ \widetilde{A}_{l_n/n}\big)+d_H\big(\widetilde{A}_{l_n/n},\ Z_n(\widetilde{A})_{l_n/n}\big)+d_H\big(Z_n(\widetilde{A})_{l_n/n},\ Z_n(\widetilde{A})_\alpha\big) \\ =\ & d_H\big(\widetilde{A}_\alpha,\ \widetilde{A}_{l_n/n}\big)+d_H\big(Z_n(\widetilde{A})_{l_n/n},\ Z_n(\widetilde{A})_\alpha\big) \\ \le\ & d_H\big(\widetilde{A}_{(l_n-1)/n},\ \widetilde{A}_{l_n/n}\big)+d_H\big(\widetilde{A}_{(l_n-1)/n},\ \widetilde{A}_{l_n/n}\big) \\ =\ & 2\cdot d_H\big(\widetilde{A}_{l_n/n},\ \widetilde{A}_{(l_n-1)/n}\big). \end{aligned} \tag{5.23}$$

Since $l_n/n \longrightarrow \alpha$, $(l_n-1)/n \longrightarrow \alpha$ when $n \to +\infty$, and $l_n/n-(l_n-1)/n = 1/n$, using Theorem 5.7 we can show, $\forall\, \varepsilon > 0$, there is a $n_0 \in \mathbb{N}$, so that for each $n > n_0$, $d_H\big(\widetilde{A}_{l_n/n},\ \widetilde{A}_{(l_n-1)/n}\big) < \varepsilon/2$. By (5.23), $d_H\big(\widetilde{A}_\alpha,\ Z_n(\widetilde{A})_\alpha\big) < \varepsilon$. Hence when $n > n_0$ it follows that

$$D\big(\widetilde{A},\ Z_n(\widetilde{A})\big) \le \sup_{\alpha\in[0,1]} \big\{d_H\big(\widetilde{A}_\alpha,\ Z_n(\widetilde{A})_\alpha\big)\big\} \le \varepsilon.$$

Thus, $\lim\limits_{n\to+\infty} D\big(\widetilde{A},\ Z_n(\widetilde{A})\big) = 0$, which implies (iii). □

Next we illustrate the approximation of $\mathcal{F}_{0c}^{tn}(\mathbb{R})$ with arbitrary degree of accuracy to each element in $\mathcal{F}_{0c}(\mathbb{R})$ by two concrete fuzzy numbers $\widetilde{A}$, $\widetilde{B}$ defined respectively as follows:

$$\widetilde{A}\,(x) = \begin{cases} \exp\Big\{-\big(\dfrac{x-2}{2}\big)^2\Big\}, & 0 \le x \le 4, \\ 0 & \text{otherwise}; \end{cases}$$

$$\widetilde{B}\,(x) = \begin{cases} 1-(1-x)^2, & 0 \le x < 1, \\ 1, & 1 \le x \le 2, \\ \dfrac{4x-x^2}{4}, & 2 < x \le 4, \\ 0, & \text{otherwise}. \end{cases}$$

Then for $n \in \mathbb{N}$, and give the error bound $\varepsilon > 0$, by (5.23) we can show

$$\begin{cases} D\big(\widetilde{A},\ Z_n(\widetilde{A})\big) \le 2\cdot \bigvee\limits_{i=1}^{n} \big\{d_H\big(\widetilde{A}_{i/n},\ \widetilde{A}_{(i-1)/n}\big)\big\} \\ \qquad \le 4\cdot \Big(\bigvee\limits_{i=\mathrm{Int}(ne^{-1})}^{n} \Big\{\sqrt{\ln n - \ln(i-1)} - \sqrt{\ln n - \ln i}\Big\}\Big), \\ D\big(\widetilde{B},\ Z_n(\widetilde{B})\big) \le 2\cdot \bigvee\limits_{i=1}^{n} \big\{d_H\big(\widetilde{B}_{i/n},\ \widetilde{B}_{(i-1)/n}\big)\big\} \\ \qquad = 2\cdot \bigvee\limits_{i=1}^{n} \Big\{\Big|\sqrt{1-\dfrac{i-1}{n}} - \sqrt{-\dfrac{i}{n}}\Big|\Big\}, \end{cases} \tag{5.24}$$

where $\text{Int}(ne^{-1})$ means the integer part of ne^{-1}. In (5.24) we can imply the following inequality:

$$\begin{aligned}&\bigvee_{i=\text{Int}(ne^{-1})}^{n}\left\{\sqrt{\ln n-\ln(i-1)}-\sqrt{\ln n-\ln i}\right\}\\ =&\bigvee_{i=\text{Int}(ne^{-1})}^{n}\left\{\frac{\ln i-\ln(i-1)}{\sqrt{\ln n-\ln(i-1)}+\sqrt{\ln n-\ln i}}\right\}\leq\frac{1}{ne^{-1}-1}\cdot\frac{1}{\sqrt{2\ln\frac{n}{n-1}}}.\end{aligned}\tag{5.25}$$

Since when $n\geq 2$, $2(n-1)\ln\big(1+1/(n-1)\big)>1$, using (5.25) we can show

$$\bigvee_{i=\text{Int}(ne^{-1})}^{n}\left\{\sqrt{\ln\frac{n}{i-1}}-\sqrt{\ln\frac{n}{i}}\right\}\leq\frac{1}{ne^{-1}-1}\cdot\sqrt{n-1}<\frac{3\sqrt{(n-1)}}{n-3}.$$

So if $n\geq 154/\varepsilon^2\geq 3$, we have $3\sqrt{(n-1)}/(n-3)<\varepsilon/4,\Longrightarrow D\big(\widetilde{A},\ Z_n(\widetilde{A})\big)<\varepsilon$. As for $\widetilde{B}$, we can imply the following facts:

$$D\big(\widetilde{B},\ Z_n(\widetilde{B})\big)\leq\varepsilon,\Longleftrightarrow n\geq\frac{2}{\varepsilon^2};\quad D\big(\widetilde{B},\ Z_n(\widetilde{B})\big)\leq\varepsilon,\Longleftrightarrow n\geq\frac{2}{\varepsilon^2}.$$

Therefore, with different error bounds ε's, we can get the corresponding n's for $Z_n(\widetilde{A})$ approximating $\widetilde{A}$, and $Z_n(\widetilde{B})$ approximating $\widetilde{B}$, respectively, as shown in following Table 5.1

Table 5.1 Values of n's corresponding to different error bounds ε's

	0.1	0.08	0.05	0.04	0.01
$\widetilde{A}$	1.54×10^4	2.32×10^4	6.16×10^4	9.63×10^4	1.54×10^6
$\widetilde{B}$	200	313	800	1250	20000

5.2.3 Extension operations based on polygonal fuzzy numbers

Let us now present some novel extension operations in $\mathcal{F}_{0c}^{tn}(\mathbb{R})$, they are fuzzy arithmetic '+', '−', '×' and '÷', which are somewhat different from ones based on Zadeh's extension principle [15, 47, 62]. Let $\widetilde{A},\ \widetilde{B}\in\mathcal{F}_{0c}^{tn}(\mathbb{R})$:

$$\widetilde{A}=([a_n^1,\ a_n^2];\ a_0^1,\dots,a_{n-1}^1,\ a_{n-1}^2,\dots,a_0^2),\ \widetilde{B}=([b_n^1,\ b_n^2];\ b_0^1,\dots,b_{n-1}^1,\ b_{n-1}^2,\dots,b_0^2).$$

Define the extension operations '+', '−', '×' as follows, respectively:

$$\begin{cases}\widetilde{A}+\widetilde{B}=([a_n^1+b_n^1,\ a_n^2+b_n^2];\ a_0^1+b_0^1,\dots,a_{n-1}^1+b_{n-1}^1,\ a_{n-1}^2+b_{n-1}^2,\dots,a_0^2+b_0^2),\\ \widetilde{A}-\widetilde{B}=([a_n^1-b_n^2,\ a_n^2-b_n^1];\ a_0^1-b_0^2,\dots,a_{n-1}^1-b_{n-1}^2,\ a_{n-1}^2-b_{n-1}^1,\dots,a_0^2-b_0^1),\\ \widetilde{A}\times\widetilde{B}=([c_n^1,\ c_n^2];\ c_0^1,\dots,c_{n-1}^1,\ c_{n-1}^2,\dots,c_0^2),\end{cases}\tag{5.26}$$

where $c_i^1,\ c_i^2\ (i=0,1,...,n)$ are determined by the interval multiplication [6, 18]: $[c_i^1,\ c_i^2]=[a_i^1,\ a_i^2]\times[b_i^1,\ b_i^2]$. If $\widetilde{B}=([b_n^1,\ b_n^2];\ b_0^1,...,b_{n-1}^1,\ b_{n-1}^2,...,b_0^2)\in\mathcal{F}_{0c}^{tn}(\mathbb{R})$, assume that either $\text{Supp}(\widetilde{B})\subset(0,\ +\infty)$ or $\text{Supp}(\widetilde{B})\subset(-\infty,\ 0)$, define $1/\widetilde{B}$ and $\widetilde{A}\div\widetilde{B}$ respectively as follows:

$$\frac{1}{\widetilde{B}}=\left(\left[\frac{1}{b_n^2},\ \frac{1}{b_n^1}\right];\frac{1}{b_0^2},...,\frac{1}{b_{n-1}^2},\ \frac{1}{b_{n-1}^1},...,\frac{1}{b_0^1}\right),\quad \widetilde{A}\div\widetilde{B}=\widetilde{A}\times\frac{1}{\widetilde{B}}. \tag{5.27}$$

If $\sigma:\ \mathbb{R}\longrightarrow\mathbb{R}$ is a monotone function, σ is extended as follows, $\sigma:\ \mathcal{F}_{0c}^{tn}(\mathbb{R})\longrightarrow\mathcal{F}_{0c}^{tn}(\mathbb{R})$, $\forall\ \widetilde{A}=([a_n^1,\ a_n^2];\ a_0^1,...,a_{n-1}^1,\ a_{n-1}^2,...,a_0^2)$, let

$$\sigma(\widetilde{A})=\begin{cases}\big([\sigma(a_n^1),\sigma(a_n^2)];\ \sigma(a_0^1),...,\sigma(a_{n-1}^1),\ \sigma(a_{n-1}^2),...,\sigma(a_0^2)\big),\sigma\text{ increasing};\\ \big([\sigma(a_n^2),\sigma(a_n^1)];\ \sigma(a_0^2),...,\sigma(a_{n-1}^2),\ \sigma(a_{n-1}^1),...,\sigma(a_0^1)\big),\sigma\text{ decreasing}.\end{cases} \tag{5.28}$$

Suppose that $a\in\mathbb{R}$, $a\cdot\widetilde{A}$ or $\widetilde{A}\cdot a$ are the scalar product of $\widetilde{A}$ by a:

$$a\cdot\widetilde{A}=\begin{cases}([aa_n^1,\ aa_n^2];\ aa_0^1,...,aa_{n-1}^1,\ aa_{n-1}^2,...,aa_0^2), & a\geq 0;\\ ([aa_n^2,\ aa_n^1];\ aa_0^2,...,aa_{n-1}^2,\ aa_{n-1}^1,...,aa_0^1), & a<0.\end{cases} \tag{5.29}$$

Obviously, if $\widetilde{A},\ \widetilde{B}\in\mathcal{F}_{0c}^{tn}(\mathbb{R})$, $a\in\mathbb{R}$, then $\widetilde{A}+\widetilde{B},\ \widetilde{A}-\widetilde{B},\ \widetilde{A}\times\widetilde{B},\ a\cdot\widetilde{A}\in\mathcal{F}_{0c}^{tn}(\mathbb{R})$. Further, if $\text{Supp}(\widetilde{B})\subset(0,\ +\infty)$ or $\text{Supp}(\widetilde{B})\subset(-\infty,\ 0)$, $\widetilde{A}\div\widetilde{B}\in\mathcal{F}_{0c}^{tn}(\mathbb{R})$. By (5.26)–(5.29), $\widetilde{A}+\widetilde{B}$, $\widetilde{A}-\widetilde{B}$ are identical with ones based Zadeh's extension principle in [18], and $\mathcal{F}_{0c}^{tn}(\mathbb{R})$ is closed under non-linear operations '$\times$', '$\div$' and $\sigma(\cdot)$, etc. These facts make it possible in $\mathcal{F}_{0c}^{tn}(\mathbb{R})$ to realize some nonlinear fuzzy operations, fast and accurately. Take '$\times$' as an example: let

$$\widetilde{A}=([0,1];-1,-0.5,1.5,2),\quad \widetilde{B}=([-0.5,0.5];-2,-1,1,2).$$

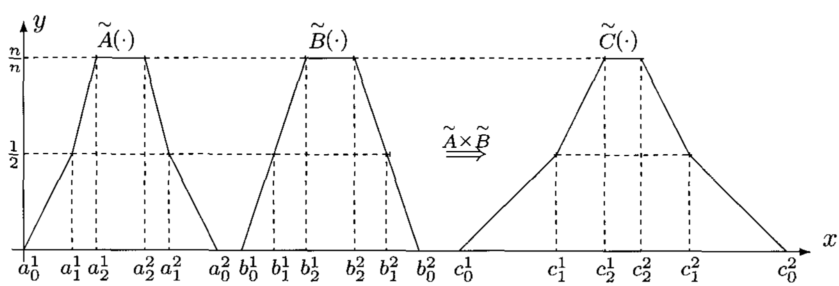


Figure 5.3 The multiplication $\widetilde{A}\times\widetilde{B}$ of $\widetilde{A}$ and $\widetilde{B}$

Then $\widetilde{C}\triangleq\widetilde{A}\times\widetilde{B}=([-0.5,\ 0.5];\ -4,-1.5,1.5,4)$, which is shown in Figure 5.3.

If $\widetilde{A}$, $\widetilde{B}$ $\mathcal{F}_{0c}(\mathbb{R})$, by $\widetilde{A}\,\tilde{+}\,\widetilde{B}$, $\widetilde{A}\,\tilde{-}\,\widetilde{B}$, $\widetilde{A}\,\tilde{\times}\,\widetilde{B}$, $\widetilde{A}\,\tilde{\div}\,\widetilde{B}$, we denote the extended addition, substraction, multiplication and division between $\widetilde{A}$, $\widetilde{B}$ based on Zadeh's principle [18], respectively. By (5.26)–(5.29), we have

$$\widetilde{A}\,\tilde{+}\,\widetilde{B}=\widetilde{A}+\widetilde{B},\ \widetilde{A}\,\tilde{-}\,\widetilde{B}=\widetilde{A}-\widetilde{B}$$

for each $\widetilde{A}$, $\widetilde{B}\in \mathcal{F}_{0c}^{tn}(\mathbb{R})$. Let us show now the respective relations between $+$ and $\tilde{+}$, $-$ and $\tilde{-}$, $\times$ and $\tilde{\times}$ through the operator $Z_n(\cdot)$.

Theorem 5.9 *Let* $\widetilde{A}$, $\widetilde{B}\in \mathcal{F}_{0c}(\mathbb{R})$, $n\in\mathbb{N}$. *Then we have*

(i) $Z_n(\widetilde{A}) = Z_n(\widetilde{B}) \iff \forall i\in\{0,1,...,n\}$, $\widetilde{A}_{i/n}=\widetilde{B}_{i/n} \iff Z_n(\widetilde{A})_{i/n} = Z_n(\widetilde{B})_{i/n}$ *for each* $i\in\{0,1,...,n\}$.

(ii) $Z_n(\widetilde{A}\,\tilde{+}\,\widetilde{B}) = Z_n(\widetilde{A}) + Z_n(\widetilde{B})$; $Z_n(\widetilde{A}\,\tilde{-}\,\widetilde{B}) = Z_n(\widetilde{A}) - Z_n(\widetilde{B})$; *And* $Z_n(\widetilde{A}\,\tilde{\times}\,\widetilde{B}) = Z_n(\widetilde{A})\times Z_n(\widetilde{B})$.

Proof. By (5.17) and Theorem 5.4, if $\widetilde{C}$, $\widetilde{D}\in \mathcal{F}_{0c}^{tn}(\mathbb{R})$, then we have, $\widetilde{C}=\widetilde{D}\iff \forall i\in\{0,1,...,n\}$, $\widetilde{C}_{i/n}=\widetilde{D}_{i/n}$, by which (i) follows directly.

(ii) At first easily we can show, $\widetilde{A}+\widetilde{B}$, $\widetilde{A}-\widetilde{B}$, $\widetilde{A}\times\widetilde{B}\in \mathcal{F}_{0c}(\mathbb{R})$. By (5.24) and (5.26)–(5.29), $\forall i\in\{0,1,...,n\}$, using the conclusions in [18], we have

$$Z_n(\widetilde{A}\,\tilde{+}\,\widetilde{B})_{i/n} = (\widetilde{A}\,\tilde{+}\,\widetilde{B})_{i/n} = \widetilde{A}_{i/n} + \widetilde{B}_{i/n} = Z_n(\widetilde{A})_{i/n} + Z_n(\widetilde{B})_{i/n}. \quad (5.30)$$

Therefore $Z_n(\widetilde{A}\,\tilde{+}\,\widetilde{B})_{i/n} = (Z_n(\widetilde{A}) + Z_n(\widetilde{B}))_{i/n}$ which implies $Z_n(\widetilde{A}\,\tilde{+}\,\widetilde{B}) = Z_n(\widetilde{A}) + Z_n(\widetilde{B})$ by (i). Similarly, $Z_n(\widetilde{A}\,\tilde{-}\,\widetilde{B}) = Z_n(\widetilde{A}) - Z_n(\widetilde{B})$. According to (5.25), we can prove the following equality:

$$\big(Z_n(\widetilde{A})\times Z_n(\widetilde{B})\big)_{i/n} = Z_n(\widetilde{A})_{i/n}\times Z_n(\widetilde{B})_{i/n}.$$

Hence similarly with (5.30), we have, $Z_n(\widetilde{A}\,\tilde{\times}\,\widetilde{B}) = Z_n(\widetilde{A})\times Z_n(\widetilde{B})$. □

Obviously, if $\widetilde{A}$, $\widetilde{B}\in \mathcal{F}_{0c}(\mathbb{R})$ and either $\mathrm{Supp}(\widetilde{B})\subset(0,+\infty)$ or $\mathrm{Supp}(\widetilde{B})\subset(-\infty,0)$, then we have for each $n\in\mathbb{N}$:

$$Z_n(\widetilde{A}\,\tilde{\div}\,\widetilde{B}) = Z_n(\widetilde{A})\div Z_n(\widetilde{B}).$$

The symmetric polygonal fuzzy numbers introduced in this section are general cases of triangular and trapezoidal fuzzy numbers. Moreover, their structures and representing forms are simple, and can provide with the approximations to a class of bounded fuzzy numbers, with any accuracy. Also we construct a novel fuzzy arithmetic that is convenient to realize. If we derive fuzzy neurons and fuzzy neural networks based on such a fuzzy arithmetic, the corresponding systems will possess strong learning capability. And fuzzy information processing can be realized, adaptively. The future research topics related include constructing a novel class of fuzzy neurons and fuzzy neural network based on symmetric polygonal fuzzy numbers. That will be an attractive field related to fuzzy neural networks and their applications.

§5.3 Polygonal FNN and learning algorithm

We call such a network system to be a polygonal FNN, that its connection weights and thresholds belong to $\mathcal{F}_{0c}^{tn}(\mathbb{R})$, its internal operations are based on (5.26)–(5.29). A polygonal fuzzy number is determined by finite points with some order relations. So the objective to study the learning algorithms related to such FNN's is to seek finite real numbers with the given order relations. Similarly with Chapter IV, we in the section employ the gradient descend method to develop another fuzzy BP algorithm for the polygonal FNN's. Since the internal operations in a polygonal FNN are defined by (5.26)–(5.29), the interval arithmetic [62] can be utilized to analyze the I/O relationships of the FNN's. By Algorithm 4.1 in §4.2, the first step to this end is to define a suitable error function $E(\cdot)$ and to develop some computation methods of partial derivatives of E with respect to the adjustable parameters. The basic tools to do this are the $\vee - \wedge$ function and Theorems 4.3–4.5 and Corollary 4.3. At first let us analyze the I/O relationships of polygonal FNN's and their expression forms, thoroughly.

5.3.1 Three layer feedforward polygonal FNN

In the subsection we focus on the single input and single output (SISO) polygonal FNN's with one hidden layer, whose structure is as follows: the output neuron is linear, the output fuzzy set is $\widetilde{Y}$, and all hidden neurons have the transfer function $\sigma:\ \mathbb{R} \longrightarrow \mathbb{R}$. Figure 5.4 illustrates the topological architecture of such a FNN, whose input signal $\widetilde{X}$ is a polygonal fuzzy number or belongs to $\mathcal{F}_{0c}(\mathbb{R})$. When $\widetilde{X}\in \mathcal{F}_{0c}(\mathbb{R})$, the input neuron has the polygonal operator $Z_n(\cdot)$.

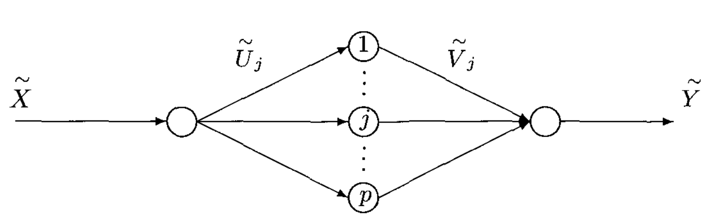


Figure 5.4 SISO three layer feedforward polygonal FNN

The fuzzy connection weights $\widetilde{U}_j$, $\widetilde{V}_j$ and fuzzy threshold $\widetilde{\Theta}_j$ in Figure 5.4 all belong to $\mathcal{F}_{0c}^{tn}(\mathbb{R})$. In the following of the section we assume σ to be continuous sigmoidal function, and to be differentiable on $\mathbb{R}$. The I/O relationship of the FNN in Figure 5.4 can be expressed as

$$\widetilde{Y} \triangleq F_{nn}(\widetilde{X}) = \sum_{j=1}^{p} \widetilde{V}_j \cdot \sigma(\widetilde{U}_j \cdot \widetilde{X} + \widetilde{\Theta}_j), \tag{5.31}$$

where the extension operations related are defined by (5.26)–(5.29). So the output fuzzy set $\widetilde{Y} \in \mathcal{F}_{0c}^{tn}(\mathbb{R})$. For $j = 1, ..., p$, set

$$\begin{cases} \widetilde{X} = \big([x_n^1, x_n^2]; x_0^1, ..., x_{n-1}^1, x_{n-1}^2, ..., x_0^2\big), \\ \widetilde{U}_j = \big([u_n^1(j), u_n^2(j)]; u_0^1(j), ..., u_{n-1}^1(j), u_{n-1}^2(j), ..., u_0^2(j)\big), \\ \widetilde{V}_j = \big([v_n^1(j), v_n^2(j)]; v_0^1(j), ..., v_{n-1}^1(j), v_{n-1}^2(j), ..., v_0^2(j)\big), \\ \widetilde{\Theta}_j = \big([\theta_n^1(j), \theta_n^2(j)]; \theta_0^1(j), ..., \theta_{n-1}^1(j), \theta_{n-1}^2(j), ..., \theta_0^2(j)\big). \end{cases} \tag{5.32}$$

For simplicity, we assume the input $\widetilde{X}$ of the polygonal FNN to belong to $\mathcal{F}_{0c}^{tn}(\mathbb{R}_+)$ or $\mathcal{F}_{0c}(\mathbb{R}_+)$. Denote

$$\widetilde{\mathcal{P}}[\sigma] = \Big\{F_{nn} : \mathcal{F}_{0c}^{tn}(\mathbb{R}_+) \longrightarrow \mathcal{F}_{0c}^{tn}(\mathbb{R}) \Big| F_{nn}(\widetilde{X}) = \sum_{j=1}^{p} \widetilde{V}_j \cdot \sigma(\widetilde{U}_j \cdot \widetilde{X} + \widetilde{\Theta}_j), \\ p \in \mathbb{N},\ \widetilde{V}_j,\ \widetilde{U}_j,\ \widetilde{\Theta}_j \in \mathcal{F}_{0c}^{tn}(\mathbb{R})\Big\};$$

$$\widetilde{\mathcal{Z}}[\sigma] = \Big\{T_{nn} : \mathcal{F}_{0c}(\mathbb{R}_+) \longrightarrow \mathcal{F}_{0c}^{tn}(\mathbb{R}) \Big| T_{nn}(\widetilde{X}) = \sum_{j=1}^{p} \widetilde{V}_j \cdot \sigma\big(\widetilde{U}_j \cdot Z_n(\widetilde{X}) + \widetilde{\Theta}_j\big), \\ n,\ p \in \mathbb{N},\ \widetilde{V}_j,\ \widetilde{U}_j,\ \widetilde{\Theta}_j \in \mathcal{F}_{0c}^{tn}(\mathbb{R})\Big\}.$$

Obviously, given $F_{nn} \in \widetilde{\mathcal{P}}[\sigma]$, then F_{nn} corresponds to a SISO three layer feedforward polygonal FNN, whose transfer function in the hidden layer is σ, and the neurons in input and output layers are linear, input signal belongs to $\mathcal{F}_{0c}^{tn}(\mathbb{R}_+)$, and output signal belongs to $\mathcal{F}_{0c}^{tn}(\mathbb{R})$, internal operations are defined by (5.26)–(2.29). If $T_{nn} \in \widetilde{\mathcal{Z}}[\sigma]$, also T_{nn} corresponds to a SISO three layer polygonal FNN, with the input including in $\mathcal{F}_{0c}(\mathbb{R}_+)$, output in $\mathcal{F}_{0c}^{tn}(\mathbb{R})$, and the input neuron having the polygonal line operator $Z_n(\cdot)$.

Theorem 5.10 *Let $p \in \mathbb{N}$, and $\widetilde{U}_j, \widetilde{V}_j, \widetilde{\Theta}_j \in \mathcal{F}_{0c}^{tn}(\mathbb{R})$ $(j = 1, ..., p)$. Then for each $\widetilde{X} \in \mathcal{F}_{0c}^{tn}(\mathbb{R})$: $\widetilde{X} = \big([x_n^1, x_n^2];\ x_0^1, ..., x_{n-1}^1,\ x_{n-1}^2, ..., x_0^2\big)$, the following facts hold:*

$$F_{nn}(\widetilde{X}) = \sum_{j=1}^{p} \widetilde{V}_j \cdot \sigma(\widetilde{U}_j \cdot \widetilde{X} + \widetilde{\Theta}_j) = \Big(\big[\gamma_n^1(x_n^1, x_n^2),\ \gamma_n^2(x_n^1, x_n^2)\big]; \\ \gamma_0^1(x_0^1, x_0^2), ..., \gamma_{n-1}^1(x_{n-1}^1, x_{n-1}^2),\ \gamma_{n-1}^2(x_{n-1}^1, x_{n-1}^2), ..., \gamma_0^2(x_0^1, x_0^2)\Big), \tag{5.33}$$

where $\gamma_i^k(x_i^1, x_i^2)$ $(k = 1, 2; i = 0, 1, ..., n)$ *can be expressed as*

$$\big[\gamma_i^1(x_i^1, x_i^2),\ \gamma_i^2(x_i^1, x_i^2)\big] = \Big[\sum_{j=1}^{p}\Big(\big\{v_i^1(j)\sigma\big(s_i^1(j)\big)\big\}\wedge\big\{v_i^1(j)\sigma\big(s_i^2(j)\big)\big\}\Big),$$
$$\sum_{j=1}^{p}\Big(\big\{v_i^2(j)\sigma\big(s_i^1(j)\big)\big\}\vee\big\{v_i^2(j)\sigma\big(s_i^2(j)\big)\big\}\Big)\Big]. \tag{5.34}$$

And $s_i^1(j)$, $s_i^2(j)$ *are defined as follows:*

$$\begin{cases} s_i^1(j) = \min\big\{u_i^1(j)x_i^1,\ u_i^1(j)x_i^2,\ u_i^2(j)x_i^1,\ u_i^2(j)x_i^2\big\}+\theta_i^1(j), \\ s_i^2(j) = \max\big\{u_i^1(j)x_i^1,\ u_i^1(j)x_i^2,\ u_i^2(j)x_i^1,\ u_i^2(j)x_i^2\big\}+\theta_i^2(j). \end{cases}$$

Proof. For $j = 1, ..., p$, let $\Gamma_j(\widetilde{X}) = \widetilde{V}_j \cdot \sigma(\widetilde{U}_j \cdot \widetilde{X} + \widetilde{\Theta}_j)$. If we denote

$$\widetilde{U}_j \cdot \widetilde{X} + \widetilde{\Theta}_j \triangleq \big([s_n^1(j),\ s_n^2(j)];\ s_0^1(j), ..., s_{n-1}^1(j),\ s_{n-1}^2(j), ..., s_0^2(j)\big),$$

and $\big[c_i^1(j),\ c_i^2(j)\big] = \big[u_i^1(j),\ u_i^2(j)\big]\times\big[x_i^1,\ x_i^2\big]$, then using the interval multiplication and (5.26) (5.32) we have

$$c_i^1(j) = \min\big\{u_i^1(j)x_i^1,\ u_i^1(j)x_i^2,\ u_i^2(j)x_i^1,\ u_i^2(j)x_i^2\big\},$$
$$c_i^2(j) = \max\big\{u_i^1(j)x_i^1,\ u_i^1(j)x_i^2,\ u_i^2(j)x_i^1,\ u_i^2(j)x_i^2\big\}.$$

By (5.26), $s_i^q(j) = c_i^q(j) + \theta_i^q(j)$ $(i = 0, 1, ..., n;\ q = 1, 2)$. So using (5.27) we can calculate $\Gamma_j(\widetilde{X})$ as

$$\widetilde{V}_j \cdot \big(\big[\sigma(s_n^1(j)),\ \sigma(s_n^2(j))\big];\ \sigma(s_0^1(j)), ..., \sigma(s_{n-1}^1(j)),\ \sigma(s_{n-1}^2(j)), ..., \sigma(s_0^2(j))\big),$$

where $\sigma(s_i^q(j)) \geq 0\,(q = 1, 2)$. Thus for $i = 0, 1, ..., n$, (5.26) (5.27) (5.32) imply

$$\big[\gamma_i^1(x_i^1, x_i^2),\ \gamma_i^2(x_i^1, x_i^2)\big] = \sum_{j=1}^{p}\Big(\big[v_i^1(j),\ v_i^2(j)\big]\times\big[\sigma(s_i^1(j)),\ \sigma(s_i^2(j))\big]\Big)$$
$$= \sum_{j=1}^{p}\big[\big\{v_i^1(j)\sigma(s_i^1(j))\big\}\wedge\big\{v_i^1(j)\sigma(s_i^2(j))\big\},\ \big\{v_i^2(j)\sigma(s_i^1(j))\big\}\vee\big\{v_i^2(j)\sigma(s_i^2(j))\big\}\big].$$

Therefore we can calculate the closed interval $\big[\gamma_i^1(x_i^1, x_i^2),\ \gamma_i^2(x_i^1, x_i^2)\big]$ as

$$\big[\gamma_i^1(x_i^1, x_i^2),\ \gamma_i^2(x_i^1, x_i^2)\big] = \Big[\sum_{j=1}^{p}\big\{v_i^1(j)\sigma(s_i^1(j))\big\}\wedge\big\{v_i^1(j)\sigma(s_i^2(j))\big\},$$
$$\sum_{j=1}^{p}\big\{v_i^2(j)\sigma(s_i^1(j))\big\}\vee\big\{v_i^2(j)\sigma(s_i^2(j))\big\}\Big],$$

which implies the following equality holds by (5.34):

$$F_{nn}(\widetilde{X}) = \big(\big[\gamma_n^1(x_n^1, x_n^2),\ \gamma_n^2(x_n^1, x_n^2)\big];$$
$$\gamma_0^1(x_0^1, x_0^2), ..., \gamma_{n-1}^1(x_{n-1}^1, x_{n-1}^2),\ \gamma_{n-1}^2(x_{n-1}^1, x_{n-1}^2), ..., \gamma_0^2(x_0^1, x_0^2)\big).$$

The theorem is proved. □

5.3.2 Learning algorithm

Next let us develop a fuzzy BP algorithm for the fuzzy weights and thresholds of the polygonal FNN's. Similarly with Algorithm 4.1 (see also [39]) we at first define a suitable error function E, then calculate the partial derivatives of E with respect to all adjustable parameters. And finally establish some iteration schemes of the parameters.

Given the fuzzy pattern pairs $\big(\widetilde{X}(1),\ \widetilde{O}(1)\big), ..., \big(\widetilde{X}(L),\ \widetilde{O}(L)\big)$ for training the FNN. That is, when the input of the polygonal FNN is $\widetilde{X}(l)$, the desired output is $\widetilde{O}(l)$, while $\widetilde{Y}(l)$ is the real output. Thus, $\widetilde{Y}(l) = F_{nn}\big(\widetilde{X}(l)\big)$, where $l = 1, ..., L$. In the following the fuzzy weights $\widetilde{V}_j$, $\widetilde{U}_j$ and fuzzy threshold $\widetilde{\Theta}_j$ are adjusted, rationally, so that for each $l = 1, ..., L$, we have, $\widetilde{Y}(l)$ can approximate $\widetilde{O}(l)$. To this end, for $l = 1, ..., L$, let

$$\widetilde{X}(l) = \big(\big[x_n^1(l), x_n^2(l)\big]; x_0^1(l), ..., x_{n-1}^1(l), x_{n-1}^2(l), ..., x_0^2(l)\big),$$
$$\widetilde{Y}(l) = \big(\big[y_n^1(l), y_n^2(l)\big]; y_0^1(l), ..., y_{n-1}^1(l), y_{n-1}^2(l), ..., y_0^2(l)\big),$$
$$\widetilde{O}(l) = \big(\big[o_n^1(l), o_n^2(l)\big]; o_0^1(l), ..., o_{n-1}^1(l), o_{n-1}^2(l), ..., o_0^2(l)\big).$$

For convenience of computing the partial derivatives related to the error function, we define a metric D_E between $\widetilde{X}$ and $\widetilde{Y}$:

$$D_E(\widetilde{X},\ \widetilde{Y}) = \Big(\sum_{i=0}^{n}\{d_E\big([x_i^1,\ x_i^2],\ [y_i^1,\ y_i^2]\big)\}^2\Big)^{\frac{1}{2}}.$$

where fuzzy sets $\widetilde{X},\ \widetilde{Y} \in \mathcal{F}_{0c}^{tn}(\mathbb{R})$, they are

$$\widetilde{X} = ([x_n^1,\ x_n^2];\ x_0^1, ..., x_{n-1}^1,\ x_{n-1}^2, ..., x_0^2),$$
$$\widetilde{Y} = ([y_n^1,\ y_n^2];\ y_0^1, ..., y_{n-1}^1,\ y_{n-1}^2, ..., y_0^2).$$

By the equivalence between d_H and d_E, and Theorem 5.4, we can get, the metrics D_E and D are equivalent. Define the error function E as follows:

$$E = \frac{1}{2}\sum_{l=1}^{L} D_E\big(\widetilde{O}(l), \widetilde{Y}(l)\big)^2 = \frac{1}{2}\sum_{l=1}^{L}\Big(\sum_{i=0}^{n}\{d_E\big([o_i^1(l), o_i^2(l)],\ [y_i^1(l), y_i^2(l)]\big)\}^2\Big). \tag{5.35}$$

Obviously $E = 0$ if and only if for each $l = 1, ..., L$, we have, $\widetilde{O}(l) = \widetilde{Y}(l)$. Since a symmetric polygonal fuzzy number can be determined by finite real parameters, we can develop some iteration schemes to update these parameters to get a new polygonal fuzzy number. Thus, an iteration learning algorithm for the fuzzy weights and thresholds can be established.

We write all adjustable parameters $u_i^q(j)$, $v_i^q(j)$, $\theta_i^q(j)$ $(i = 0, 1, ..., n;\ j = 1, ..., p;\ q = 1, 2)$ as a vector $\mathbf{w}$, consequently the error function E in (5.35) can be expressed as $E(\mathbf{w})$, where

$$\begin{aligned}\mathbf{w} &= \big(u_0^1(1), ...u_0^2(1), ..., u_0^1(p), ..., u_0^2(p),\ v_0^1(1), ..., v_0^2(1), ...,\\ &\qquad v_0^1(p), ..., v_0^2(p),\ \theta_0^1(1), ..., \theta_0^2(p)\big)\\ &\triangleq (w_1, ..., w_N).\end{aligned}$$

Theorem 5.11 *Let E be an error function defined by (5.35). Then $E = E(\mathbf{w})$ is differentiable almost everywhere (a.e.) in $\mathbb{R}^N$ with respect to Lebesgue measure. Moreover, for $l = 1, ..., L;\ i = 0, 1, ..., n;\ j = 1, ..., p$, if let*

$$\begin{cases} t_i^1(l) = o_i^1(l) - y_i^1(l),\\ t_i^2(l) = o_i^2(l) - y_i^2(l);\\ \Gamma_i^1(j,l) = \sigma'(s_i^1(j,l))v_i^1(j)\mathrm{lor}(v_i^1(j)),\ \Gamma_i^2(j,l) = \sigma'(s_i^2(j,l))v_i^1(j)\mathrm{lor}(-v_i^1(j)),\\ \Gamma_i^3(j,l) = \sigma'(s_i^1(j,l))v_i^2(j)\mathrm{lor}(-v_i^2(j)),\ \Gamma_i^4(j,l) = \sigma'(s_i^2(j,l))v_i^2(j)\mathrm{lor}(v_i^2(j));\\ \underline{\Delta}_i(j,l) = \big(\{u_i^1(j)x_i^1(l)\} \wedge \{u_i^2(j)x_i^1(l)\}\big) - \big(\{u_i^1(j)x_i^2(l)\} \wedge \{u_i^2(j)x_i^2(l)\}\big);\\ \overline{\Delta}_i(j,l) = \big(\{u_i^1(j)x_i^1(l)\} \vee \{u_i^2(j)x_i^1(l)\}\big) - \big(\{u_i^1(j)x_i^2(l)\} \vee \{u_i^2(j)x_i^2(l)\}\big);\\ S_u^1(i,j,l) = \sum\limits_{k=1}^{2} \mathrm{lor}\big((-1)^k\underline{\Delta}_i(j,l)\big)\mathrm{lor}\big((-1)^{q+1}x_i^k(l)\big)x_i^k(l);\\ S_u^2(i,j,l) = \sum\limits_{k=1}^{2} \mathrm{lor}\big((-1)^{3-k}\overline{\Delta}_i(j,l)\big)\mathrm{lor}\big((-1)^q x_i^k(l)\big)x_i^k(l). \end{cases} \tag{5.36}$$

We have the following partial derivative formula for each $i = 0, 1, ..., n;\ j = 1, ..., p;\ q = 1, 2:$

$$\text{(i)}\ \frac{\partial E}{\partial v_i^q(j)} = \sum_{l=1}^{m} t_i^q(l)\Big(\mathrm{lor}(-v_i^q(j))\sigma(s_i^{3-q}(j,l)) + \mathrm{lor}(v_i^q(j))\sigma(s_i^q(j,l))\Big);$$

$$\begin{aligned}\text{(ii)}\ \frac{\partial E}{\partial u_i^q(j)} = \sum_{l=1}^{m}\Big(&S_u^1(i,j,l)\cdot\big(t_i^1(l)\Gamma_i^1(j,l) + t_i^2(l)\Gamma_i^3(j,l)\big)\\ &+S_u^2(i,j,l)\cdot\big(t_i^1(l)\Gamma_i^2(j,l) + t_i^2(l)\Gamma_i^4(j,l)\big)\Big);\end{aligned}$$

$$\text{(iii)}\ \frac{\partial E}{\partial \theta_i^q(l)} = \sum_{l=1}^{m}\Big(\sum_{k=1}^{2} t_i^k(l)\mathrm{lor}\big((-1)^{q+2-k}v_i^k(j)\big)v_i^k(j)\Big)\sigma'(s_i^q(j,l)).$$

Proof. Similarly with Theorem 4.6 $E = E(\mathbf{w})$ is differentiable a.e. in $\mathbb{R}^N$ with respect to Lebesgue measure. To show (i)–(iii), since the proofs for (i) and (iii) are similar with one of (ii), it suffices to prove (ii). For $l=1,...,L;\ i=0,1,...,n$, we have

$$\big\{d_E\big([o_i^1(x),o_i^2(l)],\ [y_i^1(l),y_i^2(l)]\big)\big\}^2=\big(o_i^1(l)-y_i^1(l)\big)^2+\big(o_i^2(l)-y_i^2(l)\big)^2,$$

By (5.35) the definition of E, it follows for $q=1,2$ that

$$\begin{aligned}\frac{\partial E}{\partial u_i^q(j)} &= \frac{1}{2}\sum_{l=1}^{L}\frac{\partial}{\partial u_i^q(j)}\Big(\sum_{i'=0}^{n}\big\{\big(o_{i'}^1(l)\big)-y_{i'}^1(l)\big)^2+\big(o_{i'}^2(l)-y_{i'}^2(l)\big)^2\big\}\Big)\\ &= \frac{1}{2}\sum_{l=1}^{L}\frac{\partial}{\partial u_i^q(j)}\Big(\big\{\big(o_i^1(l)-y_i^1(l)\big)^2+\big(o_i^2(l)-y_i^2(l)\big)^2\big\}\\ &\qquad+\Big\{\sum_{i'\neq i}\big\{\big(o_{i'}^1(l)-y_{i'}^1(l)\big)^2+\big(o_{i'}^2(l)-y_{i'}^2(l)\big)^2\big\}\Big\}\Big)\\ &= \sum_{l=1}^{L}\Big(t_i^1(l)\frac{\partial o_i^1(l)}{\partial u_i^q(j)}+t_i^2(l)\frac{\partial o_i^2(l)}{\partial u_i^q(j)}\Big).\end{aligned}\tag{2.37}$$

Theorem 5.10 implies the following equalities:

$$o_i^1(l)=\sum_{j'=1}^{p}\big(v_i^1(j')\sigma(s_i^1(j',l))\wedge v_i^1(j')\sigma(s_i^2(j',l))\big),$$

$$o_i^2(l)=\sum_{j'=1}^{p}\big(v_i^2(j')\sigma(s_i^1(j',l))\vee v_i^2(j')\sigma(s_i^2(j',l))\big).$$

where $s_i^1(j',l)$, $s_i^2(j',l)$ can be expressed as

$$\begin{cases} s_i^1(j',l)=u_i^1(j')x_i^1(l)\wedge u_i^1(j')x_i^2(l)\wedge u_i^2(j')x_i^1(l)\wedge u_i^2(j')x_i^2(l)+\theta_i^1(j');\\ s_i^2(j',l)=u_i^1(j')x_i^1(l)\vee u_i^1(j')x_i^2(l)\vee u_i^2(j')x_i^1(l)\vee u_i^2(j')x_i^2(l)+\theta_i^2(j');\end{cases}\tag{5.38}$$

So considering Corollary 4.3 and the fact $s_i^1(j,l)\le s_i^2(j,l)$, we can show

$$\begin{aligned}\frac{\partial o_i^1(l)}{\partial u_i^q(j)} &= \frac{\partial}{\partial u_i^q(j)}\Big(v_i^1(j)\sigma(s_i^1(j,l))\wedge v_i^1(j)\sigma(s_i^2(j,l))\Big)\\ &= \mathrm{lor}(-v_i^1(j))v_i^1(j)\frac{\partial\sigma(s_i^2(j,l)}{\partial u_i^q(j)}+\mathrm{lor}(v_i^1(j))v_i^1(j)\frac{\partial\sigma(s_i^1(j,l))}{\partial u_i^q(j)}\\ &= v_i^1(j)\Big(\mathrm{lor}(-v_i^1(j))\sigma'(s_i^2(j,l))\frac{\partial s_i^2(j,l)}{\partial u_i^q(j)}+\mathrm{lor}(v_i^1(j))\sigma'(s_i^1(j,l))\frac{\partial s_i^1(j,l)}{\partial u_i^q(j)}\Big).\end{aligned}\tag{5.39}$$

Similarly it follows that

$$\frac{\partial o_i^2(l)}{\partial u_i^q(j)}=v_i^2(j)\Big[\mathrm{lor}(-v_i^2(j))\sigma'(s_i^1(j,l))\frac{\partial s_i^1(j,l)}{\partial u_i^q(j)}+\mathrm{lor}(v_i^2(j))\sigma'(s_i^2(j,l))\frac{\partial s_i^2(j,l)}{\partial u_i^q(j)}\Big].\tag{5.40}$$

For $q = 1, 2$, by (5.39) (5.40) we get

$$
\begin{aligned}
& t_i^1(l)\frac{\partial o_i^1(l)}{\partial u_i^q(j)} + t_i^2(l)\frac{\partial o_i^2(l)}{\partial u_i^q(j)} \\
&= \sigma'(s_i^1(j,l))\Big(t_i^1(l)v_i^1(j)\mathrm{lor}(v_i^1(j)) + t_i^2(l)v_i^2(j)\mathrm{lor}(-v_i^2(j))\Big)\frac{\partial s_i^1(l)}{\partial u_i^q(j)} \\
&\quad + \sigma'(s_i^2(j,l))\Big(t_i^1(l)v_i^1(j)\mathrm{lor}(-v_i^1(j)) + t_i^2(l)v_i^2(j)\mathrm{lor}(v_i^2(j))\Big)\frac{\partial s_i^2(l)}{\partial u_i^q(j)} \\
&= \big(t_i^1(l)\Gamma_i^1(j,l) + t_i^2(l)\Gamma_i^3(j,l)\big)\frac{\partial s_i^1(l)}{\partial u_i^q(j)} + \big(t_i^1(l)\Gamma_i^2(j,l) + t_i^2(l)\Gamma_i^4(j,l)\big)\frac{\partial s_i^2(l)}{\partial u_i^q(j)}.
\end{aligned} \tag{5.41}
$$

Since $s_i^1(j,l) = \{u_i^1(j)x_i^1(l)\}\wedge\{u_i^2(j)x_i^1(l)\}\wedge\{u_i^1(j)x_i^2(l)\}\wedge\{u_i^2(j)x_i^2(l)\}$, by Corollary 4.3 it follows that

$$
\begin{aligned}
\frac{\partial s_i^1(l)}{\partial u_i^q(j)} = {} & \mathrm{lor}(\underline{\Delta}_i(j,l))\mathrm{lor}((-1)^q x_i^2(l)(u_i^1(j) - u_i^2(j)))x_i^2(l) \\
& + \mathrm{lor}(-\underline{\Delta}_i(j,l))\mathrm{lor}((-1)^q x_i^1(l)(u_i^1(j) - u_i^2(j)))x_i^1(l).
\end{aligned} \tag{5.42}
$$

Since $u_i^1(j) \le u_i^2(j)$, we can write (5.42) as

$$
\frac{\partial s_i^1(l)}{\partial u_i^q(j)} = \sum_{k=1}^{2}\mathrm{lor}\big((-1)^k\underline{\Delta}_i(j,l)\big)\mathrm{lor}\big((-1)^{3-q}x_i^k(l)\big)x_i^k(l). \tag{5.43}
$$

With the same reason we have

$$
\frac{\partial s_i^2(l)}{\partial u_i^q(j)} = \sum_{k=1}^{2}\mathrm{lor}\big((-1)^{3-k}\overline{\Delta}_i(j,l)\big)\mathrm{lor}((-1)^q x_i^k(l))x_i^k(l). \tag{5.44}
$$

We replace $\partial s_i^1(l)/\partial u_i^q(j)$, $\partial s_i^2(l)/\partial u_i^q(j)$ in (5.43) (5.44) into (5.40), and replace the result into (5.37), (ii) is ensured. □

To develop the learning algorithm for the polygonal FNN in (5.31), the first step is to define iteration schemes for parameter vector $\mathbf{w}$, i.e. the iteration laws of $u_i^q(j)$, $v_i^q(j)$ and $\theta_i^q(j)$. Based on Algorithm 4.3, the learning rate η and momentum constant α are improved, so that they are the functions of the iteration step t. Let

$$
\eta = \eta[t] = \rho_0\cdot\rho\big(E\mathbf{w}[t]\big) = \frac{\rho_0\cdot E\big(\mathbf{w}[t]\big)}{\big\|\nabla E\big(\mathbf{w}[t]\big)\big\|^2};\quad \alpha = \alpha[t] = \frac{\big|\triangle E\big(\mathbf{w}[t]\big)\big|}{\sum_{k=1}^{t-1}\big|\triangle E\big(\mathbf{w}[k]\big)}, \tag{5.45}
$$

where $\triangle E\big(\mathbf{w}[t]\big) = E\big(\mathbf{w}[t]\big) - E\big(\mathbf{w}[t-1]\big)$ $(t = 1, 2, ...)$, and $E\big(\mathbf{w}[0]\big)$ is a larger real number, for example, $E\big(\mathbf{w}[0]\big) = 200$; $\mathbf{w}[t]$ is parameter vector at iteration step t; and ρ_0 is a small constant, such as $\rho_0 = 0.01$; $\nabla E(\mathbf{w}[t])$ is gradient vector of E at step t, i.e. $\nabla E(\mathbf{w}[t]) = \left(\partial E(\mathbf{w})/\partial w_1, ..., \partial E(\mathbf{w})/\partial w_N\right)\big|_{\mathbf{w}=\mathbf{w}[t]}$. For $i = 0, 1, ..., n$; $j = 1, ..., p$; $q = 1, 2$, we denote

$$\begin{cases} a_i^q(j) = u_i^q(j)[t] - \dfrac{\rho_0 \cdot E(\mathbf{w}[t])}{\|\nabla E(\mathbf{w}[t])\|^2} \cdot \dfrac{\partial E}{\partial u_i^q(j)[t]} + \dfrac{|\triangle E(\mathbf{w}[t])|}{\sum\limits_{k=1}^{t-1} |\triangle E(\mathbf{w}[k])|} \cdot \triangle u_i^q(j)[t]; \\ b_i^q(j) = v_i^q(j)[t] - \dfrac{\rho_0 \cdot E(\mathbf{w}[t])}{\|\nabla E(\mathbf{w}[t])\|^2} \cdot \dfrac{\partial E}{\partial v_i^q(j)[t]} + \dfrac{|\triangle E(\mathbf{w}[t])|}{\sum\limits_{k=1}^{t-1} |\triangle E(\mathbf{w}[k])|} \cdot \triangle v_i^q(j)[t]; \\ c_i^q(j) = \theta_i^q(j)[t] - \dfrac{\rho_0 \cdot E(\mathbf{w}[t])}{\|\nabla E(\mathbf{w}[t])\|^2} \cdot \dfrac{\partial E}{\partial \theta_i^q(j)[t]} + \dfrac{|\triangle E(\mathbf{w}[t])|}{\sum\limits_{k=1}^{t-1} |\triangle E(\mathbf{w}[k])|} \cdot \triangle \theta_i^q(j)[t], \end{cases} \tag{5.46}$$

where $q = 1, 2$, $\triangle u_i^q(j)[t] = u_i^q(j)[t] - u_i^q(j)[t-1]$, $\triangle v_i^q(j)[t] = v_i^q(j)[t] - v_i^q(j)[t-1]$, and $\triangle \theta_i^q(j)[t] = \theta_i^q(j)[t] - \theta_i^q(j)[t-1]$. For each $j = 1, ..., p$, re-array $\{a_0^1(j), ..., a_n^1(j), a_n^2(j), ..., a_0^2(j)\}$, $\{b_0^1(j), ..., b_n^1(j), b_n^2(j), ..., b_0^2(j)\}$ and $\{c_0^1(j), ..., c_n^1(j), c_n^2(j), ..., c_0^2(j)\}$ respectively, with the increasing order as the sets $\{a_1(j), a_2(j), ..., a_{2n+2}(j)\}$, $\{b_1(j), b_2(j), ..., b_{2n+2}(j)\}$ and $\{c_1(j), c_2(j), ..., c_{2n+2}(j)\}$, that is

$$\begin{cases} a_1(j) \le a_2(j) \le \cdots \le a_{n+1}(j) \le \cdots \le a_{2n+2}(j), \\ b_1(j) \le b_2(j) \le \cdots \le b_{n+1}(j) \le \cdots \le b_{2n+2}(j), \\ c_1(j) \le c_2(j) \le \cdots \le c_{n+1}(j) \le \cdots \le c_{2n+2}(j). \end{cases} \tag{5.47}$$

Define $\widetilde{V}_j[t+1]$, $\widetilde{U}_j[t+1]$, $\widetilde{\Theta}_j[t+1] \in \mathcal{F}_{0c}^{tn}(\mathbb{R})$ respectively as follows:

$$\widetilde{U}_j[t+1] = \big([u_n^1(j)[t+1], u_n^2(j)[t+1]]; \\ u_0^1(j)[t+1], ..., u_{n-1}^1(j)[t+1], u_{n-1}^2(j)[t+1], ..., u_0^2(j)[t+1]\big),$$

$$\widetilde{V}_j[t+1] = \big([v_n^1(j)[t+1], v_n^2(j)[t+1]]; \\ v_0^1(j)[t+1], ..., v_{n-1}^1(j)[t+1], v_{n-1}^2(j)[t+1], ..., v_0^2(j)[t+1]\big),$$

$$\widetilde{\Theta}_j[t+1] = \big([\theta_n^1(j)[t+1], \theta_n^2(j)[t+1]]; \\ \theta_0^1(j)[t+1], ..., \theta_{n-1}^1(j)[t+1], \theta_{n-1}^2(j)[t+1], ..., \theta_0^2(j)[t+1]\big),$$

where $u_i^q(j)$, $v_i^q(j)$, $\theta_i^q(j)$ at the step $t+1$ are determined as follows, respectively:

$$\forall i = 0, 1, ..., n, \begin{cases} u_i^1(j)[t+1] = a_{i+1}(j), \ u_i^2(j)[t+1] = a_{2n+2-i}(j), \\ v_i^1(j)[t+1] = b_{i+1}(j), \ v_i^2(j)[t+1] = b_{2n+2-i}(j), \\ \theta_i^1(j)[t+1] = c_{i+1}(j), \ \theta_i^2(j)[t+1] = c_{2n+2-i}(j). \end{cases} \tag{5.48}$$

Thus, we can get the fuzzy weights $\widetilde{U}_j[t+1]$, $\widetilde{V}_j[t+1] \in \mathcal{F}_{0c}^{tn}(\mathbb{R})$, and fuzzy threshold $\widetilde{\Theta}_j[t+1] \in \mathcal{F}_{0c}^{tn}(\mathbb{R})$ $(j = 1, ..., p)$ at the iteration step $t+1$. Now we utilize (5.46)–(5.48) to develop a fuzzy BP algorithm.

Algorithm 5.2 Fuzzy BP algorithm with variable learning rate and momentum constant.

Step 1. Randomly choose initial fuzzy weights and threshold: $\widetilde{U}_j\,[0]$, $\widetilde{V}_j\,[0]$, $\widetilde{\Theta}_j\,[0]$ $(j = 1, ..., p)$, and put $t = 0$.

Step 2. For $j = 1, ..., p$, let

$$\begin{cases} \widetilde{U}_j\,[t] = \big(\big[u_n^1(j)[t], u_n^2(j)[t]\big]; u_0^1(j)[t], ..., u_{n-1}^1(j)[t], u_{n-1}^2(j)[t], ..., u_0^2(j)[t]\big), \\ \widetilde{V}_j\,[t] = \big(\big[v_n^1(j)[t], v_n^2(j)[t]\big]; v_0^1(j)[t], ..., v_{n-1}^1(j)[t], v_{n-1}^2(j)[t], ..., v_0^2(j)[t]\big), \\ \widetilde{\Theta}_j\,[t] = \big(\big[\theta_n^1(j)[t], \theta_n^2(j)[t]\big]; \theta_0^1(j)[t], ..., \theta_{n-1}^1(j)[t], \theta_{n-1}^2(j)[t], ..., \theta_0^2(j)[t]\big). \end{cases}$$

Step 3. For $j = 1, ..., p$; $i = 0, 1, ..., n$; $q = 1, 2$, complete the following value assignment:

$$v_i^q(j)[t] \longrightarrow v_i^q(j), \quad u_i^q(j)[t] \longrightarrow u_i^q(j), \quad \theta_i^q(j)[t] \longrightarrow \theta_i^q(j).$$

Step 4. For $l = 1, ..., L$; $j = 1, ..., p$; $i = 0, ..., n$ and $q = 1, 2$, using (5.36) we calculate the following values:

$$t_i^q(l)\ (q = 1, 2),\ \Gamma_i^q(j, l)\ (q = 1, ..., 4),\ S_u^q(i, j, l)\ (q = 1, 2),\ \underline{\Delta}_i(j, l),\ \overline{\Delta}_i(j, l).$$

Step 5. By Theorem 5.11 calculate $\partial E/\partial u_i^q(j)$, $\partial E/\partial v_i^q(j)$, $\partial E/\partial \theta_i^q(j)$. so by (5.44)–(5.46) compute $u_i^q(j)[t+1]$, $v_i^q(j)[t+1]$ and $\theta_i^q(j)[t+1]$.

Step 6. Discriminate $t > M$? or for $l = 1, ..., L$, discriminate $D(\widetilde{Y}_l, \widetilde{O}_l) < \varepsilon$? If yes go to Step 7; otherwise let $t = t+1$, go to Step 2.

Step 7. Output $\widetilde{V}_j[t]$, $\widetilde{\Theta}_j[t]$, $\widetilde{U}_j[t]$ $(j = 1, ..., p)$, and $\widetilde{O}_l$ $(l = 1, ..., L)$.

In Step 6, M is a prescribed upper bound of iteration steps, ε is error bound.

5.3.3 A simulation

In the subsection we illustrate an approximate realization of SISO fuzzy inference model by the polygonal FNN's, which can be applied, efficiently in many real applications, such as the control for water level of a container [43], the control for the temperature of a smelting furnace [26] and so on. The inference rule base $\{R_l | l = 1, ..., L\}$ consist in L fuzzy inference rules, which can expressed as 'IF ... THEN ...':

$$R_l\text{: IF } t \text{ is } \widetilde{X}(l), \text{ THEN } s \text{ is } \widetilde{O}(l),$$

where $l = 1, ..., L$, and let $L = 5$ in the following. $\widetilde{X}(l)$ is the antecedent fuzzy set, $\widetilde{O}(l)$ is the consequent fuzzy set, and $\widetilde{X}(l)$, $\widetilde{O}(l) \in \mathcal{F}_{0c}^{tn}(\mathbb{R})$, where $n = 2$.

The membership curves of $\tilde{X}(l)$ and $\tilde{O}(l)$ for $l = 1, ..., 5$ are shown in Figures 5.5 and 5.6, respectively.

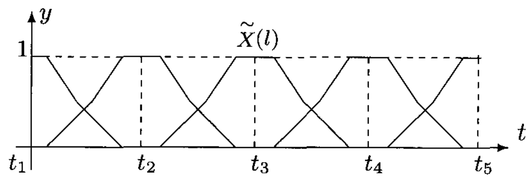


Figure 5.5 The antecedent fuzzy set of fuzzy inference rule

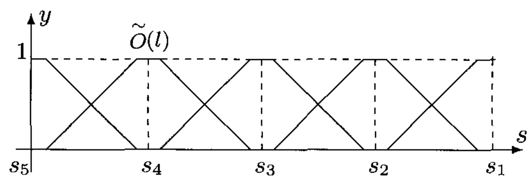


Figure 5.6 The consequent fuzzy set of fuzzy inference rule

Next we express the above antecedent and consequent pairs as input output pairs of a polygonal FNN, correspondingly. Thus, we can obtain the set of fuzzy patterns for training the FNN, that is $\mathcal{M} = \{(\tilde{X}(1), \tilde{O}(1)), ..., (\tilde{X}(5), \tilde{O}(5))\}$. Let $t_1 = s_1 = 0$, and $t_{i+1} - t_i = 0.6$, $s_i - s_{i+1} = 0.4$ $(i = 1, ..., 4)$ in Figures 5.5 and 5.6, respectively. $\tilde{X}(l)$ and $\tilde{O}(l)$ $(l = 1, ..., 5)$ are defined, respectively in Table 5.2.

Table 5.2 Training fuzzy patterns

$\tilde{X}(l)$	$\tilde{O}(l)$
$\tilde{X}(1) = ([0, 0.1]; 0, 0, 0.2, 0.5)$	$\tilde{O}(1) = ([0, 0.05]; 0, 0, 0.2, 0.35)$
$\tilde{X}(2) = ([0.5, 0.7]; 0.1, 0.4, 0.8, 1.1)$	$\tilde{O}(2) = ([0.35, 0.45]; 0.05, 0.2, 0.6, 0.75)$
$\tilde{X}(3) = ([1.1, 1.3]; 0.7, 1.0, 1.4, 1.7)$	$\tilde{O}(3) = ([0.75, 0.85]; 0.45, 0.6, 1.0, 1.15)$
$\tilde{X}(4) = ([1.7, 1.9]; 1.3, 1.6, 2.0, 2.3)$	$\tilde{O}(4) = ([1.15, 1.25]; 0.85, 1.0, 1.4, 1.55)$
$\tilde{X}(5) = ([2.3, 2.4]; 1.9, 2.2, 2.4, 2.4)$	$\tilde{O}(5) = ([1.55, 1.6]; 1.25, 1.4, 1.6, 1.6)$

From Table 5.2, $n = 2$. We choose the transfer function $\sigma : \mathbb{R} \longrightarrow \mathbb{R}$ as the following function, from which it follows that σ is continuously increasing and differentiable on $[0, +\infty)$, and is identical to zero on $(-\infty, 0]$:

$$\forall x \in \mathbb{R},\ \sigma(x) = \begin{cases} \dfrac{x^2}{1+x^2}, & x \geq 0, \\ 0, & x < 0. \end{cases}$$

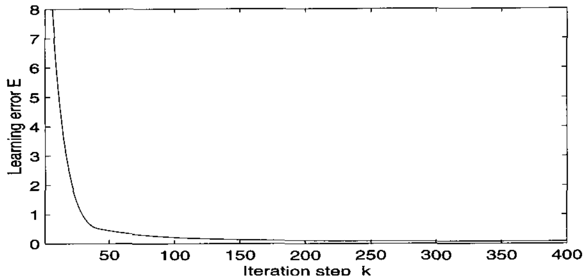


Figure 5.7 The error curve with changing learning rate and momentum constant

Choose $p = 30$, that is, there exist 30 neurons in the hidden layer of the polygonal FNN. The FNN is employed to realize above SISO fuzzy relationship, approximately. Using Algorithm 5.1, we can get the real output of the FNN corresponding to $\widetilde{X}(l)$ $(l = 1, ..., 5)$, after learning of 400 iteration steps:

$$\widetilde{Y}(1) = \big([0.0103, 0.1305]; 0.0061, 0.0076, 0.176, 0.486\big),$$

$$\widetilde{Y}(2) = \big([0.3181, 0.4603]; 0.0631, 0.232, 0.535, 0.9238\big),$$

$$\widetilde{Y}(3) = \big([0.7019, 0.865]; 0.3633, 0.5954, 0.9465, 1.3336\big),$$

$$\widetilde{Y}(4) = \big([1.1146, 1.2672]; 0.7726, 1.0106, 1.3413, 1.6780\big),$$

$$\widetilde{Y}(5) = \big([1.4936, 1.5646]; 1.204, 1.4086, 1.5698, 1.7053\big).$$

By the comparison between $\widetilde{Y}(l)$ and $\widetilde{O}(l)$, we obtain, the polygonal FNN can realize the given fuzzy inference rules with high degree of accuracy. Figure 5.7 illustrates the relation curve of E with respect to the iteration step t.

Similarly with Algorithm 4.3, we may show the convergence of Algorithm 5.2, whose convergent speed is as quick as ones of Algorithms 4.1, 4.3, as the simulation results showing. Also we can utilize GA to design some learning algorithms for the polygonal FNN's as in Algorithm 4.2 (see also [39, 60, 61]).

§5.4 Universal approximation of polygonal FNN

We may know from the simulation example of preceding section that a polygonal FNN can provide approximate realization of a family of fuzzy inference rules with given accuracy. Whether can the polygonal FNN's approximate any continuous fuzzy function defined on any compact set? That is, can the polygonal FNN's be universal approximators? In the section we focus on this problem, and study the approximating capability of the FNN's, thoroughly.

By $\mathcal{F}_{0c}(\mathbb{R}_+)$ we denote the collection of all non-negative fuzzy numbers in $\mathcal{F}_{0c}(\mathbb{R})$. That is, $\forall\ \widetilde{A} \in \mathcal{F}_{0c}(\mathbb{R}_+)$, we have $\forall x < 0,\ \widetilde{A}(x) = 0$.

5.4.1 I/O relationship analysis of polygonal FNN

Let $F:\ \mathcal{F}_{0c}^{tn}(\mathbb{R}_+)\longrightarrow\mathcal{F}_{0c}^{tn}(\mathbb{R})$ be a fuzzy function. $\widetilde{\mathcal{P}}[\sigma]$ is called to be universal approximator of F, if $\forall\varepsilon>0$, and for arbitrary compact set $\mathcal{U}\subset\mathcal{F}_{0c}^{tn}(\mathbb{R}_+)$, there is $F_{nn}\in\widetilde{\mathcal{P}}[\sigma]$, so that $\forall\ \widetilde{X}\in\mathcal{U},\ D\big(F_{nn}(\widetilde{X}),\ F(\widetilde{X})\big)<\varepsilon$. Similarly, we call $\widetilde{\mathcal{Z}}[\sigma]$ is universal approximator of $F:\ \mathcal{F}_{0c}(\mathbb{R}_+)\longrightarrow\mathcal{F}_{0c}^{tn}(\mathbb{R})$.

Using Theorem 5.4 and Theorem 5.8 we can get, if $F\in\widetilde{\mathcal{P}}[\sigma]$ (or $F\in\widetilde{\mathcal{Z}}[\sigma]$), the fuzzy function F is continuous on $\mathcal{F}_{0c}^{tn}(\mathbb{R}_+)$ (or $\mathcal{F}_{0c}(\mathbb{R}_+)$).

Theorem 5.12 *Let $F:\ \mathcal{F}_{0c}^{tn}(\mathbb{R}_+)\longrightarrow\mathcal{F}_{0c}^{tn}(\mathbb{R})$ be a fuzzy function. Then there exist $\widetilde{U},\ \widetilde{V},\ \widetilde{\Theta}\in\mathcal{F}_{0c}^{tn}(\mathbb{R})$, so that $F(\widetilde{X})=\widetilde{V}\cdot\sigma(\widetilde{U}\cdot\widetilde{X}+\widetilde{\Theta})$ if and only if the following conditions hold: $\forall\ \widetilde{X}=\big([x_n^1,\ x_n^2];x_0^1,...,x_{n-1}^1,\ x_{n-1}^2,...,x_0^2\big)\in\mathcal{F}_{0c}^{tn}(\mathbb{R}_+),\ F(\widetilde{X})\in\mathcal{F}_{0c}^{tn}(\mathbb{R})$ has the following representation:*

$$\begin{aligned}F(\widetilde{X})=\big(&[f_n^1(x_n^1,x_n^2),\ f_n^2(x_n^1,x_n^2)];\\&f_0^1(x_0^1,x_0^2),...,f_{n-1}^1(x_{n-1}^1,x_{n-1}^2),\ f_{n-1}^2(x_{n-1}^1,x_{n-1}^2),...,f_0^2(x_0^1,x_0^2)\big),\end{aligned}\tag{5.49}$$

where $f_i^q(x_i^1,x_i^2)=a(i,q)\sigma\big(b^1(i,q)x_i^1+b^2(i,q)x_i^2+\gamma(i,q)\big)$ $(i=0,1,...,n\,;q=1,2)$, and $a(i,q)$, $b^1(i,q)$, $b^2(i,q)$ and $\gamma(i,q)$ satisfying the following conditions: $\forall i\in\{0,1,...,n-1\}$, it follows that

$$\begin{cases}b^1(i,q)b^2(i,q)=0\ (q=1,2);\\ a(i,1)\le a(i+1,1)\le a(i+1,2)\le a(i,2);\\ a(i,1)\ge0,\Longrightarrow\begin{cases}0\le b^1(i,1)\le b^1(i+1,1),\ b^2(i,1)\le b^2(i+1,1)\le0,\\ b^1(i+1,2)\le b^1(i,2)\le0,\ 0\le b^2(i+1,2)\le b^2(i,2),\\ \gamma(i,1)\le\gamma(i+1,1)\le\gamma(i+1,2)\le\gamma(i,2),\\ b^1(i,1)\le b^2(i,2),\ b^2(i,1)\le b^1(i,2);\end{cases}\\ \begin{matrix}a(i,1)<0,\\ a(i,2)\ge0,\end{matrix}\Longrightarrow\begin{cases}0\ge b^1(i,1)\ge b^1(i+1,1),\ b^2(i,1)\ge b^2(i+1,1)\ge0,\\ 0\ge b^1(i,2)\ge b^1(i+1,2),\ b^2(i,2)\ge b^2(i+1,2)\ge0,\\ \gamma(i,1)\ge\gamma(i+1,1),\ \gamma(i,2)\ge\gamma(i+1,2),\\ \gamma(i,1)=\gamma(i,2),\ b^q(i,1)=b^q(i,2)\ (q=1,2);\end{cases}\\ a(i,2)<0,\Longrightarrow\begin{cases}b^1(i{+}1,1)\le b^1(i,1)\le0,\ 0\le b^2(i{+}1,1)\le b^2(i,1)\le0,\\ 0\le b^1(i,2)\le b^1(i+1,2),\ b^2(i,2)\le b^2(i+1,2)\le0,\\ b^1(i,2)\le b^2(i,1),\ b^2(i,2)\le b^1(i,1),\\ \gamma(i,2)\le\gamma(i+1,2)\le\gamma(i+1,1)\le\gamma(i,1).\end{cases}\end{cases}\tag{5.50}$$

Proof. Necessity: Let $F(\widetilde{X}) = \widetilde{V} \cdot \sigma(\widetilde{U} \cdot \widetilde{X} + \widetilde{\Theta})$ ($\widetilde{X} \in \mathcal{F}_{0c}^{tn}(\mathbb{R}_+)$), Using (5.26)–(5.29) we have, (5.49) holds for $F(\widetilde{X})$. Let

$$\widetilde{V} = ([v_n^1, v_n^2]; v_0^1, ..., v_{n-1}^1, v_{n-1}^2, ..., v_0^2),\ \widetilde{\Theta} = ([\theta_n^1, \theta_n^2]; \theta_0^1, ..., \theta_{n-1}^1, \theta_{n-1}^2, ..., \theta_0^2),$$

$$\widetilde{U} = ([u_n^1, u_n^2]; u_0^1, ..., u_{n-1}^1, u_{n-1}^2, ..., u_0^2),$$

Then we have, $\widetilde{U} \cdot \widetilde{X} + \widetilde{\Theta} = \big([c_n^1, c_n^2]; c_0^1, ..., c_{n-1}^1, c_{n-1}^2, ..., c_0^2\big)$, where $[c_i^1, c_i^2] = \big([u_i^1, u_i^2] \times [x_i^1, x_i^2]\big) + [\theta_i^1, \theta_i^2]$. By the interval operation laws [31, 62], c_i^1, c_i^2 can be expressed as

$$c_i^1 = \min\{u_i^1 x_i^1, u_i^1 x_i^2, u_i^2 x_i^1, u_i^2 x_i^2\} + \theta_i^1,$$
$$c_i^2 = \max\{u_i^1 x_i^1, u_i^1 x_i^2, u_i^2 x_i^1, u_i^2 x_i^2\} + \theta_i^2.$$

Since $x_i^1, x_i^2 \geq 0$, we have, $c_i^1 = \min\{u_i^1 x_i^1, u_i^1 x_i^2\} + \theta_i^1$, $c_i^2 = \max\{u_i^2 x_i^1, u_i^2 x_i^2\} + \theta_i^2$, i.e.

$$c_i^1 = \begin{cases} u_i^1 x_i^1 + \theta_i^1, & u_i^1 \geq 0, \\ u_i^1 x_i^2 + \theta_i^1, & u_i^1 < 0; \end{cases} \qquad c_i^2 = \begin{cases} u_i^2 x_i^2 + \theta_i^2, & u_i^2 \geq 0; \\ u_i^2 x_i^1 + \theta_i^2, & u_i^2 < 0. \end{cases} \tag{5.51}$$

Therefore, $F(\widetilde{X}) = \widetilde{V} \cdot \big([s_n^1, s_n^2]; s_0^1, ..., s_{n-1}^1, s_{n-1}^2, ..., s_0^2\big)$, where $s_i^q = \sigma(c_i^q) \geq 0$ $(q = 1, 2)$. Thus, $\big[f_i^1(x_i^1, x_i^2),\ f_i^2(x_i^1, x_i^2\big] = [v_i^1, v_i^2] \times [s_i^1, s_i^2]$, i.e. for $i = 0, 1, ..., n$, we get

$$[f_i^1(x_i^1, x_i^2),\ f_i^2(x_i^1, x_i^2)] = \begin{cases} [v_i^1 s_i^1, v_i^2 s_i^2], & v_i^1 \geq 0, \\ [v_i^1 s_i^2, v_i^2 s_i^2], & v_i^1 < 0 \leq v_i^2, \\ [v_i^1 s_i^2, v_i^2 s_i^1], & v_i^2 < 0. \end{cases} \tag{5.52}$$

By the definition of $f_i^q(x_i^1, x_i^2)$ $(q = 1, 2)$ in (5.49) and (5.52), it follows for $i = 0, 1, ..., n$, if letting $a(i, 1) = v_i^1$, $a(i, 2) = v_i^2$, and

$$\gamma(i,1) = \begin{cases} \theta_i^1 & a(i,1) \geq 0, \\ \theta_i^2, & a(i,1) < 0; \end{cases} \qquad \gamma(i,2) = \begin{cases} \theta_i^2 & a(i,2) \geq 0, \\ \theta_i^1, & a(i,2) < 0. \end{cases} \tag{5.53}$$

$$b^1(i,1) = \begin{cases} u_i^1, & a(i,1) \geq 0,\ u_i^1 \geq 0, \\ u_i^2, & a(i,1) < 0,\ u_i^2 < 0, \\ 0 & \text{otherwise}; \end{cases} \quad b^2(i,1) = \begin{cases} u_i^1, & a(i,1) \geq 0,\ u_i^1 < 0, \\ u_i^2, & a(i,1) < 0,\ u_i^2 \geq 0, \\ 0 & \text{otherwise}. \end{cases} \tag{5.54}$$

$$b^1(i,2) = \begin{cases} u_i^2, & a(i,2) \geq 0,\ u_i^2 < 0, \\ u_i^1, & a(i,2) < 0,\ u_i^1 \geq 0, \\ 0 & \text{otherwise}; \end{cases} \quad b^2(i,2) = \begin{cases} u_i^2, & a(i,2) \geq 0,\ u_i^2 \geq 0, \\ u_i^1, & a(i,2) < 0,\ u_i^1 < 0, \\ 0 & \text{otherwise}. \end{cases} \tag{5.55}$$

we can obtain (5.50) by (5.53)–(5.55).

Sufficiency: By (5.50) for $F(\widetilde{X})$ it follows that

$$f_i^1(x_i^1,x_i^2) \le f_{i+1}^1(x_{i+1}^1,x_{i+1}^2) \le f_{i+1}^2(x_{i+1}^1,x_{i+1}^2) \le f_i^2(x_i^1,x_i^2).$$

For $i = 0,1,...,n$, define v_i^1, v_i^2; u_i^1, u_i^2; θ_i^1, θ_i^2 as follows, respectively: $v_i^1 = a(i,1)$, $v_i^2 = a(i,2)$; and

$$\theta_i^1 = \begin{cases} \gamma(i,1), & a(i,1) \ge 0, \\ \gamma(i,2), & a(i,2) < 0, \\ 0, & \text{otherwise}; \end{cases} \quad \theta_i^2 = \begin{cases} \gamma(i,2), & a(i,1) \ge 0, \\ \gamma(i,1), & a(i,2) < 0, \\ 0, & \text{otherwise}; \end{cases}$$

$$u_i^1 = \begin{cases} b^1(i,1), & a(i,1) > 0,\ b^1(i,1) > 0, \\ b^2(i,1), & a(i,1) > 0,\ b^2(i,1) < 0, \\ b^1(i,2), & a(i,2) < 0,\ b^1(i,2) > 0, \\ b^2(i,2), & a(i,2) < 0,\ b^2(i,2) < 0, \\ 0 & \text{otherwise}; \end{cases}$$

$$u_i^2 = \begin{cases} b^1(i,1), & a(i,1) < 0,\ b^1(i,1) < 0, \\ b^2(i,1), & a(i,1) < 0,\ b^2(i,1) > 0, \\ b^1(i,2), & a(i,2) > 0,\ b^1(i,2) < 0, \\ b^2(i,2), & a(i,2) > 0,\ b^2(i,2) > 0, \\ 0 & \text{otherwise}. \end{cases}$$

Using (5.50) we can show, $\forall i = 0,1,...,n-1$, it follows that

$$v_i^1 \le v_{i+1}^1 \le v_{i+1}^2 \le v_i^2,\ u_i^1 \le u_{i+1}^1 \le u_{i+1}^2 \le u_i^2,\ \theta_i^1 \le \theta_{i+1}^1 \le \theta_{i+1}^2 \le \theta_i^2. \tag{5.56}$$

It suffices to show, $u_i^1 \le u_i^2$ $(i = 0,1,...,n)$ in (5.56) because the other inequalities can be proved, similarly. If $a(i,1) > 0$, then $a(i,2) > 0$. By (5.50), it follows that $b^1(i,1) \ge 0$, $b^2(i,1) \le 0$.

Case I, $b^1(i,1) = 0$: $b^2(i,1) = 0 \Longrightarrow u_i^2$ is either $b^2(i,2)$ or 0, for if $u_i^2 = b^1(i,2) < 0$, by (5.46) we have, $b^1(i,2)b^2(i,2) = 0$, implying $b^2(i,2) = 0$. Then by $f_i^1(x_i^1,x_i^2) \le f_i^2(x_i^1,x_i^2)$ $(x_i^1,\ x_i^2 \in \mathbb{R}_+)$ we can get contradiction. So $u_i^1 \le u_i^2$. If $b^2(i,1) < 0$, then u_i^2 is either $b^2(i,2)$ or $b^1(i,2)$, which can imply, $u_i^1 \le u_i^2$, since by (5.50), $b^2(i,2) \ge 0$ and $b^2(i,1) \le b^1(i,2)$.

case II, $b^1(i,1) > 0$: Using (5.50), we have $b^2(i,1) = 0$. Similarly with case I, $u_i^2 = b^2(i,2)$. Also by (5.50) it follows that, $b^1(i,1) \le b^2(i,2)$, so $u_i^1 \le u_i^2$.

As for the cases of $a(i,2) < 0$ or, $a(i,1) \le 0 \le a(i,2)$, similarly we can show, $u_i^1 \le u_i^2$.

Define respectively $\widetilde{V}$, $\widetilde{U}$, $\widetilde{\Theta}\in \mathcal{F}_{0c}^{tn}(\mathbb{R})$ as follows:

$$\begin{cases} \widetilde{V}=\big([v_n^1,\ v_n^2];\ v_0^1,...,v_{n-1}^1,\ v_{n-1}^2,...,v_0^2\big),\\ \widetilde{U}=\big([u_n^1,\ u_n^2];\ u_0^1,...,u_{n-1}^1,\ u_{n-1}^2,...,u_0^2\big),\\ \widetilde{\Theta}=\big([\theta_n^1,\ \theta_n^2];\ \theta_0^1,...,\theta_{n-1}^1,\ \theta_{n-1}^2,...,\theta_0^2\big).\end{cases}$$

By above discussions we can show, $F(\widetilde{X})=\widetilde{V}\cdot\sigma(\widetilde{U}\cdot\widetilde{X}+\widetilde{\Theta})$. The theorem is proved. □

Corollary 5.1 *Let $F_{nn}:\mathcal{F}_{0c}^{tn}(\mathbb{R}_+)\longrightarrow\mathcal{F}_{0c}^{tn}(\mathbb{R})$ be a fuzzy function. Then the following propositions hold:*

(i) If $F_{nn}\in\widetilde{\mathcal{P}}[\sigma]$, then $\forall\ \widetilde{A},\widetilde{B}\in\mathcal{F}_{0c}^{tn}(\mathbb{R}_+)$, $\widetilde{A}\subset\widetilde{B},\Longrightarrow F_{nn}(\widetilde{A})\subset F_{nn}(\widetilde{B})$;

(ii) $F_{nn}\in\widetilde{\mathcal{P}}[\sigma]\Longleftrightarrow\forall\ \widetilde{X}=([x_n^1,\ x_n^2];x_0^1,...,x_{n-1}^1,\ x_{n-1}^2,...,x_0^2)\in\mathcal{F}_{0c}^{tn}(\mathbb{R}_+)$, we have

$$\begin{aligned}F_{nn}(\widetilde{X})=\big(&[s_n^1(x_n^1,x_n^2),\ s_n^2(x_n^1,x_n^2)];\\ &s_0^1(x_0^1,x_0^2),...,s_{n-1}^1(x_{n-1}^1,x_{n-1}^2),\ s_{n-1}^2(x_{n-1}^1,x_{n-1}^2),...,s_0^2(x_0^1,x_0^2)\big),\end{aligned}\tag{5.57}$$

where $s_i^q(x_i^1,x_i^2)=\sum\limits_{j=1}^{p}v_j(i,q)\cdot\sigma(u_j^1(i,q)x_i^1+u_j^2(i,q)x_i^2+\theta_j(i,q))$ $(q=1,2;i=0,1,...,n)$ satisfying, $\forall j=1,...,p$, if let

$$a(i,q)=v_j(i,q),\ b^1(i,q)=u_j^1(i,q),\ b^2(i,q)=u_j^2(i,q),\ \gamma(i,q)=\theta_j(i,q),$$

(5.50) of Theorem 5.12 holds. So if $h_i^q(j)(x_i^1,x_i^2)\triangleq v_j(i,q)\cdot\sigma\big(u_j^1(i,q)x_i^1+u_j^2(i,q)x_i^2+\theta_j(i,q)\big)$, it follows that

$$h_i^1(j)(x_i^1,x_i^2)\le h_{i+1}^1(j)(x_{i+1}^1,x_{i+1}^2)\le h_{i+1}^2(j)(x_{i+1}^1,x_{i+1}^2)\le h_i^2(j)(x_i^1,x_i^2).$$

Proof. (i) For $\widetilde{X}=\big([x_n^1,\ x_n^2];\ x_0^1,...,x_{n-1}^1,\ x_{n-1}^2,...,x_0^2\big)\in\mathcal{F}_{0c}^{tn}(\mathbb{R}_+)$, let

$$F_{nn}(\widetilde{X})=\sum_{j=1}^{p}\widetilde{V}_j\cdot\sigma(\widetilde{U}_j\cdot\widetilde{X}+\widetilde{\Theta}_j).\tag{5.58}$$

Since $\widetilde{A}\subset\widetilde{B}$, using the extension criteria (5.26)–(5.29) and the interval arithmetic, we get, $\forall j=1,...,p$, $\widetilde{U}_j\cdot\widetilde{A}\subset\widetilde{U}_j\cdot\widetilde{B}$. Therefore

$$\widetilde{U}_j\cdot\widetilde{A}+\widetilde{\Theta}_j\subset\widetilde{U}_j\cdot\widetilde{B}+\widetilde{\Theta}_j),\Longrightarrow\sigma\big(\widetilde{U}_j\cdot\widetilde{A}+\widetilde{\Theta}_j\ \big)\subset\ \sigma\big(\widetilde{U}_j\cdot\widetilde{B}+\widetilde{\Theta}_j\ \big).$$

With the same reason, $\sum\limits_{j=1}^{p}\widetilde{V}_j\cdot\sigma\big(\widetilde{U}_j\cdot\widetilde{A}+\widetilde{\Theta}_j\ \big)\subset\ \sum\limits_{j=1}^{p}\widetilde{V}_j\cdot\sigma\big(\widetilde{U}\cdot\widetilde{B}+\widetilde{\Theta}_j\ \big)$, that is, $F_{nn}(\widetilde{A})\subset F_{nn}(\widetilde{B})$.

(ii) Necessity: Let $F_{nn} \in \widetilde{\mathcal{P}}[\sigma]$, so that $F_{nn}(\widetilde{X})$ can be expressed as (5.58). For $j \in \{1, ..., p\}$, denote $H_j(\widetilde{X}) = \widetilde{V}_j \cdot \sigma(\widetilde{U}_j \cdot \widetilde{X} + \widetilde{\Theta}_j)$. By Theorem 5.12, if let $\widetilde{X} = ([x_n^1, x_n^2];\ x_0^1, ..., x_{n-1}^1, x_{n_1}^2, ..., x_0^2) \in \mathcal{F}_{0c}^{tn}(\mathbb{R})$, then

$$H_j(\widetilde{X}) = \big([h_n^1(j)(x_n^1, x_n^2),\ h_n^2(j)(x_n^1, x_n^2)];\ h_0^1(j)(x_0^1, x_0^2), ..., \\ h_{n-1}^1(j)(x_{n-1}^1, x_{n-1}^2),\ h_{n-1}^2(j)(x_{n-1}^1, x_{n-1}^2), ..., h_0^2(j)(x_0^1, x_0^2)\big).$$

Moreover, $h_i^q(j)(x_i^1, x_i^2) = v_j(i,q) \cdot \sigma\big(u_j^1(i,q)x_i^1 + u_j^2(i,q)x_i^2 + \theta_j(i,q)\big)$ $(q = 1, 2; i = 0, 1, ..., n)$. Thus, by Theorem 5.12, if let

$$a(i,q) = v_j(i,q),\ b^1(i,q) = u_j^1(i,q),\ b^2(i,q) = u_j^2(i,q), \gamma(i,q) = \theta_j(i,q),$$

we can imply (5.50). Denote $s_i^q(x_i^1, x_i^2) = \sum_{j=1}^{p} s_i^q(j)(x_i^1, x_i^2)$. By (5.26)–(5.29) it follows that (5.57) holds, i.e.

$$F_{nn}(\widetilde{X}) = \big([s_n^1(x_n^1, x_n^2),\ s_n^2(x_n^1, x_n^2)];\ s_0^1(x_0^1, x_0^2), ..., \\ s_{n-1}^1(x_{n-1}^1, x_{n-1}^2),\ s_{n-1}^2(x_{n-1}^1, x_{n-1}^2), ..., s_0^2(x_0^1, x_0^2)\big),$$

Sufficiency: Using the assumption and Theorem 5.12, we get, $\forall j = 1, ..., p;\ i = 0, 1, ..., n$ if let

$$h_i^q(j)(x_i^1, x_i^2) = v_j(i,q) \cdot \sigma\big(u_j^1(i,q)x_i^1 + u_j^2(i,q)x_i^2 + \theta_j(i,q)\big)\ (q = 1, 2),$$

there are $\widetilde{V}_j$, $\widetilde{U}_j$ and $\widetilde{\Theta}_j \in \mathcal{F}_{0c}^{tn}(\mathbb{R})$, so that provided

$$H_j(\widetilde{X}) = \big([h_n^1(j)(x_n^1, x_n^2),\ h_n^2(j)(x_n^1, x_n^2)];\ h_0^1(j)(x_0^1, x_0^2), ..., \\ h_{n-1}^1(j)(x_{n-1}^1, x_{n-1}^2),\ h_{n-1}^2(j)(x_{n-1}^1, x_{n-1}^2), ..., h_0^2(j)(x_0^1, x_0^2)\big),$$

it follows that $H_j(\widetilde{X}) = \widetilde{V}_j \cdot \sigma(\widetilde{U}_j \cdot \widetilde{X} + \widetilde{\Theta}_j)$. Therefore

$$F_{nn}(\widetilde{X}) = \sum_{j=1}^{p} H_j(\widetilde{X}) = \sum_{j=1}^{p} \widetilde{V}_j \cdot \sigma(\widetilde{U}_j \cdot \widetilde{X} + \widetilde{\Theta}_j).$$

Hence $F_{nn} \in \widetilde{\mathcal{P}}[\sigma]$. the theorem is proved. □

Theorem 5.13 *Let* $F : \mathcal{F}_{0c}(\mathbb{R}_+) \longrightarrow \mathcal{F}_{0c}(\mathbb{R})$ *be a fuzzy function, and* $\widetilde{\mathcal{Z}}[\sigma]$ *be the universal approximator of* F. *Then* F *is increasing, that is, if* $\widetilde{A}, \widetilde{B} \in \mathcal{F}_{0c}(\mathbb{R}_+) : \widetilde{A} \subset \widetilde{B}$, *we have* $F(\widetilde{A}) \subset F(\widetilde{B})$.

Proof. By reduction to absurdity to show the conclusion. If the theorem does not hold, then there exist $\widetilde{A}, \widetilde{B} \in \mathcal{F}_{0c}(\mathbb{R}_+) : \widetilde{A} \subset \widetilde{B}$, but $F(\widetilde{A}) \not\subset F(\widetilde{B})$. By

Theorem 5.8, there is $n \in \mathbb{N}$, so that $Z_n\big(F(\widetilde{A})\big)\not\subset Z_n\big(F(\widetilde{B})\big)$. For $\widetilde{X}\in \mathcal{F}_{0c}(\mathbb{R}_+)$, we write

$$Z_n\big(F(\widetilde{X})\big)= \big([f_n^1(\widetilde{X}),\ f_n^2(\widetilde{X})];\ f_0^1(\widetilde{X}),...,f_{n-1}^1(\widetilde{X}),\ f_{n-1}^2(\widetilde{X}),...,f_0^2(\widetilde{X})\big),$$

there is $i \in \{0,1,...,n\}$, satisfying $f_i^1(\widetilde{A}) < f_i^1(\widetilde{B})$ or, $f_i^2(\widetilde{A}) > f_i^2(\widetilde{B})$. It is no harm to assume $f_i^1(\widetilde{A}) < f_i^1(\widetilde{B})$. Let $\mathcal{U} = \{\widetilde{A},\ \widetilde{B}\}$, $\varepsilon = \big(f_i^1(\widetilde{B}) - f_i^1(\widetilde{A})\big)/3$. By the assumption we have, there is $T_{nn} \in \widetilde{\mathcal{Z}}[\sigma]$, so that $D\big(F(\widetilde{X}),\ T_{nn}(\widetilde{X})\big)< \varepsilon$ for $\widetilde{X}=\widetilde{A}$, or $\widetilde{X}=\widetilde{B}$. Theorem 5.8 implying

$$D\big(Z_n(F(\widetilde{X})),\ Z_n(T_{nn}(\widetilde{X}))\big)\leq D\big(F(\widetilde{X}),\ T_{nn}(\widetilde{X})\big)< \varepsilon\ (\widetilde{X}\in \{\widetilde{A},\ \widetilde{B}\}).$$

For $\widetilde{X}\in \mathcal{F}_{0c}(\mathbb{R}_+)$, letting

$$Z_n\big(T_{nn}(\widetilde{X})\big)= \big([t_n^1(\widetilde{X}),t_n^2(\widetilde{X})];t_0^1(\widetilde{X}),...,t_{n-1}^1(\widetilde{X}),t_{n-1}^2(\widetilde{X}),...,t_0^2(\widetilde{X})\big),$$

we show by Theorem 5.8 that $\big|f_i^1(\widetilde{X}) - t_i^1(\widetilde{X})\big|\vee\big|f_i^2(\widetilde{X}) - t_i^2(\widetilde{X})\big|< \varepsilon$ for $\widetilde{X}=\widetilde{A}$ or, $\widetilde{X}=\widetilde{B}$. Therefore

$$-2\varepsilon < t_i^1(\widetilde{B}) - t_i^1(\widetilde{A}) + f_i^1(\widetilde{A}) - f_i^1(\widetilde{B}) < 2\varepsilon \Longrightarrow t_i^1(\widetilde{B}) > t_i^1(\widetilde{A}). \qquad (5.59)$$

Using Corollary 5.1 and Theorem 5.4 we get, $T_{nn}(\widetilde{A}) \subset T_{nn}(\widetilde{B})$, which contradict (5.59). Hence the proof is completed. □

As a consequence of Theorem 5.8 and Theorem 5.13, it follows that

Corollary 5.2 *Let* $F:\ \mathcal{F}_{0c}^{tn}(\mathbb{R}_+) \longrightarrow \mathcal{F}_{0c}^{tn}(\mathbb{R})$ *be a fuzzy function, and* $\widetilde{\mathcal{P}}[\sigma]$ *is universal approximator of* F. *Then* F *is increasing.*

Proof. If the conclusion does not hold, we have, $\widetilde{A},\ \widetilde{B}\in \mathcal{F}_{0c}^{tn}(\mathbb{R}_+) :\ \widetilde{A}\subset\widetilde{B}$, but $F(\widetilde{A}) \not\subset F(\widetilde{B})$. For $\widetilde{X}\in \mathcal{F}_{0c}^{tn}(\mathbb{R}_+))$, if let

$$F(\widetilde{X}) = \big([f_n^1(\widetilde{X}),f_n^2(\widetilde{X})];f_0^1(\widetilde{X}),...,f_{n-1}^1(\widetilde{X}),f_{n-1}^2(\widetilde{X}),...,f_0^2(\widetilde{X})\big),$$

there is $i \in \{0,1,...,n\}$, so that either $f_i^1(\widetilde{A}) < f_i^1(\widetilde{B})$ or, $f_i^2(\widetilde{A}) > f_i^2(\widetilde{B})$. it is no harm to assume $f_i^1(\widetilde{A}) < f_i^1(\widetilde{B})$. Choose the compact set $\mathcal{U} = \{\widetilde{A},\ \widetilde{B}\}$, $\varepsilon = \big(f_i^1(\widetilde{B}) - f_i^1(\widetilde{A})\big)/3$. By the assumption it follows that there is $F_{nn} \in \widetilde{\mathcal{P}}[\sigma]$, so that $D\big(F(\widetilde{X}),\ F_{nn}(\widetilde{X})\big)< \varepsilon$ if $\widetilde{X}=\widetilde{A}$ or, $\widetilde{X}=\widetilde{B}$. Thus, provided $F_{nn}(\widetilde{X})$ can be expressed as

$$F_{nn}(\widetilde{X}) = \big([s_n^1(\widetilde{A}),s_n^2(\widetilde{A})];s_0^1(\widetilde{A}),...,s_{n-1}^1(\widetilde{A}),s_{n-1}^2(\widetilde{A}),...,s_0^2(\widetilde{A})\big),$$

we have by Theorem 5.8, $|f_i^1(\widetilde{A}) - s_i^1(\widetilde{A})| < \varepsilon$, $|f_i^1(\widetilde{B}) - s_i^1(\widetilde{B})| < \varepsilon$. Therefore

$$-2\varepsilon < s_i^1(\widetilde{B}) - s_i^1(\widetilde{A}) + f_i^1(\widetilde{A}) - f_i^1(\widetilde{B}) < 2\varepsilon, \Longrightarrow s_i^1(\widetilde{B}) > s_i^1(\widetilde{A}).$$

However, $F_{nn}(\widetilde{A}) \subset F_{nn}(\widetilde{B})$, which is a contradiction! So the proof completed. □

Let us now study the approximation capability of a class of polygonal FNN's. That is done through the finite real values with given order relations, which correspond to the polygonal fuzzy numbers.

5.4.2 Approximation of polygonal FNN

In the following we focus on some subsets of $\widetilde{\mathcal{P}}[\sigma]$ or $\widetilde{\mathcal{Z}}[\sigma]$, where $\sigma(x) = 1/(1+\mathrm{e}^{-x})$. Define

$$\widetilde{\mathcal{P}}_0[\sigma] = \Big\{F:\ \mathcal{F}_{0c}^{tn}(\mathbb{R}_+) \longrightarrow \mathcal{F}_{0c}^{tn}(\mathbb{R}_+)\Big| F(\widetilde{X}) = \sum_{j=1}^{p} \widetilde{V}_j \cdot \sigma\big(u_j \cdot \widetilde{X} + \theta_j\big),$$
$$p \in \mathbb{N},\ \widetilde{V}_j \in \mathcal{F}_{0c}^{tn}(\mathbb{R}_+),\ \theta_j \in \mathbb{R},\ u_j \in \mathbb{R}_+\Big\};]$$
$$\widetilde{\mathcal{Z}}_0[\sigma] = \Big\{F:\ \mathcal{F}_{0c}(\mathbb{R}_+) \longrightarrow \mathcal{F}_{0c}^{tn}(\mathbb{R}_+)\Big| F(\widetilde{X}) = \sum_{j=1}^{p} \widetilde{V}_j \cdot \sigma\big(u_j \cdot Z_n(\widetilde{X}) + \theta_j\big),$$
$$n,\ p \in \mathbb{N},\ \widetilde{V}_j \in \mathcal{F}_{0c}^{tn}(\mathbb{R}_+),\ \theta_j \in \mathbb{R},\ u_j \in \mathbb{R}_+\Big\}.$$

As a direct result of Corollary 5.1 and Theorem 5.12 we can show

Corollary 5.3 *Let* $F:\ \mathcal{F}_{0c}^{tn}(\mathbb{R}_+) \longrightarrow \mathcal{F}_{0c}^{tn}(\mathbb{R}_+)$ *be a fuzzy function. Then* $F \in \widetilde{\mathcal{P}}_0[\sigma]$ *if and only if* $\forall\ \widetilde{X} = ([x_n^1,\ x_n^2];\ x_0^1, ..., x_{n-1}^1,\ x_{n-1}^2, ..., x_0^2) \in \mathcal{F}_{0c}^{tn}(\mathbb{R}_+)$, *it follows that*

$$F(\widetilde{X}) = \big([f_n^1(x_n^1),\ f_n^2(x_n^2)];\ f_0^1(x_0^1), ..., f_{n-1}^1(x_{n-1}^1),\ f_{n-1}^2(x_{n-1}^2), ..., f_0^2(x_0^2)\big),$$

where $f_i^1(x_i^1)$, $f_i^2(x_i^2)$ $(i = 0, 1, ..., n)$ *satisfying: there are* $p \in \mathbb{N}$, $\widetilde{V}_j \in \mathcal{F}_{0c}^{tn}(\mathbb{R}_+)$ *for each* $j \in \{1, ..., p\}$:

$$\widetilde{V}_j = \big([v_n^1(j),\ v_n^2(j)];\ v_0^1(j), ..., v_{n-1}^1(j),\ v_{n-1}^2(j), ..., v_0^2(j)\big) \in \mathcal{F}_{0c}^{tn}(\mathbb{R}_+),$$

and $u_j \in \mathbb{R}_+, \theta_j \in \mathbb{R}$, *so that* $f_i^1(x_i^1) = \sum\limits_{j=1}^{p} v_i^1(j)\sigma\big(u_j x_i^1 + \theta_j\big)$, $f_i^2(x_i^2) = \sum\limits_{j=1}^{p} v_i^2(j)\sigma\big(u_j x_i^2 + \theta_j\big)$.

Lemma 5.3 *Let* $F:\ \mathcal{F}_{0c}^{tn}(\mathbb{R}_+) \longrightarrow \mathcal{F}_{0c}^{tn}(\mathbb{R}_+)$ *be a fuzzy function, and* $F \in \widetilde{\mathcal{P}}_0[\sigma]$, *satisfying* $\forall\ \widetilde{X} \in \mathcal{F}_{0c}^{tn}(\mathbb{R}_+)$,

$$F(\widetilde{X}) = \big([f_n^1(x_n^1),\ f_n^2(x_n^2)];\ f_0^1(x_0^1), ..., f_{n-1}^1(x_{n-1}^1),\ f_{n-1}^2(x_{n-1}^2), ..., f_0^2(x_0^2)\big).$$

Then $\forall i \in \{0, 1, ..., n\}$, *both* $f_i^1(\cdot)$ *and* $f_i^2(\cdot)$ *are non-decreasing, moreover for arbitrary* $a_1, a_2 \in \mathbb{R}_+ : a_1 < a_2$, *and for each* $i \in \{1, ..., n\}$, *we have*

$$\begin{aligned} f_i^1(a_2) - f_i^1(a_1) &\le f_{i+1}^1(a_2) - f_{i+1}^1(a_1) \\ &\le f_{i+1}^2(a_2) - f_{i+1}^2(a_1) \le f_i^2(a_2) - f_i^2(a_1). \end{aligned} \tag{5.60}$$

Proof. Using Corollary 5.3 we get, $f_i^1(\cdot)$, $f_i^2(\cdot)$ are non-decreasing. Moreover, considering $\sigma(\cdot)$ is an increasing function we obtain

$$\begin{aligned} f_i^1(a_2) - f_i^1(a_1) &= \sum_{j=1}^{p} v_i^1(j)\{\sigma(u_j a_2 + \theta_j) - \sigma(u_j a_1 + \theta_j)\} \\ &\le \sum_{j=1}^{p} v_{i+1}^1(j)\{\sigma(u_j a_2 + \theta_j) - \sigma(u_j a_1 + \theta_j)\} \\ &= f_{i+1}^1(a_2) - f_{i+1}^1(a_1). \end{aligned}$$

Similarly the other inequalities in (5.60) hold. □

Let us now study the approximation capability of $\widetilde{\mathcal{P}}_0[\sigma]$ and $\widetilde{\mathcal{Z}}_0[\sigma]$ to the continuous fuzzy functions. To this end, we define the operators 'max' and 'min' in $\mathcal{F}_{0c}^{tn}(\mathbb{R})$. For $\widetilde{A}, \widetilde{B} \in \mathcal{F}_{0c}^{tn}(\mathbb{R})$:

$$\widetilde{A} = ([a_n^1, a_n^2]; a_0^1, ..., a_{n-1}^1, a_{n-1}^2, ..., a_0^2), \ \widetilde{B} = ([b_n^1, b_n^2]; b_0^1, ..., b_{n-1}^1, b_{n-1}^2, ..., b_0^2),$$

we define $\max(\widetilde{A}, \widetilde{B})$, $\min(\widetilde{A}, \widetilde{B}) \in \mathcal{F}_{0c}^{tn}(\mathbb{R})$ respectively as follows:

$$\max(\widetilde{A}, \widetilde{B}) = ([a_n^1 \vee b_n^1, a_n^2 \vee b_n^2]; a_0^1 \vee b_0^1, ..., a_{n-1}^1 \vee b_{n-1}^1, a_{n-1}^2 \vee b_{n-1}^2, ..., a_0^1 \vee b_0^1);$$

$$\min(\widetilde{A}, \widetilde{B}) = ([a_n^1 \wedge b_n^1, a_n^2 \wedge b_n^2]; a_0^1 \wedge b_0^1, ..., a_{n-1}^1 \wedge b_{n-1}^1, a_{n-1}^2 \wedge b_{n-1}^2, ..., a_0^1 \wedge b_0^1).$$

For fuzzy functions $F, G : \mathcal{F}_{0c}^{tn}(\mathbb{R}) \longrightarrow \mathcal{F}_{0c}^{tn}(\mathbb{R})$, define $\max(F, G)$, $\min(F, G)$: $\mathcal{F}_{0c}^{tn}(\mathbb{R}) \longrightarrow \mathcal{F}_{0c}^{tn}(\mathbb{R})$ respectively as follows: $\forall \widetilde{X} \in \mathcal{F}_{0c}^{tn}(\mathbb{R})$,

$$\max(F, G)(\widetilde{X}) = \max(F(\widetilde{X}), G(\widetilde{X})), \ \min(F, G)(\widetilde{X}) = \min(F(\widetilde{X}), G(\widetilde{X})). \tag{5.61}$$

Theorem 5.14 *Suppose* $F, G \in \widetilde{\mathcal{P}}_0[\sigma]$ *be fuzzy functions. Then* $\forall \varepsilon > 0$, *there is* $F_{nn} \in \widetilde{\mathcal{P}}_0[\sigma]$, *so that* $\forall \widetilde{X} \in \mathcal{F}_{0c}^{tn}(\mathbb{R}_+)$, $D(F_{nn}(\widetilde{X}), \max(F, G)(\widetilde{X})) < \varepsilon$. *Therefore,* $\widetilde{\mathcal{P}}_0[\sigma]$ *is universal approximator of* $\max(F, G)$. *Similarly,* $\widetilde{\mathcal{P}}_0[\sigma]$ *is also universal approximator of* $\min(F, G)$.

Proof. Give $\varepsilon > 0$. For $\widetilde{X} = ([x_n^1, x_n^2]; x_0^1, ..., x_{n-1}^1, x_{n-1}^2, ..., x_0^2) \in \mathcal{F}_{0c}^{tn}(\mathbb{R}_+)$, since $F, G \in \widetilde{\mathcal{P}}_0[\sigma]$, by Lemma 5.3 we may assume

$$F(\widetilde{X}) = ([f_n^1(x_n^1), f_n^2(x_n^2)]; f_0^1(x_0^1), ..., f_{n-1}^1(x_{n-1}^1), f_{n-1}^2(x_{n-1}^2), ..., f_0^2(x_0^2)),$$

$$G(\widetilde{X}) = ([g_n^1(x_n^1), g_n^2(x_n^2)]; g_0^1(x_0^1), ..., g_{n-1}^1(x_{n-1}^1), g_{n-1}^2(x_{n-1}^2), ..., g_0^2(x_0^2)).$$

Furthermore, there is $p \in \mathbb{N}$, so that for $q = 1,2;\ i = 0,1,...,n$, it follows that

$$\begin{cases} f_i^q(x) = \sum\limits_{j=1}^{p} v_i^q(f,j)\cdot\sigma\big(u_j(f)x+\theta_j(f)\big), \\ g_i^q(x) = \sum\limits_{j=1}^{p} v_i^q(g,j)\cdot\sigma\big(u_j(g)x+\theta_j(g)\big). \end{cases}$$

By (5.61), $\forall\ \widetilde{X} \in \mathcal{F}_{0c}^{tn}(\mathbb{R}_+)$, $\max(F,\ G)(\widetilde{X})$ can be represented as

$$\big(\big[r_n^1(x_n^1),\ r_n^2(x_n^2)\big];\ r_0^1(x_0^1),...,r_{n-1}^1(x_{n-1}^1),\ r_{n-1}^2(x_{n-1}^2),...,r_0^2(x_0^2)\big),$$

where for $q = 1,2;\ i = 0,1,...,n,\ r_i^q(x_i^q) = f_i^q(x_i^q) \vee g_i^q(x_i^q)$. Let

$$\mathcal{C}_0 = \big\{r_0^1,\ r_1^1,...,r_n^1,\ r_0^2,\ r_1^2,...,r_n^2\big\}.$$

Easily by Corollary 5.3 it follows that for $q \in \{1,2\},\ i \in \{0,1,...,n\}$, the limit $\lim\limits_{x\to+\infty} r_i^q(x)$ exists. Moreover, $\forall x \in \mathbb{R}_+,\ r_i^1(x) \le r_{i+1}^1(x) \le r_{i+1}^2(x) \le r_i^2(x)$. For each $a_1,\ a_2 \in \mathbb{R}_+ :\ a_1 < a_2$, and $i \in \{0,1,...,n-1\}$, next we shall show

$$r_i^1(a_2)-r_i^1(a_1) \le r_{i+1}^1(a_2)-r_{i+1}^1(a_1) \le r_{i+1}^2(a_2)-r_{i+1}^2(a_1) \le r_i^2(a_2)-f_i^2(a_1). \tag{5.62}$$

At first by Lemma 5.3 we obtain

$$\begin{aligned} f_i^1(a_2)-f_i^1(a_1) &\le f_{i+1}^1(a_2)-f_{i+1}^1(a_1) \\ &\le f_{i+1}^2(a_2)-f_{i+1}^2(a_1) \le f_i^2(a_2)-f_i^2(a_1), \\ g_i^1(a_2)-g_i^1(a_1) &\le g_{i+1}^1(a_2)-g_{i+1}^1(a_1) \\ &\le g_{i+1}^2(a_2)-g_{i+1}^2(a_1) \le g_i^2(a_2)-f_i^2(a_1). \end{aligned} \tag{5.63}$$

Choose $\mathcal{C}_{or} = \big\{f_i^q \vee g_i^q \big| q = 1,2;\ i = 0,1,...,n\big\}$. We can show that $\mathcal{C}_{or}$ is a quasi-difference order-preserved set. In fact, since $\sigma(x) = 1/(1+e^{-x})$, and $F,\ G \in \widetilde{\mathcal{P}}_0[\sigma]$, it follows by (5.60) that $\forall A > 0$, there is $m_0 \in \mathbb{N}$, so that

$$\forall i \in \{0,1,...,n\},\ q \in \{1,2\}, f_i^q \not\equiv g_i^q,\ \mathrm{Card}\big(\big\{x \in [0,A]\big|f_i^q(x) = g_i^q(x)\big\}\big)\le m_0.$$

There is $m_1 \in \mathbb{N}$, $\forall m \in \mathbb{N} :\ m > m_1$, if partition the interval $[0,\ A]$ into m equal length parts: $0 < A/m < \cdots < A(m-1)/m < A$, then $\forall i \in \{0,1,...,n\}$, $\forall q \in \{1,2\}$, $\forall k \in \{1,...,m\}$, at most there is one $x \in \big[(k-1)A/m,\ kA/m\big]$, satisfying $f_i^q(x) = g_i^q(x)$. Define the collection $\mathcal{O}$ of sub-intervals as follows:

$$\mathcal{O} = \Big\{\Delta = \Big[\frac{(k-1)A}{m},\ \frac{kA}{m}\Big]\Big|\ \exists q \in \{1,2\},\ i \in \{1,...,n\},\ x \in \Delta,\ f_i^q(x) = g_i^q(x)\Big\}.$$

Then $\mathrm{Card}(\mathcal{O}) \le m_0$. For arbitrary $a_1,\ a_2 \in \mathbb{R}_+ :\ |a_1 - a_2| < A/m$, if $a_1,\ a_2 \in \big[(k-1)A/m,\ kA/m\big]$, and $\big[(k-1)A/m,\ kA/m\big]\notin \mathcal{O}$, then either $f_i^q(a_1) >$

$g_i^q(a_1),\ f_i^q(a_2) > g_i^q(a_2)$ or, $f_i^q(a_1) < g_i^q(a_1),\ f_i^q(a_2) < g_i^q(a_2)$. By (5.63), it can imply (5.62) holds at this two cases. Thus, $\mathcal{C}_{or}$ is a quasi-difference order-preserved set. By Theorem 5.3, $\forall\varepsilon > 0$, there are $p \in \mathbb{N},\ u_j \in \mathbb{R}_+,\ \theta_j \in \mathbb{R}$, and order-preserved functional $\phi_j : \mathcal{C}_{or} \longrightarrow \mathbb{R}_+\ (j = 1, ..., q)$, such that $\forall i = 0, 1, ..., n;\ q = 1, 2$, the following inequality holds: $\forall x \in \mathbb{R}_+$,

$$\left|r_i^q(x) - \sum_{j=1}^{p} \phi_j\big(r_i^q\big)\sigma(u_j x + \theta_j)\right| = \left|f_i^q(x) \vee g_i^q(x) - \sum_{j=1}^{p} \phi_j\big(r_i^q\big)\sigma(u_j x + \theta_j)\right| < \varepsilon. \tag{5.64}$$

For $j \in \{1, ..., p\}$, define $\widetilde{V}_j \in \mathcal{F}_{0c}^{tn}(\mathbb{R}_+),\ \widetilde{\Theta}_j \in \mathcal{F}_{0c}^{tn}(\mathbb{R})$ as follows, respectively: $\widetilde{\Theta}_j = \theta_j$, and

$$\widetilde{V}_j = \big([\phi_j(r_n^1),\ \phi_j(r_n^2)];\ \phi_j(r_0^1), ..., \phi_j(r_{n-1}^1),\ \phi_j(r_{n-1}^2), ..., \phi_j(r_0^2)\big).$$

Easily by Corollary 5.3 we have, if let $\widetilde{F}_{nn}(\widetilde{X}) = \sum_{j=1}^{p} \widetilde{V}_j \cdot \sigma\big(u_j \cdot \widetilde{X} + \widetilde{\Theta}_j\big)$ for $\widetilde{X} \in \mathcal{F}_{0c}^{tn}(\mathbb{R}_+)$, then $\widetilde{F}_{nn} \in \widetilde{\mathcal{P}}_0[\sigma]$. Using Theorem 5.4 and (5.64) we can conclude that $\forall\ \widetilde{X} = ([x_n^1,\ x_n^2];\ x_0^1, ..., x_{n-1}^1,\ x_{n-1}^2, ..., x_0^2) \in \mathcal{F}_{0c}^{tn}(\mathbb{R}_+)$, it follows that $D\big(\max(F, G)(\widetilde{X}),\ \widetilde{F}_{nn}(\widetilde{X})\big) < \varepsilon$. Consequently, $\widetilde{\mathcal{P}}_0[\sigma]$ is universal approximator of $\max(F, G)$. Similarly $\widetilde{\mathcal{P}}_0[\sigma]$ is universal approximator of $\min(F, G)$. □

Assume that all the fuzzy functions $F_1, ..., F_m$ belong to $\widetilde{\mathcal{P}}_0[\sigma]$. And we write $F = \max(F_1, ..., F_m),\ G = \min(F_1, ..., F_m)$. By Theorem 5.14 and the induction method it follows that $\widetilde{\mathcal{P}}_0[\sigma]$ is universal approximator respectively to F, G. Theorem 5.14 plays a key role in the following research on approximating capability of polygonal FNN's. The proof of the following lemma is identical to one in [14, 31].

Lemma 5.4 *Let* $\widetilde{A},\ \widetilde{B} \in \mathcal{F}_{0c}^{tn}(\mathbb{R}_+),\ \widetilde{C},\ \widetilde{D} \in \mathcal{F}_{0c}^{tn}(\mathbb{R}_+)$, *and* $\widetilde{A} \subset \widetilde{B} \Longrightarrow \widetilde{C} \subset \widetilde{D}$; $\widetilde{B} \subset \widetilde{A} \Longrightarrow \widetilde{D} \subset \widetilde{C}$. *Then there is* $F_{nn} \in \widetilde{\mathcal{P}}_0[\sigma]$, *satisfying* $F_{nn}(\widetilde{A}) = \widetilde{C},\ F_{nn}(\widetilde{B}) = \widetilde{D}$.

Proof. Let $F_{nn}(\widetilde{X}) = \sum_{j=1}^{p} \widetilde{V}_j \cdot \sigma(u_j \cdot \widetilde{X} + \theta_j)\ (\widetilde{X} \in \mathcal{F}_{0c}^{tn}(\mathbb{R}_+))$. And denote

$$\widetilde{A} = \big([a_n^1,\ a_n^2];\ a_0^1, ..., a_{n-1}^1,\ a_{n-1}^2, ..., a_0^2\big),$$

$$\widetilde{B} = \big([b_n^1,\ b_n^2];\ b_0^1, ..., b_{n-1}^1,\ b_{n-1}^2, ..., b_0^2\big),$$

$$\widetilde{C} = \big([c_n^1,\ c_n^2];\ c_0^1, ..., c_{n-1}^1,\ c_{n-1}^2, ..., c_0^2\big),$$

$$\widetilde{D} = \big([d_n^1,\ d_n^2];\ d_0^1, ..., d_{n-1}^1,\ d_{n-1}^2, ..., d_0^2\big),$$

$$\widetilde{V}_j = \big([v_n^1(j),\ v_n^2(j)];\ v_0^1(j), ..., v_{n-1}^1(j),\ v_{n-1}^2(j), ..., v_0^2(j)\big).$$

where $j = 1, ..., p$. If $\widetilde{X} = ([x_n^1, x_n^2]; x_0^1, ..., x_{n-1}^1, x_{n-1}^2, ..., x_0^2) \in \mathcal{F}_{0c}^{tn}(\mathbb{R}_+)$, then we employ Corollary 5.3 to suppose

$$F_{nn}(\widetilde{X}) = \big([s_n^1(x_n^1), s_n^2(x_n^2)]; s_0^1(x_0^1), ..., s_{n-1}^1(x_{n-1}^1), s_{n-1}^2(x_{n-1}^2), ..., s_0^2(x_0^2)\big),$$

where $s_i^q(x_i^q) = \sum_{j=1}^{p} v_i^q(j) \cdot \sigma(u_j x_i^q + \theta_j)\,(q = 1, 2; i = 0, 1, ..., n)$. And $\forall i = 0, 1, ..., n$, by assumption there are infinite solutions of the following system of equations with respect to $v_i^q(j)$ $(q = 1, 2)$, and u_j, θ_j :

$$\begin{cases} s_i^1(a_i^1) = \sum_{j=1}^{p} v_i^1(j) \cdot \sigma(u_j a_i^1 + \theta_j) = c_i^1 \\ s_i^2(a_i^2) = \sum_{j=1}^{p} v_i^2(j) \cdot \sigma(u_j a_i^2 + \theta_j) = c_i^2 \\ s_i^1(b_i^1) = \sum_{j=1}^{p} v_i^1(j) \cdot \sigma(u_j b_i^1 + \theta_j) = d_i^1 \\ s_i^2(b_i^2) = \sum_{j=1}^{p} v_i^2(j) \cdot \sigma(u_j b_i^2 + \theta_j) = d_i^2. \end{cases}$$

Using Corollary 5.3 we choose an arbitrary solution , from which u_j, $\widetilde{V}_j$, θ_j can be determined. So the lemma holds. □

Theorem 5.15 *Suppose* $F : \mathcal{F}_{0c}^{tn}(\mathbb{R}_+) \longrightarrow \mathcal{F}_{0c}^{tn}(\mathbb{R}_+)$ *is a continuous fuzzy function. Then* $\widetilde{\mathcal{P}}_0[\sigma]$ *is the universal approximator of* F *if and only if* F *is increasing.*

Proof. Corollary 5.2 can ensure the necessity, So it suffices to prove the sufficiency. For arbitrary $\varepsilon > 0$, and compact set $\mathcal{U} \subset \mathcal{F}_{0c}^{tn}(\mathbb{R}_+)$, $\forall \widetilde{A} \in \mathcal{U}$, $\forall \widetilde{B} \in \mathcal{U}$, let $\widetilde{C} = F(\widetilde{A})$, $\widetilde{D} = F(\widetilde{B})$. By the assumption for $\widetilde{A}$, $\widetilde{B}$, $\widetilde{C}$, $\widetilde{D}$, the conditions of Lemma 5.4 hold, so there is fuzzy function $H[\widetilde{A}, \widetilde{B}] \in \widetilde{\mathcal{P}}[\sigma]$, so that the equalities, $H[\widetilde{A}, \widetilde{B}](\widetilde{A}) = F(\widetilde{A})$, $H[\widetilde{A}, \widetilde{B}](\widetilde{B}) = F(\widetilde{B})$ hold. By the continuity of $H[\widetilde{A}, \widetilde{B}]$ and F, there is $\delta > 0$, satisfying $D\big(F(\widetilde{X}), H[\widetilde{A}, \widetilde{B}](\widetilde{X})\big) < \varepsilon/4$ for each $\widetilde{X} \in \mathcal{V}_\delta(\widetilde{B}) \cap \mathcal{U}$, where $\mathcal{V}_\delta(\widetilde{B})$ is a δ−neighborhood of $\widetilde{B}$ in the metric space $(\mathcal{F}_{0c}^{tn}(\mathbb{R}_+), D)$. Since $\mathcal{U}$ is a compact set, there are $\widetilde{B}_1, ..., \widetilde{B}_m \in \mathcal{F}_{0c}^{tn}(\mathbb{R}_+)$, so that $\bigcup_{j=1}^{m} \mathcal{V}_\delta(\widetilde{B}_j) \supset \mathcal{U}$. It is no harm to assume $\widetilde{B}_1 = \widetilde{B}$. Choose $R[\widetilde{A}] = \max\big(H[\widetilde{A}, \widetilde{B}_1], ..., H[\widetilde{A}, \widetilde{B}_m]\big)$. By Theorem 5.14, there is $T[\widetilde{A}] \in \widetilde{\mathcal{P}}[\sigma]$, such that

$$\forall \widetilde{X} \in \mathcal{U}, \quad D\big(R[\widetilde{A}](\widetilde{X}), T[\widetilde{A}](\widetilde{X})\big) < \frac{\varepsilon}{4}. \tag{5.65}$$

For $\widetilde{X}\in \mathcal{F}_{0c}^{tn}(\mathbb{R}_+)$, put

$$\begin{cases} F(\widetilde{X}) = \big([f_n^1(\widetilde{X}),\, f_n^2(\widetilde{X})];\ f_0^1(\widetilde{X}),...,f_{n-1}^1(\widetilde{X}),\, f_{n-1}^2(\widetilde{X}),...,f_0^2(\widetilde{X})\big); \\ R[\widetilde{A}](\widetilde{X}) = \big([r_n^1(\widetilde{A})(\widetilde{X}),\, r_n^2(\widetilde{A})(\widetilde{X})]; \\ \qquad r_0^1(\widetilde{A})(\widetilde{X}),...,r_{n-1}^1(\widetilde{A})(\widetilde{X}),\, r_{n-1}^2(\widetilde{A})(\widetilde{X}),...,r_0^2(\widetilde{A})(\widetilde{X})\big); \\ T[\widetilde{A}](\widetilde{X}) = \big([t_n^1(\widetilde{A})(\widetilde{X}),\, t_n^2(\widetilde{A})(\widetilde{X})]; \\ \qquad t_0^1(\widetilde{A})(\widetilde{X}),...,t_{n-1}^1(\widetilde{A})(\widetilde{X}),\, t_{n-1}^2(\widetilde{A})(\widetilde{X}),...,t_0^2(\widetilde{A})(\widetilde{X})\big). \end{cases}$$

Using (5.65) and Theorem 5.4, we can show, $\forall\ \widetilde{X}\in \mathcal{U}, \forall i = 0,1,...,n,\ q \in \{1,2\},\ r_i^q(\widetilde{A})(\widetilde{X}) > f_i^q(\widetilde{X}) - \varepsilon/4$. Since $H[\widetilde{A},\widetilde{B}_j](\widetilde{A}) = F(\widetilde{A})\ (j = 1,...,m)$, we get, $R[\widetilde{A}](\widetilde{A}) = F(\widetilde{A})$. With the same reason, there is $\eta > 0$, so that

$$\forall\ \widetilde{X}\in \mathcal{V}_\eta(\widetilde{A}) \cap \mathcal{U}, D\big(F(\widetilde{X}), R[\widetilde{A}](\widetilde{X})\big) < \frac{\varepsilon}{4}. \tag{5.66}$$

We obtain $\widetilde{A}_1,...,\widetilde{A}_q\in \mathcal{U}:\ \bigcup_{j=1}^{q} \mathcal{V}_\eta(\widetilde{A}_j) \supset \mathcal{U}$. Let $G = \min\big(R[\widetilde{A}_1],...,R[\widetilde{A}_q]\big)$. By Theorem 5.14, there is $T_{nn} \in \widetilde{\mathcal{P}}[\sigma]$, satisfying $\forall\ \widetilde{X}\in \mathcal{U},\ D\big(G(\widetilde{X}),\ T_{nn}(\widetilde{X})\big)< \varepsilon/4$. Set

$$G(\widetilde{X}) = \big([g_n^1(\widetilde{X}),\, g_n^2(\widetilde{X})];\ g_0^1(\widetilde{X}),...,g_{n-1}^1(\widetilde{X}),\, g_{n-1}^2(\widetilde{X}),...,g_0^2(\widetilde{X})\big),$$
$$T_{nn}(\widetilde{X}) = \big([t_n^1(\widetilde{X}),\, t_n^2(\widetilde{X})];\ t_0^1(\widetilde{X}),...,t_{n-1}^1(\widetilde{X}),\, t_{n-1}^2(\widetilde{X}),...,t_0^2(\widetilde{X})\big).$$

So by the definition for G and (5.66) we get, $\forall q \in \{1,2\},\ i = 0,1,...,n,\ \forall\ \widetilde{X}\in \mathcal{U},\ g_i^q(\widetilde{X}) < f_i^q(\widetilde{X}) + \varepsilon/4$. Thus, $\forall q \in \{1,2\}$, it follows that

$$f_i^q(\widetilde{X}) - \frac{\varepsilon}{4} \le \bigwedge_{j=1}^{q}\{r_i^q(\widetilde{A}_j)(\widetilde{X})\}= g_i^q(\widetilde{X}) < t_i^q(\widetilde{X}) + \frac{\varepsilon}{4} < f_i^q(\widetilde{X}) + \frac{\varepsilon}{2}.$$

Hence $\forall\ \widetilde{X}\in \mathcal{U},\ q \in \{1,2\},\ i = 0,1,...,n,\ \big|f_i^q(\widetilde{X}) - t_i^q(\widetilde{X})\big|< \varepsilon/2$. Theorem 5.8 can imply the inequality: $D\big(F(\widetilde{X}),\ T_{nn}(\widetilde{X})\big)< \varepsilon$. That is, $\widetilde{\mathcal{P}}[\sigma]$ is the universal approximator of F. □

Lemma 5.5 *Let $\mathcal{U} \subset \mathcal{F}_{0c}(\mathbb{R})$ be a compact set. Then $\forall \varepsilon > 0$, there is $n \in \mathbb{N}$, so that $\forall\ \widetilde{A}\in \mathcal{U},\ D\big(\widetilde{A},\ Z_n(A)\big)< \varepsilon$.*

Proof. For arbitrary $\varepsilon > 0$, since $\mathcal{U} \subset \mathcal{F}_{0c}(\mathbb{R})$ is a compact set, there exist $m \in \mathbb{N}$, and $\widetilde{B}_1,...,\widetilde{B}_m\in \mathcal{U}$, satisfying $\bigcup_{j=1}^{m} \mathcal{V}_{\frac{\varepsilon}{3}}(\widetilde{B}_j) \supset \mathcal{U}$. By Theorem 5.8, it ensures to exist $n \in \mathbb{N}$, such that $\forall j = 1, ..., m$, $D(\widetilde{B}_j, \ Z_n(\widetilde{B}_j)) < \varepsilon/3$. Therefore, $\forall \ \widetilde{A} \in \mathcal{U}$, there is $j \in \{1, ..., m\}$, so that $\widetilde{A} \in \mathcal{V}_{\varepsilon/3}(\widetilde{B}_j)$. Considering $D(Z_n(\widetilde{B}_j), \ Z_n(\widetilde{A})) \leq D(\widetilde{B}_j, \ \widetilde{A})$, we get

$$\begin{aligned} D(\widetilde{A}, \ Z_n(\widetilde{A})) &\leq D(\widetilde{A}, \ \widetilde{B}_j) + D(\widetilde{B}_j, \ Z_n(\widetilde{B}_j)) + D(Z_n(\widetilde{B}_j), \ Z_n(\widetilde{A})) \\ &< \frac{\varepsilon}{3} + \frac{\varepsilon}{3} + \frac{\varepsilon}{3} = \varepsilon. \end{aligned}$$

So the lemma is proved. □

Theorem 5.16 *Let $F : \mathcal{F}_{0c}(\mathbb{R}_+) \longrightarrow \mathcal{F}_{0c}(\mathbb{R}_+)$ be a continuous fuzzy function. Then $\widetilde{\mathcal{Z}}_0[\sigma]$ is universal approximator of F if and only if F is increasing.*

Proof. Theorem 5.13 and Corollary 5.2 can ensure the necessity. it suffices to prove the sufficiency. For arbitrary compact $\mathcal{U} \subset \mathcal{F}_{0c}(\mathbb{R}_+)$ and $\varepsilon > 0$, the continuity of F implies, $F(\mathcal{U}) \triangleq \{F(\widetilde{X}) | \ \widetilde{X} \in \mathcal{U}\} \subset \mathcal{F}_{0c}(\mathbb{R})$ is a compact set. Moreover, by Theorem 5.8 it follows that there is $n \in \mathbb{N}$, and $\delta > 0$, such that

$$\forall \ \widetilde{X} \in \mathcal{U} \cup F(\mathcal{U}), \ D(Z_n(\widetilde{X}), \ \widetilde{X}) < \delta; \ \ \forall \ \widetilde{X} \in \mathcal{U}, \ D(F(\widetilde{X}), \ F(Z_n(\widetilde{X}))) < \frac{\varepsilon}{4}. \tag{5.67}$$

In $\mathcal{F}_{0c}^{tn}(\mathbb{R}_+)$ there is a compact set $\mathcal{U}_n$, satisfying $Z_n(\mathcal{U}) \triangleq \{Z_n(\widetilde{X}) | \ \widetilde{X} \in \mathcal{U}\} \subset \mathcal{U}_n$. Arbitrarily given $\widetilde{Y} \in \mathcal{F}_{0c}^{tn}(\mathbb{R}_+)$, let $G(\widetilde{Y}) = Z_n(F(\widetilde{Y}))$. Theorem 5.13 implies that G is continuous; Theorem 5.15 ensures to exist $F_{nn} \in \widetilde{\mathcal{P}}[\sigma]$, satisfying

$$\forall \ \widetilde{Y} \in \mathcal{U}_n, \ D(G(\widetilde{Y}), F_{nn}(\widetilde{Y})) < \frac{\varepsilon}{4}.$$

Let $T_{nn}(\widetilde{X}) = F_{nn}(Z_n(\widetilde{X})) \ (\widetilde{X} \in \mathcal{F}_{0c}(\mathbb{R}_+))$. Then $D(G(Z_n(\widetilde{X})), \ T_{nn}(\widetilde{X})) < \varepsilon/4$ for each $\widetilde{X} \in \mathcal{U}$, moreover, $T \in \widetilde{\mathcal{Z}}_0[\sigma]$. Therefore, $\forall \ \widetilde{X} \in \mathcal{U}$, by Theorem 5.8 and (5.67) it follows that

$$\begin{aligned} D(F(\widetilde{X}), \ T_{nn}(\widetilde{X})) \leq & D(F(\widetilde{X}), \ Z_n(F(\widetilde{X}))) + D(Z_n(F(\widetilde{X})), \ G(Z_n(\widetilde{X}))) \\ & + D(G(Z_n(\widetilde{X})), \ T_{nn}(\widetilde{X})) \leq \frac{\varepsilon}{4} + \frac{\varepsilon}{4} + \frac{\varepsilon}{4} < \varepsilon. \end{aligned}$$

i.e. $\widetilde{\mathcal{Z}}_0[\sigma]$ is the universal approximator of F. □

In the section the polygonal fuzzy numbers are employed to define the polygonal FNN's, which possess the following advantages: (i) In contrast to §4.5 and [41, 42], the equivalent conditions for the fuzzy functions, under which the polygonal FNN's can be universal approximators are much simpler. So the results are more applicable; (ii) As the FNN's with triangular or trapezoidal fuzzy number weights and thresholds [14], it is easy for the polygonal FNN's to develop learning algorithms for fuzzy weights and thresholds; (iii) Since the systems can directly process fuzzy information, they possess the stronger input output capability than general fuzzy systems [46]; (iv) Compared with with the regular FNN's based on Zadeh's extension principle, which are not universal approximators to continuous and increasing fuzzy functions, the systems are advantageous in approximation and learning capability.

Obviously the discussions are fit for the cases of negative fuzzy numbers and multiple-input and single-output. Although in practice much fuzzy information may be expressed as the positive or negative fuzzy numbers[14, 19–21], it is still an important problem how to generalize such fuzzy numbers to general cases.

References

[1] Baekeland R. & Kerre E., Piecewise linear fuzzy quantities: A way to implement fuzzy information into expert systems and fuzzy databases, in Bouchon B., Saitta L. & Yager R. R., Eds. *Uncertainty and Intelligent Systems— Lecture Notes in Computer Sciences,* Springer-Verlag, Berlin, 1988, **313**: 119–126.

[2] Buckley J. J. & Hayashi Y., Can neural nets be universal approximators for fuzzy functions? *Fuzzy Sets and Systems,*1999, **101**: 323–330.

[3] Buckley J. J. & Hayashi Y., Fuzzy input-output controllers are universal approximators, *Fuzzy Sets and Systems,* 1993, **58**: 273–278.

[4] Cai K. Y., *Introduction to fuzzy reliability,* Kluwer Academic Publishers, Boston–Dordrecht–London, 1996.

[5] Chen C. L., Lee C. S. & Kuo Y. H., Synthesis of weighted fuzzy mean filters with generic LR fuzzy cells, *4th International Conference on Soft Computing,* Iizuka, Fukuoka, Japan, September 30–October 5, 1996, **1**: 89–94.

[6] Chen S. H., Operations of fuzzy numbers with step form membership function using function principle, *Information Sciences,* 1998, **108**: 149–155.

[7] Chen S. M., A new method for evaluating weapon systems using fuzzy set theory, *IEEE Trans. on Systems, Man, and Cybernet., Part A,* 1996, **26**: 493–497.

[8] Chen T. P. & Liang Y., Neural networks calculate output capability of dynamical systems defined on whole space, *Science in China, Series E*(in Chinese), 1998, **28** (4): 338–349.

[9] Chen T. P., Chen H. & Liu R. W., Approximation capability in $C(\overline{\mathbb{R}}^n)$ by multilayer feedforward networks and related problems, *IEEE Trans. on Neural Networks,* 1995, **6**(1): 25–30.

[10] Diamond P. & Kloeden P., Characterization of compact subsets of fuzzy sets, *Fuzzy Sets and Systems,* 1989, **29**: 341–348.

[11] Diamond P. & Kloeden P., *Metric Spaces of Fuzzy Sets,* Singapore: World Scientific Publishing, 1994.

[12] Dubois D., Kerre E., Mesiar R. & Prade H., Fuzzy interval analysis, in Dubois D. & Prade H. Eds. *Fundamentals of Fuzzy Sets—The Handbooks of Fuzzy Sets Series,* Kluwer Academic Publishers, Boston–Mass, 2000, 483-581.

[13] Feuring T., Learning in fuzzy neural networks, *Proc.IEEE Internat.Conf.on Neural Networks,* 1996, **2**: 1061–1066.

[14] Feuring T. & Lippe W. M., The fuzzy neural network approximation lemma, *Fuzzy Sets and Systems,* 1999, **102**: 227–237.

[15] Gerla G. & Scarpati L., Extension principles for fuzzy set theory, *Information Sciences,* 1998, **106**: 49–69.

[16] Ghoshray S., fuzzy linear regression analysis by symmetric triangular fuzzy number coefficients, *Proc. of IEEE International Conference on Intelligent Engineering Systems,* 1997, **1**: 307–313.

[17] Ghyym S. H., A semi-linguistic fuzzy approach to multi-actor decision-making: application to aggregation of experts' judgments, *Annals of Nuclear Energy,* 1999, **26**: 1097–1112.

[18] Giachetti R. E. & Young R. E., Analysis of the error in the standard approximation used for multiplication of triangular and trapezoidal fuzzy numbers and the development of a new approximation, *Fuzzy Sets and Systems,* 1997, **91**: 1-13.

[19] Ishibuchi H., Fujioka R. & Tanaka H., Neural networks that learn from fuzzy if–then rules, *IEEE Trans. on Fuzzy Systems,* 1993, **1**: 85–97.

[20] Ishibuchi H., Kwon K. & Tanaka H. A., Learning algorithm of fuzzy neural networks with triangular fuzzy weights, *Fuzzy sets and Systems*, 1995, **71**: 277–293.

[21] Ishibuchi H., Morioka K. & Turksen I. B., Learning by fuzzified neural networks, *Internat. J. Approx. Reasoning,* 1995, **13**: 327–358.

[22] Ishibuchi H. & Nii M., Numerical analysis of the learning of fuzzified neural networks from fuzzy if-then rules, *Fuzzy Sets and Systems,* 2001, **120**: 281–307.

[23] Ishibuchi H. & Nii M., Fuzzy regression using asymmetric fuzzy coefficients and fuzzified neural networks, *Fuzzy Sets and Systems,* 2001, **119**: 273–290.

[24] Ishibuchi H. & Nii M., Fuzzy regression analysis by neural networks with non-symmetric fuzzy number weights, *Proc. of IEEE International Conference on Neural Networks,* 1996, **2**: 1191–1196.

[25] Ishibuchi H. & Nii M., Fuzzy regression analysis with non-symmetric fuzzy number coefficients and its neural network implementation, *Proc. of the Fifth IEEE International Conference on Fuzzy Systems,* 1996, **1**:318–324.

[26] Jang J. S. R., Sun C. T. & Mizutani E., *Neoro-fuzzy and soft computing,* Prentice Hall, 1997.

[27] Jiang Chuanhai, Approximation problems of neural networks, *Ann. of Math., Series A,*(in Chinese) 1998, **17**(3): 295–300.

[28] Kuo Y. H. & Chen C. L., Generic LR fuzzy cells for fuzzy hardware synthesis, *IEEE Trans. on Fuzzy Systems,* 1998, **6**(2): 266–285.

[29] Lascio L. D. & Gisolfi A., On the Algebraic Properties of Some fuzzy numbers, *The Journal of fuzzy mathematics,* 2002, **10**: 151–168.

[30] Liu Puyin, Uniformity analyses for four-layer feedforward neural networks as universal approximators, *Fuzzy System and Math.*, 2000, **14**(4): 36–43.

[31] Liu Puyin, A novel fuzzy neural network and its approximation capability, *Science in China, Series F*, 2001, **44**(3): 184–194.

[32] Liu Puyin, An efficient algorithm of polygonal fuzzy neural network, *Proc. of 4th WSEAS Int. Conf. on Soft Computing, Optimization, Stimulation and Manufacturing Systems*, April 21–23, 2004, Miami, Florida, USA.

[33] Liu Puyin, Fuzzy valued functions approximated arbitrarily close by a class of regular fuzzy neural networks, *J.Fuzzy Mathematics,* 1999, **7**(4): 826–837.

[34] Liu Puyin, On the approximation realization of fuzzy closure mapping by multilayer regular fuzzy neural network, *Multiple Valued Logic,* 2000, **5**(2): 463–480.

[35] Liu Puyin, Analyses of regular fuzzy neural networks for approximation capabilities, *Fuzzy sets and Systems,* 2000, **114**(3): 329–338

[36] Liu Puyin, Approximation analyses for fuzzy valued functions in $L_1(\mu)$ by regular fuzzy neural networks, *J. Electron. Sci.*, 2000, **17**(2): 132–138.

[37] Liu Puyin & Li Hongxing, Symmetric polygonal fuzzy numbers, *Proc. of 9th Internat. Conf. on Fuzzy Theory and Technology,* Sep. 26–30, 2003, Durham NC, USA.

[38] Liu Puyin & Li Hongxing, Symmetric polygonal fuzzy number space, *Journal of Fuzzy Math.,* (accepted for publication).

[39] Liu Puyin & Li Hongxing, Efficient learning algorithms for three-layer feedforward regular fuzzy neural networks, *IEEE Trans. on Neural Networks.*, 2004, **15**(2).

[40] Liu Puyin & Li Hongxing, Uniformity analyses for four-layer feedforward neural networks as universal approximators, *Fuzzy System and Math.*, 2003, **17**(2): 29–37.

[41] Liu Puyin & Wang Hao, Approximation capability of regular fuzzy neural networks to continuous fuzzy functions, *Science in China(Series E),* 1999, **41**(1): 20–27.

[42] Liu Puyin & Wang Huaxing, Universal approximation of fuzzy regular fuzzy neural networks to a class of continuous fuzzy valued functions, *Acta Sinca Electron.,* 1997, **25**(11): 48–52.

[43] Liu Puyin & Wu Mengda, *Fuzzy theory and applications,* Changsha: Press of National University if Defence Tech., 1998.

[44] Liu Puyin & Zhang Hanjiang, Research on fuzzy neural network theory—a survey, *Fuzzy Systems and Math.*, 1998, **10**(1): 77-87.

[45] Ma Ming, Kandel A. & Friedman M., Representing and comparing two kinds of fuzzy numbers, *IEEE Trans. on Systems, Man and Cybernet. Part C,* 1998, **28**: 573–576.

[46] Mao Zhihong, Zhang Xuefeng & Li Yeda, Research of fuzzy systems as universal function approximators, *Science in China, Series E,* 1997, **27**(4): 362–371.

[47] Nguyen N. T., A note on the extension principle for fuzzy sets, *J. Math. Anal. Appl.,* 1978, **64**: 369–380.

[48] Park S. & Han T., Iterative inversion of fuzzified neural networks, *IEEE Trans. on Fuzzy Systems,* 2000, **8**(3): 268–280.

[49] Rommelfanger H., Fuzzy linear programming and applications, *Europ. J. Operat. Research,* 1996, **92**: 512–527.

[50] Sandberg I. W. & Xu L.,Uniform approximation of multidimensional myopic maps, *IEEE Trans. on Circuits and Systems — I: Fundamental Theory and Application,* 1997, **44**(6): 477–485.

[51] Sandberg I. W. & Xu L., Uniform approximation of discrete-space multidimensional myopic maps, *Circuits Systems and Signal Processing.* 1997, **16**(3): 387–403.

[52] Scarselli F. & Tsoi A. C., Universal approximation using feedforward neural networks: a survey of some existing methods,and some new results, *Neural Networks,* 1998, **11**(1): 15–37.

[53] Steyaert H., Van Parys F., Baekeland R. & Kerre E., Implementation of piecewise linear fuzzy quantities, *Int. J. Intelligent Systems,* 1995, **10**: 1049–1059.

[54] Stinchcombe M. B., Neural network approximation of continuous functionals and continuous functions on compactifications, *Neural Networks,* 1999, **12**: 467–477.

[55] Wang Guojun, Separability, local compactness and completeness of triangular fuzzy number space, *J. Engineering Math., 1996,* **13**(3): 1–5.

[56] Wu Cong, Hui Xiaofeng & Zhu Hongwen, Fuzzy linear regression forecasting models, *Journal of Harbin Institute of Technology,* 2002, **9**(3): 280–281.

[57] Yen K. K., Roig G. & Ghoshray S., A linear regression model using triangular fuzzy number coefficients, *Proc. of Artificial Neural Networks in Engineering Conference (ANNIE'98),* St. Louis, Missouri, USA, 1998, **1**: 175–180.

[58] Yen K. K., Ghoshrya S. & Roig G., A linear regression model using triangular fuzzy number coefficients, *Fuzzy Sets and Systems,* 1999, **106**: 167–177.

[59] Yu Chianson & Li Hanlin, An algorithm for generalized fuzzy binary linear programming problems, *European Journal of Operational Research,* 2001, **133**: 496-511.

[60] Zhang X. H. Tan S. H. & Huang C. C., et al, An efficient computational algorithm for min-max operations,*Fuzzy Stes and Systems,* 1999, **104**, 297–304.

[61] Zhang X. H., Huang C. C. & Tan S. H., et al, The min-max function differentiation and training of fuzzy neural networks, *IEEE Trans. on Neural Networks,* 1996, **7**, 1139–1150.

[62] A special issue on fuzzy arithmetic, *Fuzzy Sets and Systems,* 1997, **91**(3).

CHAPTER VI

Approximation Analysis of Fuzzy Systems

Fuzzy system is an efficient model to simulate inference function of human brain [4, 13, 16–18, 60]. One most important advantage of using fuzzy systems is that linguistic fuzzy IF—THEN rules are naturally utilized in the systems. Linguistic fuzzy IF—THEN rules can be developed by human experts who are familiar with the process under consideration. As an important research topic related to fuzzy systems, universal approximation has attracted much attention in recent years (see [2, 21–23, 48, 52, 54, 55, 63, 72–79] etc). In most of fuzzy system applications, the main objective is to design a fuzzy system to approximate desired output, such as control output and ultimate decision and so on. From a mathematical point of view, such an aim is to seek a mapping from input space to output space to approximate a prescribed function with a given accuracy. That is, given $f \in C(\mathbb{R}^d)$, for arbitrary $\varepsilon > 0$, and for each compact set $U \subset \mathbb{R}^d$, there is a fuzzy system as $T: \mathbb{R}^d \longrightarrow \mathbb{R}$, so that $\|f - T\|_{\infty,U} < \varepsilon$. The achievements related to universal approximation of fuzzy systems have been applied successfully in many areas, for example, telecommunication [11], industry process control [18, 46, 58, 65], space techniques [18], system modeling [4, 13, 44], and pattern recognition [22, 43] and so on.

Up to present the main achievements related to the subject concern mainly about universal approximation to continuous function class. However, in addition to continuous systems, there exist many other types of real systems, for instance, control processes of many nonlinear optimal control models [42] and the impulse circuit, the systems related are non-continuous but integrable. Furthermore, randomness is a common phenomenon in real systems, so it is also an important problem how to deal with random systems. Thus, the systematical research for universal approximation of fuzzy systems in the noncontinuous or random environments is of much significance both in theory and application, which are the main research objectives in this chapter and the following chapter.

With respect to fuzzy control and modeling the existing fuzzy systems can be classified into two major types, namely Mamdani fuzzy systems and Takagi–Sugeno (T–S) fuzzy systems. The primary difference between them lies in their inference rule consequent. T–S fuzzy systems use linear functions of input variables as the rule consequent [33, 46–47, 58, 68, 73–75], whereas Mamdani fuzzy systems employ fuzzy sets as the consequent [31, 32, 38, 39, 77–79]. In this chapter we utilize a generalized defuzzification method to study this two types of fuzzy systems within a general framework, that is, generalized Mamdani fuzzy systems and generalized T–S fuzzy systems, which are called generalized fuzzy systems. With integral norm we shall show that this two generalized fuzzy systems can be universal approximators, respectively, that is, they can approximate each $p-$ integrable function with any degree of accuracy. Moreover, the realizing procedures of the approximations are also demonstrated.

A troublesome problem in fuzzy system application is 'rule explosion', which means the size of fuzzy rule base increases exponentially when the number of system input variables increases. An efficient method to deal with this problem is to introduce the hierarchical system configuration [49–51, 63, 64], i.e. instead of applying a fuzzy system with higher-dimensional input, a number of lower-dimensional fuzzy systems are linked in a hierarchical fashion. By such a hierarchy, the number of the fuzzy rules will increase linearly with the number of the input variables. This hierarchy is called hierarchical fuzzy system (HFS). Naturally we may put forward an important problem, that is, how can representation capability of HFS's be analyzed? Kikuchi et al in [19] show that it is impossible to give the precise expression of arbitrarily given continuous function by HFS's. So we have to analyze the approximation capability of HFS's, i.e. whether are HFS's universal approximators or not? If a function is continuously differentiable on the whole space, Wang in [63] shows an arbitrarily close approximation of the function by HFS's. He also in [64] gives sensitivity properties of HFS's and designs a suitable system structure. For each compact set U and an arbitrarily continuous function f on U, how can we find a HFS to approximate uniformly f with the arbitrary given error bound ε? In order to analyze these problems, we in the chapter also show that the I/O relationship of a HFS can be represented as one of a standard fuzzy system. Then universal approximations of HFS's are systematically studied. Comparing with the approximation methods suggested by Wang [63, 66], Buckley [2], Ying [72–76], Zeng and Singh [77, 78], we may easily find that the methods in the chapter are directly based on the I/O relationship information of the functions to be approximated, no intermediate step need, consequently they are more applicable in practice.

§6.1 Piecewise linear function

As a basic tool to study universal approximation of fuzzy systems, a piecewise linear function which is called square piecewise linear function (SPLF) is presented in this section. SPLF is a multivariate version of one-variate piecewise linear function, and it plays a role of bridge in studying universal approximation of fuzzy systems.

6.1.1 SPLF and its properties

Next let us show some useful properties of SPLF. For a given SPLF, the one-side partial derivatives exist and are bounded. Then we use SPLF's to build the approximate representations of continuous and integrable functions under the sense of '$\approx_\varepsilon$' according to the maximum norm and integral norm, respectively.

Definition 6.1 Let $S : \mathbb{R}^d \longrightarrow \mathbb{R}$. We call S to be a SPLF, if the following conditions hold:

(i) S is a continuous function;

(ii) There is $a > 0$, so that S is identical zero outside the following cube

$$\Delta = \{(x_1, ..., x_d) \in \mathbb{R}^d \big| -a \leq x_i \leq a,\ i = 1, ..., d\},$$

so S has a compact support;

(iii) There exist $N_S \in \mathbb{N}$, and d dimensional polyhedrons $\Delta_1, ..., \Delta_{N_S}$, so that S is a linear function on each Δ_j, moreover for $j = 1, ..., N_S$, we have

$$(x_1, ..., x_d) \in \Delta_j, \Longrightarrow S(x_1, ..., x_d) = \sum_{i=1}^{d} \lambda_{ij} \cdot x_i + \gamma_j,$$
$$\Delta_1 \cup \cdots \cup \Delta_{N_S} = \Delta,$$

where, λ_{ij}, γ_j are constants. Below $\Delta_1, ..., \Delta_{N_S}$ are called the polyhedrons corresponding to S.

By $\mathcal{D}_d$ we denote the collection of all SPLF's, and $\mathcal{D}_d^0$ the set of all SPLF's in $\mathcal{D}_d$, whose supports are included in $[-1, 1]^d$. For $S \in \mathcal{D}_d$, denote $V(\Delta_j)$ as the vertex set of Δ_j, and $V(S)$ the set of all vertices of $\Delta_1, ..., \Delta_{N_S}$, i.e. $V(S) = \bigcup_{j=1}^{N_S} V(\Delta_j)$. By Definition 6.1 easily we have, if $S \in \mathcal{D}_d$, then $\forall (x_1, ..., x_d) \in \mathbb{R}^d$, $\forall i \in \{1, ..., d\}$, both left derivative $\partial S_-(x_1, ..., x_d)/\partial x_i$ and right derivative $\partial S_+(x_1, ..., x_d)/\partial x_i$ exist. Moreover, $\forall \mathbf{x}^0 = (x_1^0, ..., x_d^0) \in \mathbb{R}^d$, it follows that

$$\left|\frac{\partial S_+(\mathbf{x}^0)}{\partial x_i}\right| \vee \left|\frac{\partial S_-(\mathbf{x}^0)}{\partial x_i}\right| \leq \bigvee_{(x_1,...,x_d) \in V(S)} \left\{\left|\frac{\partial S_+(x_1, ..., x_d)}{\partial x_i}\right| \vee \left|\frac{\partial S_-(x_1, ..., x_d)}{\partial x_i}\right|\right\}. \tag{6.1}$$

For arbitrary $i \in \{1, ..., d\}$, write

$$D_i(S) = \bigvee_{(x_1,...,x_d) \in V(S)} \left\{\left|\frac{\partial S_+(x_1, ..., x_d)}{\partial x_i}\right| \vee \left|\frac{\partial S_-(x_1, ..., x_d)}{\partial x_i}\right|\right\}. \tag{6.2}$$

Then $\forall h \in \mathbb{R}$, $i \in \{1, ..., d\}$, it is easy to show

$$\big|S(x_1, ..., x_{i-1}, x_i + h, x_{i+1}, ..., x_d) - S(x_1, ..., x_d)\big| \leq |h| \cdot D_i(S). \tag{6.3}$$

Lemma 6.1 *Suppose $S : \mathbb{R}^d \longrightarrow \mathbb{R}$ is a SPLF, and $h_1, ..., h_d \in \mathbb{R}$ be given d constants. Then*

$$\forall (x_1, ..., x_d) \in \mathbb{R}^d,\ \big|S(x_1 + h_1, ..., x_d + h_d) - S(x_1, ..., x_d)\big| \leq \sum_{i=1}^{d} D_i(S) \cdot |h_i|.$$

Proof. At first it is easy to show the following inequalities:

$$\begin{aligned}
&\big|S(x_1+h_1,...,x_d+h_d)-S(x_1,...,x_d)\big| \\
&\qquad=\big|S(x_1+h_1,...,x_d+h_d)-S(x_1,x_2+h_2,...,x_d+h_d) \\
&\qquad\quad+S(x_1,x_2+h_2,...,x_d+h_d)-S(x_1,x_2,x_3+h_3,....,x_d+h_d) \\
&\qquad\quad+\cdots+S(x_1,...,x_{d-1},x_d+h_d)-S(x_1,...,x_d)\big| \\
&\qquad\leq\big|S(x_1+h_1,...,x_d+h_d)-S(x_1,x_2+h_2,...,x_d+h_d)\big| \\
&\qquad\quad+\big|S(x_1,x_2+h_2...,x_d+h_d)+S(x_1,x_2,x_3+h_3,...,x_d+h_d)\big| \\
&\qquad\quad+\cdots+\big|S(x_1,...,x_{d-1},x_d+h_d)-S(x_1,...,x_d)\big|.
\end{aligned}$$

Using (6.3) we can get the following fact:

$$\big|S(x_1+h_1,...,x_d+h_d)-S(x_1,...,x_d)\big|\leq\sum_{i=1}^{d}D_i(S)\cdot|h_i|.$$

The lemma is proved. □

6.1.2 Approximation of SPLF's

Next let us present the approximation representations of functions in $L_p(\mu)$ or $C(U)$ by SPLF's with arbitrary accuracy $\varepsilon>0$, where $U\subset\mathbb{R}^d$ is a compact set.

Theorem 6.1 *Let μ be Lebesgue measure on $\mathbb{R}^d$. Then $\mathcal{D}_d$ is dense in $L_p(\mu)$ with $L_p(\mu)-$norm, that is, for any $\varepsilon>0$, and $f\in L_p(\mu)$, there is $S\in\mathcal{D}_d$, so that $\|f-S\|_{\mu,p}<\varepsilon$.*

Proof. For simplicity we show the conclusion in the two dimensional space $\mathbb{R}^2$, i.e. $d=2$. For the cases of $d>2$ or $d=1$, the proofs are similar. Since $f\in L_p(\mu)$ we get, $\int_{\mathbb{R}^2}|f(\mathbf{x})|^p\mathrm{d}\mu<+\infty$, that is

$$\int_{\mathbb{R}^2}|f(\mathbf{x})|^p\mathrm{d}\mu=\sum_{m=0}^{+\infty}\int_{m\leq\|\mathbf{x}\|<m+1}|f(\mathbf{x})|^p\mathrm{d}\mu<+\infty.$$

Hence for $\varepsilon>0$, there is $m_0\in\mathbb{N}$, satisfying

$$\sum_{m=m_0-1}^{+\infty}\int_{m\leq\|\mathbf{x}\|<m+1}|f(\mathbf{x})|^p\mathrm{d}\mu<\frac{\varepsilon^p}{2},\Longrightarrow\int_{\|\mathbf{x}\|\geq m_0-1}|f(\mathbf{x})|^p\mathrm{d}\mu<\frac{\varepsilon^p}{2}.$$

Let $a>0$, so that $\Delta\triangleq\big\{\mathbf{x}\in\mathbb{R}^2\big|-a\leq x_1,\,x_2\leq a\big\}\supset\big\{\mathbf{x}\in\mathbb{R}^2\big|\ \|\mathbf{x}\|\leq m_0\big\}$. Therefore

$$\int_{\Delta^c}|f(\mathbf{x})|^p\mathrm{d}\mu\leq\int_{\|\mathbf{x}\|>m_0}|f(\mathbf{x})|^p\mathrm{d}\mu<\frac{\varepsilon^p}{2}.$$

Since $\int_{\Delta} |f(\mathbf{x})|^p \mathrm{d}\mu < +\infty$, in $\|\cdot\|_{\Delta,p}$−norm the function f can be approximated by continuous functions on Δ to any degree of accuracy [53]. Hence we can assume that f is Riemann integrable on Δ, and $f(\mathbf{x}) = 0$ when $\mathbf{x} \in \partial\Delta$, the boundary of Δ, i.e.

$$\partial\Delta = \big\{(x_1, x_2) \in \Delta \big|\ |x_1| = a,\ |x_2| \le a\big\} \cup \big\{(x_1, x_2) \in \Delta \big|\ |x_1| \le a,\ |x_2| = a\big\}.$$

Thus, there is $m \in \mathbb{N}$, if partition each side of Δ into $2m$ identical length parts, and $4m^2$ squares as $W_1, ..., W_{4m^2}$ are obtained. Let

$$\overline{\delta}_j = \bigvee_{\mathbf{x} \in W_j} \{f(\mathbf{x})\},\ \ \underline{\delta}_j = \bigwedge_{\mathbf{x} \in W_j} \{f(\mathbf{x})\}\ (j = 1, ..., 4m^2),$$

Then $\sum_{j=1}^{4m^2} \int_{W_j} |\overline{\delta}_j - \underline{\delta}_j|^p \mathrm{d}\mu < \varepsilon^p/4$. Moreover,

$$\sum_{j \in B(\Delta)} \int_{W_j} (|\overline{\delta}_j| \vee |\underline{\delta}_j|)^p \mathrm{d}\mu < \frac{\varepsilon^p}{8 \times 2^p},$$

where $B(\Delta)$ is the index set of $8m-4$ next-door neighbor squares of the boundary $\partial\Delta$. Respectively linking a pair of oppose vertices of W_j $(j = 1, ..., 4m^2)$, we obtain $8m^2$ equicrural rectangular triangles $\Delta_1, ..., \Delta_{8m^2}$. For each $j = 1, ..., 8m^2$, Using the function values of f at three vertices of W_j we can establish a spatial triangle (if the corresponding three spatial points are collinear a line segment is established). By these piecewise triangles we can define a SPLF S, and obviously $S \in \mathcal{D}_2$. Denote $B^c(\Delta) \triangleq \{1, ..., 4m^2\} \setminus B(\Delta)$. Considering the following facts:

$$\forall j \in B^c(\Delta),\ \int_{W_j} |f(\mathbf{x}) - S(\mathbf{x})|^p \mathrm{d}\mu \le \int_{W_j} |\overline{\delta}_j - \underline{\delta}_j|^p \mathrm{d}\mu;$$

$$\forall j \in B(\Delta),\ \int_{W_j} |S(\mathbf{x})|^p \mathrm{d}\mu \le \int_{W_j} \big(|\overline{\delta}_j| \vee |\underline{\delta}_j|\big)^p \mathrm{d}\mu,$$

and the inequality: $\big(|a| + |b|\big)^p \le 2^p\big(|a|^p + |b|^p\big)$, we can show

$$\begin{aligned}
&\int_{\Delta} |f(\mathbf{x}) - S(\mathbf{x})|^p \mathrm{d}\mu \\
&\quad = \sum_{j \in B^c(\Delta)} \int_{W_j} |f(\mathbf{x}) - S(\mathbf{x})|^p \mathrm{d}\mu + \sum_{j \in B(\Delta)} \int_{W_j} |f(\mathbf{x}) - S(\mathbf{x})|^p \mathrm{d}\mu \\
&\quad \le \sum_{j \in B^c(\Delta)} \int_{W_j} |\overline{\delta}_j - \underline{\delta}_j|^p \mathrm{d}\mu + 2^{p+1} \sum_{j \in B(\Delta)} \int_{W_j} \big(|\overline{\delta}_j| \vee |\underline{\delta}_j|\big)^p \mathrm{d}\mu \\
&\quad < \frac{\varepsilon^p}{4} + 2^{p+1} \cdot \frac{1}{8} \cdot \Big(\frac{\varepsilon}{2}\Big)^p = \frac{\varepsilon^p}{2}.
\end{aligned}$$

So we can conclude that the following equality holds:

$$\|f-S\|_{\mu,p}=\Big(\int_{\Delta}|f(\mathbf{x})-S(\mathbf{x})|^p\mathrm{d}\mu+\int_{\Delta^c}|f(\mathbf{x})|^p\mathrm{d}\mu\Big)^{\frac{1}{p}}<\Big(\frac{\varepsilon^p}{2}+\frac{\varepsilon^p}{2}\Big)^{\frac{1}{p}}=\varepsilon.$$

□

Remark 6.1 In Theorem 6.1, since the SPLF S is identical to aero outside a cube as $\Delta=\{(x_1,...,x_d)|-a\le x_i\le a,\ i=1,...,d\}$, by a suitable linear transform we can change S into $S_1\in\mathcal{D}_d^0$, so that S_1 is identical to zero outside $[-1,1]^d$. Correspondingly f is changed into f_1. If let $A=[-1,\ 1]^d$, then we have, $\forall\varepsilon>0$, there is $S\in\mathcal{D}_d$, $\|f-S\|_{\mu,p}<\varepsilon\iff\forall\varepsilon>0$, there exists $S_1\in\mathcal{D}_d^0$, $\|f_1-S_1\|_{A,p}<\varepsilon$.

If S is the SPLF obtained by the proof of Theorem 6.1, we can use f to calculate the supremum $D_i(S)$ $(i=1,...,d)$ defined by (6.2), i.e.

Corollary 6.1 *Let $f:\mathbb{R}^d\longrightarrow\mathbb{R}$ be a Riemann integrable function, and $S\in\mathcal{D}_d$ be a SPLF obtained in Theorem 6.1. Then there exist $a>0$, and $h>0$, so that for $i\in\{1,...,d\}$, if let $\mathbf{x}^{h^i}=(x_1,...,x_{i-1},x_i+h,x_{i+1},...,x_d)$, and*

$$D_i(f)=\bigvee_{x_1,...,x_d,x_i+h\in[-a,a]}\{h^{-1}\big|f(\mathbf{x}^{h^i})-f(x_1,...,x_d)\big|\}\qquad(6.4)$$

we have, $\forall i\in\{1,...,d\}$, $D_i(S)=D_i(f)$.

Proof. Similarly with Theorem 6.1, it suffices to prove the conclusion when $d=2$. Let a be a positive number obtained in the proof of Theorem 6.1, and $h>0$ be the identical side length of the squares $W_1,...,W_{4m^2}$. Moreover, let the three vertices of the rectangular triangle Δ_j $(j=1,...,8m^2)$ be (x_1^1,x_2^1), (x_1^2,x_2^2), (x_1^3,x_2^3), respectively. Then the corresponding function values are as following: $f(x_1^1,x_2^1)$, $f(x_1^2,x_2^2)$, $f(x_1^3,x_2^3)$. By three points on the surface $z=f(x_1,x_2)$ as $\big(x_1^1,\ x_2^1,\ f(x_1^1,x_2^1)\big)$, $\big(x_1^2,\ x_2^2,\ f(x_1^2,x_2^2)\big)$, $\big(x_1^3,\ x_2^3,\ f(x_1^3,x_2^3)\big)$, we obtain a spatial triangle whose algebraic equation $z=S(x_1,x_2)$ is

$$\begin{vmatrix} x_1 & x_2 & S(x_1,x_2) & 1\\ x_1^1 & x_2^1 & f(x_1^1,x_2^1) & 1\\ x_1^2 & x_2^2 & f(x_1^2,x_2^2) & 1\\ x_1^3 & x_2^3 & f(x_1^3,x_2^3) & 1\end{vmatrix}=0.\qquad(6.5)$$

On Δ_j we can calculate the partial derivatives of S, $\partial S/\partial x_1$, $\partial S/\partial x_2$ respectively as follows:

$$\begin{cases}\dfrac{\partial S(x_1,x_2)}{\partial x_1}=\dfrac{(x_2^2-x_2^3)f(x_1^1,x_2^1)+(x_2^1-x_2^2)f(x_1^3,x_2^3)+(x_2^3-x_2^1)f(x_1^2,x_2^2)}{(x_2^3-x_2^1)x_1^2+(x_2^1-x_2^2)x_1^3+(x_2^2-x_2^3)x_1^1},\\[2ex] \dfrac{\partial S(x_1,x_2)}{\partial x_2}=\dfrac{(x_1^3-x_1^2)f(x_1^1,x_2^1)+(x_1^2-x_1^1)f(x_1^3,x_2^3)+(x_1^1-x_1^3)f(x_1^2,x_2^2)}{(x_2^3-x_2^1)x_1^2+(x_2^1-x_2^2)x_1^3+(x_2^2-x_2^3)x_1^1}.\end{cases}\qquad(6.6)$$

Since the three vertices of Δ_j lie in horizontal line and vertical line, respectively, and one is the crossover point of the horizontal and vertical lines, we can set $x_1^1 = x_1^2$, $x_2^1 = x_2^3$. Using the fact $|x_2^1 - x_2^2| = |x_1^1 - x_1^3| = |x_2^2 - x_2^3| = h$, and (6.6), we get

$$\Big|\frac{\partial S(x_1,x_2)}{\partial x_1}\Big| = \frac{|f(x_1^1+h, x_2^1) - f(x_1^1, x_2^1)|}{h},$$
$$\Big|\frac{\partial S(x_1,x_2)}{\partial x_2}\Big| = \frac{|f(x_1^1, x_2^1+h) - f(x_1^1, x_2^1)|}{h}.$$

So by (6.2) it follows that (6.4) holds, which proves the lemma. □

Corollary 6.2 *Let μ be Lebesgue measure on $\mathbb{R}_+^2$, and $g : \mathbb{R}_+^2 \longrightarrow \mathbb{R}$ satisfy: $\int_{\mathbb{R}_+^2} |g(t,s)|^2 \mathrm{d}\mu < +\infty$. Then for arbitrary $\varepsilon > 0$, there exist $a > 0$, and a SPLF S so that*

$$Supp(S) \subset \big\{(t,s) \in \mathbb{R}_+^2 \,\big|\, 0 \le t,\ s \le a\big\} \triangleq [0,\ a]^2.$$

Moreover, the following facts hold:

(i) $\big\{\int_{\mathbb{R}_+^2} |g(t,s) - S(t,s)|^2 \mathrm{d}\mu\big\}^{\frac{1}{2}} < \varepsilon$;

(ii) If let

$$D_1(g) \triangleq \bigvee_{t_1, t_1+h, t_2 \in [0,a]} \Big\{\Big|\frac{g(t_1+h, t_2) - g(t_1,t_2)}{h}\Big|\Big\},$$
$$D_2(g) \triangleq \bigvee_{t_1, t_2, t_2+h \in [0,a]} \Big\{\Big|\frac{g(t_1, t_2+h) - g(t_1,t_2)}{h}\Big|\Big\},$$

then for sufficiently small $h > 0$, it follows that $D_i(S) = D_i(g)$ $(i = 1, 2)$.

The proof of Corollary 6.2 is similar with ones of Theorem 6.1 and Corollary 6.1, for Corollary 6.2 is different from Theorem 6.1, or Corollary 6.1 only in the domains, the former being $\mathbb{R}_+^2$ whereas the latter being $\mathbb{R}^d$. Corollary 6.2 will paly an important role in studying the approximating capability of fuzzy systems in random environment in chapter VII.

Theorem 6.2 *Let $f : \mathbb{R}^d \longrightarrow \mathbb{R}$ be a continuous function. Suppose $U \subset \mathbb{R}^d$ is an arbitrary compact set. Then there is $a > 0$, so that $U \subset [-a,\ a]^d$, moreover, for any $\varepsilon > 0$, there exists $S \in \mathcal{D}_d$, satisfying*

$$\mathrm{Supp}(S) \subset [-a,\ a]^d, \quad \|f - S\|_{\infty, U} < \varepsilon.$$

Proof. As doing in Theorem 6.1, we complete the proof in $\mathbb{R}^2$. Since $U \subset \mathbb{R}^2$ is a compact set, there is $a > 0$, satisfying $U \subset [-a,\ a]^2$. For simplicity we may let $[-a,\ a]^2 = [-1,\ 1]^2$. The continuity of f implies that f is uniformly continuous on $[-1,\ 1]^2$. For arbitrary $\varepsilon > 0$, there exists $m_0 \in \mathbb{N}$, such that for any $m > m_0$, if partition $[-1,\ 1]$ into $2m$ subintervals with identical length, $\big[-1,\ (1-m)/m\big], ..., \big[0,\ 1/m\big], ..., \big[(m-1)/m,\ 1\big]$, then $4m^2$ small squares D_{ij}^m's defined as following are obtained, where $i,\ j = -m+1,\ -m+2, ..., m-1,\ m$.

$$D_{ij}^m = \Big\{(x,y) \in \big[-1,\ 1\big]^2 \Big| \frac{i-1}{m} \le x \le \frac{i}{m},\ \frac{j-1}{m} \le y \le \frac{j}{m}\Big\},$$

then $\forall (x_1,y_1),\ (x_2,y_2) \in D_{ij}^m,\ \big|f(x_1,y_1) - f(x_2,y_2)\big| < \varepsilon/5$. Therefore

$$\overline{\delta}_{ij} \triangleq \bigvee_{(x,y)\in D_{ij}^m} \{f(x,y)\},\ \ \underline{\delta}_{ij} \triangleq \bigwedge_{(x,y)\in D_{ij}^m} \{f(x,y)\} \Longrightarrow \big|\overline{\delta}_{ij} - \underline{\delta}_{ij}\big| < \frac{\varepsilon}{4},$$

where $i,j = -m+1,\ -m+2, ..., m-1,\ m$. Similarly with Theorem 6.1, through the small square D_{ij}^m $(i,j = -m+1,\ -m+2, ..., m-1,\ m)$ we obtain $8m^2$ equicrural rectangular triangles $\Delta_1, ..., \Delta_{8m^2}$. For each $k \in \{1, ..., 8m^2\}$, by the spatial points $\big(x_1^k, y_1^k,\ f(x_1^k,y_1^k)\big),\ \big(x_2^k, y_2^k,\ f(x_2^k,y_2^k)\big),\ \big(x_3^k, y_3^k,\ f(x_3^k,y_3^k)\big)$ we can determine a plane equation $z = S(x,y)$ as (6.5). Let h be side length of D_{ij}^k, i.e. $h = 1/m$. It is no harm to assume $y_1^k = y_2^k,\ x_1^k = x_3^k,\ x_2^k = x_1^k + h,\ y_3^k = y_1^k + h$. Rewriting (6.5) we obtain

$$\begin{aligned} S(x,y) = f(x_1^k,y_1^k) + \frac{1}{h}\big[&(x-x_1^k)\big(f(x_2^k,y_2^k) - f(x_1^k,y_1^k)\big) \\ &+(y-y_1^k)\big(f(x_3^k,y_3^k) - f(x_1^k,y_1^k)\big)\big]. \end{aligned} \tag{6.7}$$

Given arbitrarily $(x,y) \in U$, there is $k \in \{1, ..., 8m^2\}$, so that $(x,y) \in \Delta_k$. By (6.7) easily we can show

$$\begin{aligned} \big|f(x,y) - S(x,y)\big| \le &\ \big|f(x,y) - f(x_1^k,y_1^k)\big| + \\ &+\frac{|x-x_1^k|}{h}\big|f(x_2^k,y_2^k) - f(x_1^k,y_1^k)\big| + \frac{|y-y_1^k|}{h}\big|f(x_3^k,y_3^k) - f(x_1^k,y_1^k)\big|. \end{aligned}$$

Using the fact $|x - x_1^k|/h \le 1,\ |y - y_1^k|/h \le 1$, and (6.7) we get

$$\big|f(x,y) - S(x,y)\big| \le \frac{\varepsilon}{4} + \frac{\varepsilon}{4} + \frac{\varepsilon}{4} = \frac{3\varepsilon}{4}.$$

Thus, $\big\|f - S\big\|_{\infty,U} \le 3\varepsilon/4 < \varepsilon$. □

Using Theorem 6.1 and Theorem 6.2 easily we can show the following conclusion.

Remark 6.2 For a given function f, we can obtain a SPLF S by partitioning $[-a,\ a]$ into $2m$ identical length parts, that is, partitioning the cube $D = [-a,\ a]^d$ into $(2m)^d$ small cubes, moreover

$$f\Big(\frac{ap_1}{m}, ..., \frac{ap_d}{m}\Big) = S\Big(\frac{ap_1}{m}, ..., \frac{ap_2}{m}\Big)\ (p_1, ..., p_d = -m, -m+1, ..., m-1, m).$$

By Theorem 6.1 and Theorem 6.2, if $f \in L_p(\mu)$, or $U \subset \mathbb{R}^d$ is a compact set, and $g \in C(U)$, then for arbitrary $\varepsilon >, 0$ there exist $S_1, S_2 \in \mathcal{D}_d$ satisfying respectively: $\|f - S_1\|_{L_p(\mu)} < \varepsilon$; $\|g - S_2\|_{U,\infty} < \varepsilon$. By such a fact, universal approximations of fuzzy systems to continuous function class or, integrable function class are also the approximating problems in $\mathcal{D}_d$ by fuzzy systems with respective metric senses. From our discussions in the section, SPLF's possess many useful properties, such as they are zero outside a compact set of $\mathbb{R}^d$; Their one-sided derivatives exist and are bounded; They are uniformly continuous on $\mathbb{R}^d$ and so on, which provide us with much convenience for studying universal approximation of generalized fuzzy systems.

§6.2 Approximation of generalized fuzzy systems with integral norm

By partitioning the input space of fuzzy systems we define the corresponding SPLF $S \in \mathcal{D}_d$, and show that generalized fuzzy systems can be universal approximators with $L_p(\mu)$−norm. These constitute the main research topics in the section. The proofs of the approximation theorems for generalized fuzzy systems are constructive, and so the corresponding approximating procedure can be convenient to realize.

Give an adjustable parameter $a > 0$. For $m \in \mathbb{N}$, partition $[-a, a]$ into $m_1 + m_2$ parts: $-a = a_{-m_1} < a_{-m_1+1} < \cdots < a_{-1} < 0 < a_1 < \cdots < a_{m_2-1} < a_{m_2} = a$. Then define the fuzzy set $\tilde{A}_{ij} \in \mathcal{F}(\mathbb{R})$ $(i = 1, ..., d;\ j = -m_1, -m_1+1, ..., m_2-1, m_2)$. For convenience of application, in the following we array $\tilde{A}_{ij}$ $(i = 1, ..., d;\ j = -m_1, -m_1+1, ..., m_2-1, m_2)$ with certain order, that is, this class of fuzzy sets can be defined as the following definition.

Definition 6.2 Give adjustable parameters $a > 0$, and $m \in \mathbb{N}$. The class of fuzzy sets $\{\tilde{A}_{ij}\ | i \in \{1, ..., d\}, j \in \{-m_1, -m_1+1, ..., m_2-1, m_2\}\}$ is called to satisfy S–L condition, if the following facts hold:

(i) $\tilde{A}_{ij}(\cdot)$ is Riemann integrable on $\mathbb{R}$;

(ii) Each $\tilde{A}_{ij}$ is a fuzzy number, and the kernel $\mathrm{Ker}(\tilde{A}_{ij})$ includes $\{a_j\}$, the support $\mathrm{Supp}(\tilde{A}_{ij}) \subset [-a, a]$, moreover for arbitrarily given $k, k_1, k_2 \in \{-m_1, -m_1+1, ..., m_2-1, m_2\}$, we have

$$k_1 < k < k_2,\ \tilde{A}_{ik_1}(x) > 0,\ \tilde{A}_{ik_2}(x) > 0, \Longrightarrow \tilde{A}_{ik}(x) > 0;$$

(iii) There is a constant $c_0 \in \mathbb{N}$, independent of a, m so that $\forall x \in [-a, a]$,

$$\forall i \in \{1, ..., d\}, 1 \le \mathrm{Card}(\{j |\ \tilde{A}_{ij}(x) > 0\}) \le c_0. \tag{6.8}$$

In the following we denote the fuzzy set family with S–L condition of Definition 6.2 by $\widetilde{\mathcal{O}}(a,\ m_1+m_2)$, i.e.

$$\widetilde{\mathcal{O}}(a,\ m_1+m_2) = \big\{\widetilde{A}_{ij}\ \big|i\in\{1,...,d\},\ j\in\{-m_1,\ -m_1+1,...,m_2-1,\ m_2\}\big\}.$$

If $m_1=m_2=m$, then write $\widetilde{\mathcal{O}}(a,\ m_1+m_2)\triangleq\widetilde{\mathcal{O}}(a,\ m)$. And let $\xi(a,\ m)$ be the maximal length of the partition intervals corresponding to $\widetilde{\mathcal{O}}(a,\ m)$, that is, $\xi(a,\ m)=\bigvee\limits_{-m\le i\le m-1}\big\{|a_{i+1}-a_i|\big\}$. If the fuzzy set family $\{\widetilde{A}_{ij}\ |i=1,...,d;\ j=-m,\ -m+1,....,m-1,\ m\}$ with S–L condition as obtained by dividing the interval $[-a,\ a]$, equally, i.e. $a_j=aj/m$ $(j=-m,\ -m+1,...,m-1,\ m)$, then we denote the fuzzy set family by $\widetilde{\mathcal{O}}_0(a,\ m)$. Figure 6.1 demonstrates the membership curves of fuzzy numbers in the fuzzy set family $\widetilde{\mathcal{O}}_0(a,\ m)$, by which we know $c_0=2$. If $\widetilde{A}_{ij}\in\widetilde{\mathcal{O}}_0(a,\ m)$, for given i, we can describe $\widetilde{A}_{ij}$ with some natural language senses according to the value of j, such as 'positive large' 'positive small' 'negative small' 'negative large', and so on. When designing fuzzy inference rules, we take $\widetilde{\mathcal{O}}_0(a,\ m)$ as the antecedent fuzzy set family of fuzzy rules.

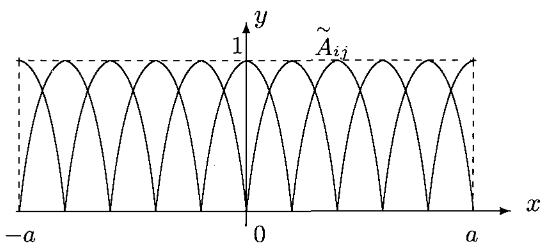


Figure 6.1 Membership curves of $\widetilde{A}_{ij}$'s

Let T be a continuous t−norm, satisfying $\alpha_1>0,\ \alpha_2>0,\Longrightarrow\alpha_1T\alpha_2>0$. We introduce the following notations:

$$\alpha_1T\alpha_2=T(\alpha_1,\ \alpha_2),\ \alpha_1T\alpha_2T\alpha_3=(\alpha_1T\alpha_2)T\alpha_3,...,\ (\alpha_1,\ \alpha_2,\ \alpha_3\in[0,1]).$$

For $(p_1,...,p_d)\in\{-m,\ -m+1,...,m-1,\ m\}^d$, and $\widetilde{A}_{1p_1},...,\widetilde{A}_{dp_d}\in\widetilde{\mathcal{O}}(a,\ m)$, let

$$H_{p_1...p_d}(x_1,...,x_d)=\widetilde{A}_{1p_1}(x_1)\ T\ \widetilde{A}_{2p_2}(x_2)\ T\cdots T\ \widetilde{A}_{dp_d}(x_d).$$

By Definition 6.2 it follows that $\forall(x_1,...,x_d)\in[-a,\ a]^d$, there is $(p_1,...,p_d)\in\{-m,\ -m+1,...,m-1,\ m\}^d$, so that $H_{p_1...p_d}(x_1,...,x_d)>0$. Give $(x_1,...,x_d)\in[-a,\ a]^d$, denote

$$N(x_1,...,x_d)=\big\{(p_1,...,p_d)\in\{-m,\ -m+1,...,m-1,\ m\}^d\big|\widetilde{A}_{1p_1},...,\widetilde{A}_{dp_d}\in\widetilde{\mathcal{O}}(a,\ m),\ \widetilde{A}_{1p_1}(x_1)\ T\cdots T\ \widetilde{A}_{dp_d}(x_d)>0\big\}.$$

Lemma 6.2 *Let* $(x_1, ..., x_d) \in \mathbb{R}^d$. *Then the following facts hold:*

(i) If $\{\widetilde{A}_{ij} \mid i = 1, ..., d;\ j = -m, -m+1, ..., m-1, m\} = \widetilde{\mathcal{O}}(a, m)$, *we have*

$$(p_1, ..., p_d) \in N(x_1, ..., x_d), \Longrightarrow \forall i \in \{1, ..., d\},\ a_{p_i - c_0} \leq x_i \leq a_{p_i + c_0};$$

(ii) If $\{\widetilde{A}_{ij} \mid i = 1, ..., d;\ j = -m, -m+1, ..., m-1, m\} = \widetilde{\mathcal{O}}_0(a, m)$, *it follows that*

$$\forall (p_1, ..., p_d) \in N(x_1, ..., x_d),\ \forall i \in \{1, ..., d\},\ \frac{a(p_i - c_0)}{m} \leq x_i \leq \frac{a(p_i + c_0)}{m}.$$

Proof. It suffices to show (i), for the proof of (ii) is similar. For $(p_1, ..., p_d) \in N(x_1, ..., x_d)$, we have, $N(x_1, ..., x_d) \neq \emptyset$, and $(x_1, ..., x_d) \in [-a, a]^d$. Considering $\widetilde{A}_{1p_1}(x_1)\ T \cdots T\ \widetilde{A}_{dp_d}(x_d) > 0$, we get, $\forall i = 1, ..., d,\ \widetilde{A}_{ip_i}(x_i) > 0$. For each $i \in \{1, ..., d\}$, since $\widetilde{A}_{ip_i}$ is a fuzzy number, and $\widetilde{A}_{ip_i}(a_{p_i}) = 1$, we can show, $\widetilde{A}_{ip_i}(a_{p_i + c_0}) \geq \widetilde{A}_{ip_i}(x_i) > 0$, moreover $a_{p_i + c_0} \in [-a, a]$ by the properties of fuzzy numbers [35] and the fact: $x_i > a_{p_i + c_0}, \Longrightarrow x_i > a_{p_i}$. Since $\widetilde{A}_{i(p_i + c_0)}(a_{p_i + c_0}) = 1 > 0$, by the definition of $\widetilde{A}_{ij}$ it follows that $\forall j:\ p_i \leq j \leq p_i + c_0,\ \widetilde{A}_{ij}(a_{p_i + c_0}) > 0$. Therefore

$$\text{Card}\left(\left\{j \middle| \widetilde{A}_{ij}(a_{p_i + c_0}) > 0\right\}\right) \geq c_0 + 1,$$

which contradicts (6.8) of Definition 6.2. Thus, $x_i \leq a_{p_i + c_0}$. Similarly we can show, $x_i \geq a_{p_i - c_0}$. So $a_{p_i - c_0} \leq x_i \leq a_{p_i + c_0}$ $(i = 1, ..., d)$. The lemma is proved. □

By Lemma 6.2, if assume $\lim_{m \to +\infty} \xi(a, m) = 0$, then for any $(x_1, ..., x_d) \in \mathbb{R}^d$, and $\forall (p_1, ...p_d) \in N(x_1, ..., x_d)$, it follows that $\lim_{m \to +\infty} a_{p_i} = x_i$ $(1 \leq i \leq d)$.

6.2.1 Generalized Mamdani fuzzy system

The fuzzy rule base of a Mamdani fuzzy system consists of Mamdani type fuzzy inference rules. For given $\widetilde{A}_{1p_1}, ..., \widetilde{A}_{dp_d} \in \widetilde{\mathcal{O}}_0(a, m)$, we can express the corresponding Mamdani fuzzy rule as follows [32, 72, 76–78]:

$$R_{p_1 \cdots p_d}: \text{IF } x_1 \text{ is } \widetilde{A}_{1p_1} \text{ and } \cdots \text{ and } x_d \text{ is } \widetilde{A}_{dp_d} \text{ THEN } u \text{ is } \widetilde{U}_{r(p_1, \ldots, p_d)},$$

where $x_1, ..., x_d$ are input variables, $\widetilde{A}_{1p_1}, ..., \widetilde{A}_{dp_d}$ are antecedent fuzzy sets, and u is output variable whose range is the interval $U = [-b, b]$ $(b > 0)$, and $\widetilde{U}_{r(p_1, \ldots, p_d)}$ is consequent fuzzy set, r is an adjustable real function, and $r\big|_{\mathbb{Z}}: \mathbb{Z} \longrightarrow \mathbb{Z}$. b is also an adjustable parameter.

Using above Mamdani inference rule $R_{p_1 \cdots p_d}$ we can define a fuzzy implication relation $\widetilde{R}_{p_1 \cdots p_d}$ as

$$\widetilde{R}_{p_1 \cdots p_d} = \widetilde{A}_{1p_1} \times \cdots \times \widetilde{A}_{dp_d} \longrightarrow \widetilde{U}_{r(p_1, \ldots, p_d)},$$

The implication relation is a fuzzy set on $[-a,\ a]^d \times [-b,\ b]$, that is, $\widetilde{R}_{p_1 \dots p_d} \in \mathcal{F}\big([-a,\ a]^d \times [-b,\ b]\big)$. And its membership function is

$$\widetilde{R}_{p_1 \dots p_d}\ (x_1, ..., x_d,\ u) = \widetilde{A}_{1p_1}(x_1)\ T \cdots T\ \widetilde{A}_{dp_d}(x_d)\ T\ \widetilde{U}_{r(p_1, \dots, p_d)}(u).$$

For any $\widetilde{A} \in \mathcal{F}\big([-a,\ a]^d\big)$, using the fuzzy relation $\widetilde{R}_{p_1 \dots p_d}$ and the $\vee - T$ composition '$\circ$', we can establish a fuzzy set as $\widetilde{A} \circ \widetilde{R}_{p_1 \dots p_d}$ on $[-b,\ b]$:

$$\begin{aligned}\big(\widetilde{A} \circ \widetilde{R}_{p_1 \dots p_d}\big)(u) &= \bigvee_{(x_1', \dots, x_d') \in [-a, a]^d} \big\{\widetilde{A}(x_1', ..., x_d')\ T\ \widetilde{R}_{p_1 \dots p_d}(x_1', ..., x_d', u)\big\} \\ &= \bigvee_{(x_1', \dots, x_d') \in [-a, a]^d} \big\{\widetilde{A}(x_1', ..., x_d')\, T\ \widetilde{A}_{kp_1}(x_1')\, T \cdots T\ \widetilde{A}_{dp_d}(x_d')\, T\ \widetilde{U}_{r(p_1, \dots, p_d)}(u)\big\}.\end{aligned} \tag{6.9}$$

In (6.9) choose $\widetilde{A}$ as a singleton fuzzification at $(x_1, ..., x_n)$, i.e.

$$\widetilde{A}(x_1, ..., x_d) = 1, \ \ \widetilde{A}(x_1', ..., x_d') = 0\ \big((x_1', ..., x_d') \neq (x_1, ..., x_d)\big).$$

Then (6.9) can be expressed as

$$\begin{aligned}\big(\widetilde{A} \circ \widetilde{R}_{p_1 \dots p_d}\big)(u) &= \widetilde{A}(x_1, ..., x_d)\, T\ \widetilde{A}_{kp_1}(x_1)\, T \cdots T\ \widetilde{A}_{dp_d}(x_d)\, T\ \widetilde{U}_{r(p_1, \dots, p_d)}(u) \\ &= H_{p_1 \dots p_d}(x_1, ..., x_d)\ T\ \widetilde{U}_{r(p_1, \dots, p_d)}(u).\end{aligned} \tag{6.10}$$

We assume that the rule base of a Mamdani fuzzy system is complete, that is, the base includes all fuzzy rules corresponding to all possible combinations of antecedent fuzzy sets $\widetilde{A}_{ij}$'s $(i = 1, ..., d;\ j = 0, \pm 1, ..., \pm m)$. So there exist $(2m+1)^d$ fuzzy rules in the rule base, and the size of the fuzzy rule base is $(2m+1)^d$. Let

$$Q_d(r) = \bigvee_{p_1, \dots, p_d = -m}^{m} \big\{\big|r(p_1, ..., p_d)\big|\big\}.$$

Also let $\widetilde{U}_{r(p_1, \dots, p_d)} \in \mathcal{F}([-b,\ b])$: $\mathrm{Ker}(\widetilde{U}_{r(p_1, \dots, p_d)}) = \big\{b \cdot r(p_1, ..., p_d)/Q_d(r)\big\}$, and $\widetilde{U}_{r(p_1, \dots, p_d)}$ be a fuzzy number. Using the generalized centroid defuzzification (see [18, 23, 31, 32, 43, 65, 66, 72, 77] etc), and (6.10) we can obtain a crisp output [31, 32, 72]:

$$M_m(x_1, ..., x_d) = \frac{\sum\limits_{p_1, \dots, p_d = -m}^{m} \frac{b}{Q_d(r)} \cdot r(p_1, ..., p_d) \cdot H_{p_1 \dots p_d}(x_1, ..., x_d)^{\alpha}}{\sum\limits_{p_1, \dots, p_d = -m}^{m} H_{p_1 \dots p_d}(x_1, ..., x_d)^{\alpha}}, \tag{6.11}$$

where $(x_1, ..., x_d) \in \mathbb{R}^d$ is system input, α : $0 \leq \alpha \leq +\infty$ is an adjustable parameter, and let $0/0 \equiv 0$. The I/O relationship (6.11) is called a generalized Mamdani fuzzy system.

By [9], if $\alpha = +\infty$, (6.11) is a fuzzy system with the maximum mean defuzzification [18, 65]; If $\alpha = 1$, (6.11) is a fuzzy system with the centroid defuzzier [26–29, 37, 38, 68, 76, 77]; If $\alpha = 0$, (6.11) is a weighted sum fuzzy system [19, 21, 41]; If $\alpha = 1 - \gamma + \gamma/d$ ($\gamma \in [0,1]$), (6.11) is a compensatory neuro-fuzzy systems [79].

Theorem 6.3 *Let μ be Lebesgue measure on $\mathbb{R}^d$. Then with $L_p(\mu)$−norm generalized Mamdani fuzzy systems can be universal approximators to SPLF's, that is, for any $S \in \mathcal{D}_d$, and $\forall \varepsilon > 0$, there is $m_0 \in \mathbb{N}$, so that $\forall m > m_0$, it follows that*

$$\Big(\int_{\mathbb{R}^d} |M_m(x_1,...,x_d) - S(x_1,...,x_d)|^p \mathrm{d}\mu\Big)^{\frac{1}{p}} < \varepsilon,$$

i.e. $\|M_m - S\|_{\mu,p} < \varepsilon$.

Proof. Let $[-a,\ a]^d$ be the support of $S \in \mathcal{D}_d$, and $\Delta_1, ..., \Delta_{N_S}$ be polyhedrons corresponding to S, moreover $\bigcup\limits_{j=1}^{N_S} \Delta_j = [-a,\ a]^d$. Let $x_0 \equiv 1$, and denote

$$S(x_1,...,x_d) = \begin{cases} \sum\limits_{i=0}^{d} s_{i1} \cdot x_i, & (x_1,...,x_d) \in \Delta_1, \\ \cdots\cdots \\ \sum\limits_{i=0}^{d} s_{iN_S} \cdot x_i, & (x_1,...,x_d) \in \Delta_{N_S}, \\ 0, & \text{otherwise.} \end{cases} \tag{6.12}$$

it is no harm to assume s_{ij} $(i = 0,1,...,d;\ j = 1,...,N_S)$ can be expressed as a decimal number (otherwise s_{ij} can be approximated by such a number). Denote

$$s_0 = \min\big\{s \in \mathbb{N}\big|10^s \cdot s_{ij} \in \mathbb{Z}, i = 0,1,...,d; j = 1,...,N_S\big\}.$$

Let $m_{ij} = 10^{s_0} \cdot s_{ij}$ $(i = 0,1,...,d;\ j = 1,...,N_S)$. Then $m_{ij} \in \mathbb{Z}$. Define the function $r: \{-m,\ -m+1,...,m-1,\ m\}^d \longrightarrow \mathbb{Z}$ as follows:

$$r(p_1,...,p_d) = \begin{cases} m \cdot \sum\limits_{i=0}^{d} m_{i1} \cdot \dfrac{ap_i}{m}, & \Big(\dfrac{ap_1}{m},...,\dfrac{ap_d}{m}\Big) \in \Delta_1, \\ \cdots\cdots \\ m \cdot \sum\limits_{i=0}^{d} m_{iN_S} \cdot \dfrac{ap_i}{m}, & \Big(\dfrac{ap_1}{m},...,\dfrac{ap_d}{m}\Big) \in \Delta_{N_S}, \end{cases} \tag{6.13}$$

It is easy to versify the following fact:

$$Q_d(r) = \bigvee_{p_1,...,p_d=-m}^{m} \big\{\big|r(p_1,...,p_d)\big|\big\} = m \cdot \bigvee_{j=1}^{N_S} \Big\{\sum_{i=0}^{d} \big|m_{ij}\big|\Big\}.$$

Put $b = 10^{-s_0} \cdot Q_d(r)$. For any $(p_1, ..., p_d) \in \{-m, -m+1, ..., m-1, m\}^d$, if $(ap_1/m, ..., ap_d/m) \in \Delta_j$, easily we can show

$$S\Big(\frac{ap_1}{m}, ..., \frac{ap_d}{m}\Big) = \sum_{i=0}^{d} s_{ij} \cdot \frac{ap_i}{m} = 10^{-s_0} \cdot \sum_{i=0}^{d} m_{ij} \frac{ap_i}{m} = \frac{b}{Q_d(r)} r(p_1, ..., p_d). \tag{6.14}$$

So $b \cdot r(p_1, ..., p_d)/Q_d(r) = S(ap_1/m, ..., ap_d/m)$. By Lemma 6.2, $\forall (x_1, ..., x_d) \in \mathbb{R}^d$, $(p_1, ..., p_d) \in N(x_1, ..., x_d)$, let $ap_i/m = x_i + \theta_{p_i}/m$, where $|\theta_{p_i}| \leq ac_0$. So by (6.11) (6.14) and Lemma 6.2 if let $\mathbf{x} = (x_1, ..., x_d)$ it follows that $H_{p_1 \cdots p_d}(x_1, ..., x_d) = H_{p_1 \cdots p_d}(\mathbf{x})$, $N(x_1, ..., x_d) = N(\mathbf{x})$, $S(x_1, ..., x_d) = S(\mathbf{x})$,

$$\begin{aligned}
\big\|M_m - S\big\|_{\mu,p}^p &= \int_{\mathbb{R}^d} \left| \frac{\sum\limits_{p_1,...,p_d=-m}^{m} S\big(\frac{ap_1}{m}, ..., \frac{ap_d}{m}\big) H_{p_1 \cdots p_d}(\mathbf{x})^\alpha}{\sum\limits_{p_1,...,p_m=-m}^{m} H_{p_1 \cdots p_d}(\mathbf{x})^\alpha} - S(\mathbf{x}) \right|^p \mathrm{d}\mu \\
&= \int_{[-a,a]^d} \frac{\left| \sum\limits_{p_1,...,p_d=-m}^{m} \Big(S\big(\frac{ap_1}{m}, ..., \frac{ap_d}{m}\big) - S(\mathbf{x}) \Big) H_{p_1 \cdots p_d}(\mathbf{x})^\alpha \right|^p}{\left| \sum\limits_{p_1,...,p_d=-m}^{m} H_{p_1 \cdots p_d}(\mathbf{x})^\alpha \right|^p} \mathrm{d}\mu \\
&= \int_{[-a,a]^d} \frac{\left| \sum\limits_{(p_1,...,p_d)\in N(\mathbf{x})} \Big(S\big(x_1 + \frac{\theta_{p_1}}{m}, ..., x_d + \frac{\theta_{p_d}}{m}\big) - S(\mathbf{x}) \Big) H_{p_1 \cdots p_d}(\mathbf{x})^\alpha \right|^p}{\left| \sum\limits_{(p_1,...,p_d)\in N(\mathbf{x})} H_{p_1 \cdots p_d}(\mathbf{x})^\alpha \right|^p} \mathrm{d}\mu \\
&\leq \int_{[-a,a]^d} \frac{\left[\sum\limits_{(p_1,...,p_d)\in N(\mathbf{x})} \Big| S\big(x_1 + \frac{\theta_{p_1}}{m}, ..., x_d + \frac{\theta_{p_d}}{m}\big) - S(\mathbf{x}) \Big| H_{p_1 \cdots p_d}(\mathbf{x})^\alpha \right]^p}{\left| \sum\limits_{(p_1,...,p_d)\in N(\mathbf{x})} H_{p_1 \cdots p_d}(\mathbf{x})^\alpha \right|^p} \mathrm{d}\mu \\
&\leq \int_{[-a,a]^d} \frac{\left(\sum\limits_{(p_1,...,p_d)\in N(\mathbf{x})} \sum\limits_{i=0}^{d} \frac{|\theta_{p_i}| D_i(S)}{m} \cdot H_{p_1 \cdots p_d}(\mathbf{x})^\alpha \right)^p}{\left| \sum\limits_{(p_1,...,p_d)\in N(\mathbf{x})} H_{p_1 \cdots p_d}(\mathbf{x})^\alpha \right|^p} \mathrm{d}\mu \\
&\leq \Big(\frac{ac_0}{m} \sum_{i=0}^{d} D_i(S) \Big)^p \cdot \int_{[-a,a]^d} \frac{\left(\sum\limits_{(p_1,...,p_d)\in N(\mathbf{x})} H_{p_1 \cdots p_d}(\mathbf{x})^\alpha \right)^p}{\left| \sum\limits_{(p_1,...,p_d)\in N(\mathbf{x})} H_{p_1 \cdots p_d}(\mathbf{x})^\alpha \right|^p} \mathrm{d}\mu \\
&\leq \Big(\frac{ac_0}{m} \sum_{i=0}^{d} D_i(S) \Big)^p \cdot \mu([-a,a]^d) = \Big(\frac{ac_0}{m} \sum_{i=0}^{d} D_i(S) \Big)^p \cdot (2a)^d.
\end{aligned} \tag{6.15}$$

Therefore, $\|M_m - S\|_{\mu,p} \leq 2^{d/p} \cdot a^{1+d/p} c_0 \cdot \sum_{i=0}^{d} D_i(S)/m$. Thus

$$m > \frac{2^{d/p} a^{1+d/p} c_0}{\varepsilon} \sum_{i=0}^{d} D_i(S), \Longrightarrow \|M_m - S\|_{\mu,p} < \varepsilon.$$

The theorem is proved. □

Corollary 6.3 *With $L_p(\mu)$−norm the generalized Mamdani fuzzy system M_m can be universal approximator, that is, if μ is Lebesgue measure on $\mathbb{R}^d$, then for any $\varepsilon > 0$ and $f \in L_p(\mu)$, there is $m_0 \in \mathbb{N}$, so that $\forall m > m_0$, $\|M_m - f\|_{\mu,p} < \varepsilon$.*

Proof. For $\varepsilon > 0$, By Theorem 6.1 and its proving procedure, there exist $a > 0$ and a SPLF $S \in \mathcal{D}_d$, such that

$$\text{Supp}(S) = \{(x_1, ..., x_d) \in \mathbb{R}^d | -a \leq x_i \leq a, i = 1, ..., d\}, \quad \|f - S\|_{\mu,p} < \frac{\varepsilon}{2}.$$

By Theorem 6.3, there is $m_0 \in \mathbb{N}$ satisfying: $\forall m > m_0$, $\|S - M_m\|_{\mu,p} < \frac{\varepsilon}{2}$. Therefore

$$\|M_m - f\|_{\mu,p} \leq \|M_m - S\|_{\mu,p} + \|f - S\|_{\mu,p} < \frac{\varepsilon}{2} + \frac{\varepsilon}{2} = \varepsilon.$$

Thus the conclusion is proved. □

For a given error bound $\varepsilon > 0$, next let us estimate the size of the fuzzy rule base of a fuzzy system.

Theorem 6.4 *Suppose μ is Lebesgue measure on $\mathbb{R}^d$, and $f : \mathbb{R}^d \longrightarrow \mathbb{R}$ is Riemann integrable. Then for arbitrary $\varepsilon > 0$, there are $h > 0$ and $a > 0$, if for any $i = 1, ..., d$, let $\mathbf{x}^{h^i} = (x_1, ..., x_{i-1}, x_i + h, x_{i+1}, ..., x_d)$ and*

$$D_H(f) = \bigvee_{i=1}^{d} \bigvee_{x_1,...,x_d, x_i+h \in [-a,a]} \{h^{-1} \cdot |f(\mathbf{x}^{h^i}) - f(x_1, ..., x_d)|\} = \bigvee_{i=1}^{d} \{D_i(f)\}. \tag{6.16}$$

we have, when $m > (2a)^{1+d/p} \cdot d \cdot D_H(f) \cdot c_0/\varepsilon$, it follows that $\|M_m - f\|_{\mu,p} < \varepsilon$.

Proof. By the proof of Theorem 6.1, for $\varepsilon > 0$, there is $a > 0$. Partition $[-a, a]^d$ identically into sufficiently small cubes, and then divide these small cubes into d dimensional polyhedrons $\Delta_1, ..., \Delta_N$. Thus we can define $S \in \mathcal{D}_d$, so that $\|f - S\|_{\mu,p} < \varepsilon/2$. Suppose the side length of these cubes is $h > 0$. By Corollary 6.2 easily we have, $D_H(f) = \bigvee_{i=1}^{d} \{D_i(S)\}$. Therefore, $\forall i = 1, ..., d$, $D_i(S) \leq D_H(f)$. Let $m > (2a)^{1+d/p} \cdot d \cdot D_H(f) \cdot c_0/\varepsilon$. So $m >$

$2^{1+d/p}a^{1+d/p}c_0 \sum_{i=1}^{d} D_i(S)/\varepsilon$. By Theorem 6.3 it follows that $\|M_m - S\|_{\mu,p} < \varepsilon/2$. Hence

$$\|f - M_m\|_{\mu,p} \le \|f - S\|_{\mu,p} + \|S - M_m\|_{\mu,p} < \frac{\varepsilon}{2} + \frac{\varepsilon}{2} = \varepsilon.$$

The theorem is therefore proved. □

In Theorem 6.4, if choose $c_0 = 2$, $p = d = 1$, and $\text{Supp}(f) \subset [-1, 1]^d$, that is, $a = 1$. Then the corresponding m satisfies: $m > 8D_H(f)/\varepsilon$. In the following we utilize a real example to demonstrate the realizing procedure of the approximation obtained.

Example 6.1 Choose $c_0 = 2$, and $p = d = 1$, i.e. we illustrate our example in one–dimensional space $\mathbb{R}$. Give the error bound $\varepsilon = 0.1$. Define the function $f : \mathbb{R} \longrightarrow \mathbb{R}$ as follows:

$$f(x) = \begin{cases} \frac{1}{8}\cos\frac{(2x+1)\pi}{2}, & -1 \le x \le -\frac{1}{2}, \\ \frac{1}{2}(x+\frac{1}{2})^2, & -\frac{1}{2} < x \le 0, \\ \frac{1}{2}x^2, & 0 < x \le \frac{1}{2}, \\ \frac{1}{8}\sin(\pi x), & \frac{1}{2} < x \le 1, \\ 0, & \text{otherwise.} \end{cases}$$

Easily we can show, if partition $[-1, 1]$ identically into 20 parts, and then obtain the SPLF $S \in \mathcal{D}_1$, so that $\text{Supp}(S) = [-1, 1]$, then $\|f - S\|_{\mu,p} < \varepsilon/2$. The curves of the functions f and S are shown in Figure 6.2 and Figure 6.3, respectively.

At first let $\alpha = 1$. By Theorem 6.4, $D_H(f) < 1$, and if $m > 4D_H(f)/0.1$, by (6.11) we can define the generalized Mamdani fuzzy system M_m satisfying $\|f - M_m\|_{\mu,p} < \varepsilon$. Choose $m = 40$, the size of fuzzy rule base corresponding to M_m is $(2m+1)^1 = 81$.

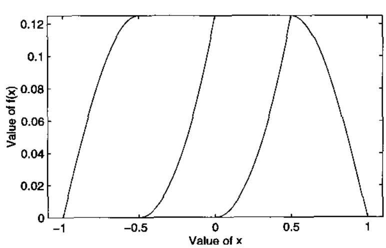


Figure 6.2 The curve of f

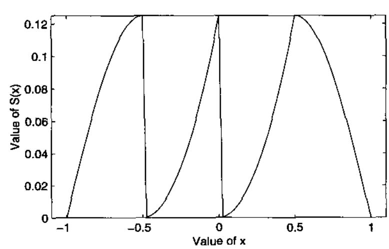


Figure 6.3 The curve of S

Let $\tilde{A}_{1j}$ $(j = 0, \pm1, ..., \pm40)$ be obtained by the translation of the fuzzy number $\tilde{A}$ defined as follows:

$$\tilde{A}(x) = \begin{cases} 40\left(\frac{1}{40} - x\right)x, & 0 \le x \le \frac{1}{40}, \\ 40\left(\frac{1}{40} + x\right), & -\frac{1}{40} \le x < 0, \\ 0, & \text{otherwise;} \end{cases}$$

$$\tilde{A}_{1(40)}(x) = \begin{cases} 40\left(x - \frac{39}{40}\right), & \frac{39}{40} \le x \le 1, \\ 0, & \text{otherwise;} \end{cases}$$

$$\tilde{A}_{1(-40)}(x) = \begin{cases} -40\left(x + \frac{39}{40}\right), & -1 \le x \le -\frac{39}{40}, \\ 0, & \text{otherwise.} \end{cases}$$

And $\tilde{A}_{1j}(x) = \tilde{A}(x - j/40)$ $(j = 0, \pm1, ..., \pm39)$.

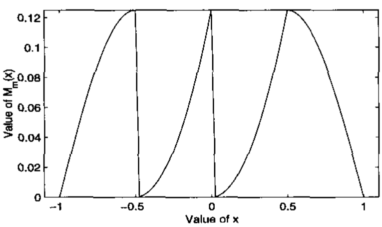


Figure 6.4 I/O of M_m when $\alpha = 1$

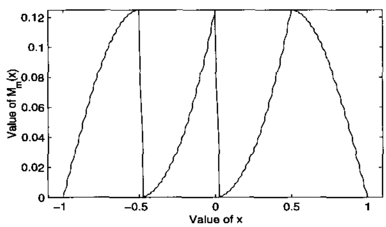


Figure 6.5 I/O of M_m when $\alpha = 1/2$

Since $H_j(x) = \tilde{A}_{1j}(x)$, by (6.11) (6.15) and the fact $f(j/40) = S(j/40)$ we obtain the I/O relationship of the fuzzy system $M_m(\cdot)$ as follows:

$$M_m(x) = \frac{\sum_{j=-40}^{40} \tilde{A}_{1j}(x) \cdot f\left(\frac{j}{40}\right)}{\sum_{j=-40}^{40} \tilde{A}_{1j}(x)}.$$

I/O relationship curve of the corresponding generalized Mamdani fuzzy system is shown in Figure 6.4.

Then choose $\alpha = 1/2$, the corresponding generalized Mamdani fuzzy system can expressed as

$$M_m(x) = \frac{\sum_{j=-40}^{40} (\tilde{A}_{1j}(x))^{\frac{1}{2}} \cdot f\left(\frac{j}{40}\right)}{\sum_{j=-40}^{40} (\tilde{A}_{1j}(x))^{\frac{1}{2}}}.$$

Figure 6.5 illustrates the I/O relationship curve.

From above example we can find that it is easy to realize the approximation of the given function by a generalized Mamdani fuzzy system, moreover with $L_p(\mu)-$norm the high approximating accuracy can be ensured.

6.2.2 Generalized T–S fuzzy system

The consequent of a T–S type fuzzy inference is the function of input variables. If the rule base of a fuzzy system consists of T–S inference rules, the fuzzy system is called a T–S fuzzy system. Given the antecedent fuzzy sets $\widetilde{A}_{1p_1}, ..., \widetilde{A}_{dp_d} \in \widetilde{\mathcal{O}}_0(a,\ m)$, in the subsection the consequent function is chosen to be a linear function of the input variables. T–S fuzzy inference rule can be expressed as follows [33, 34, 73–75]:

$$TR_{p_1...p_d}:\ \text{IF } x_1 \text{ is } \widetilde{A}_{1p_1} \text{ and } x_2 \text{ is } \widetilde{A}_{2p_2} \text{ and ... and } x_d \text{ is } \widetilde{A}_{dp_d}$$
$$\text{THEN } u \text{ is } b_{0;p_1...p_d} + b_{1;p_1...p_d}x_1 + \cdots + b_{d;p_1...p_d}x_d.$$

where $b_{k;p_1\cdots p_d}$ $(k = 0, 1, ..., d)$ is an adjustable real parameter. Similarly with Mamdani fuzzy system, we assume the fuzzy rule base for a T–S fuzzy system is complete, i.e. the base includes all fuzzy rules corresponding to all possible combinations of $\widetilde{A}_{1p_1}, ..., \widetilde{A}_{dp_d} \in \widetilde{\mathcal{O}}_0(a,\ m)$ for $p_1, ..., p_d = -m,\ -m+1, ..., m-1,\ m)$. The size of the rule base is also $(2m+1)^d$. Based on the generalized centroid defuzzifier we can define a generalized T–S fuzzy system whose I/O relationship is [33, 62, 73, 75]

$$T_m(x_1, ..., x_d) = \frac{\sum\limits_{p_1,...,p_d=-m}^{m} \Big(H_{p_1...p_d}(x_1, ..., x_d)^{\alpha} \cdot \big(\sum\limits_{k=0}^{d} b_{k;p_1...p_d}x_k\big)\Big)}{\sum\limits_{p_1,...,p_d=-m}^{m} H_{p_1...p_d}(x_1, ..., x_d)^{\alpha}}, \quad (6.17)$$

where let $0/0 = 0$, and $\alpha \in [0, +\infty]$ is an adjustable, $x_0 \equiv 1$. Similarly with (6.11), if $\alpha = 1$, (6.17) is a T–S system with centroid defuzzification; If $\alpha = +\infty$, (6.17) is a T–S fuzzy system with maximum mean defuzzification; If $\alpha = 0$, (6.17) is also a weighted sum T–S system. In (6.17) if $\forall k \in \{1, ..., d\}$, $b_{k;p_1...p_d} = 0$, the system is call a simple generalized T–S fuzzy system.

Theorem 6.5 *Suppose μ is Lebesgue measure on $\mathbb{R}^d$. Then generalized T–S fuzzy systems can be universal approximators to SPLF's with $L_p(\mu)-$norm, that is, for any $S \in \mathcal{D}_d$, and $\forall \varepsilon > 0$, there is $m \in \mathbb{N}$, so that*

$$\Big(\int_{\mathbb{R}^d} |T_m(x_1, ..., x_d) - S(x_1, ..., x_d)|^p \mathrm{d}\mu\Big)^{\frac{1}{p}} < \varepsilon, \quad i.e.\ \big\|T_m - S\big\|_{\mu,p} < \varepsilon.$$

Proof. If $S \in \mathcal{D}_d$, as in Theorem 6.3, let $\mathrm{Supp}(S) = [-a,\ a]^d$, and define S as (6.12). Define the parameter $b_{k;p_1\cdots p_m}$ $(k = 0, 1, ..., d;\ p_1, ..., p_d = -m,\ -m+$

$1, ..., m-1, m$) as follows:

$$\begin{cases} b_{0;p_1...p_d} = S\Big(\dfrac{p_1}{m}, ..., \dfrac{p_d}{m}\Big), \\ b_{k;p_1...p_d} = 0, \quad k = 1, ..., d. \end{cases} \tag{6.18}$$

By lemma 6.2, $\forall (p_1, ..., p_d) \in N(x_1, ..., x_d)$, denote $p_i/m = x_i + \theta_{p_i}/m$ ($i = 1, ..., d$), then $|\theta_{p_i}| \leq ac_0$. Using Lemma 6.1 we can show

$$(p_1, ..., p_d) \in N(x_1, ..., x_d), \Longrightarrow \Big|S(x_1, ..., x_d) - S\Big(\frac{p_1}{m}, ..., \frac{p_d}{m}\Big)\Big| \leq \frac{ac_0}{m}\sum_{i=1}^{d} D_i(S). \tag{6.19}$$

So similarly with (6.15) and using (6.18) (6.19), we let $\mathbf{x} = (x_1, ..., x_d)$ and obtain

$$\|T_m - S\|_{\mu,p}^p = \int_{\mathbb{R}^d}\left|\frac{\sum\limits_{p_1,...,p_d=-m}^{m}\Big(H_{p_1...p_d}(\mathbf{x})^\alpha \cdot \big(\sum\limits_{k=0}^{d} b_{k;p_1...p_d} x_k\big)\Big)}{\sum\limits_{p_1,...,p_m=-m}^{m} H_{p_1...p_d}(\mathbf{x})^\alpha} - S(\mathbf{x})\right|^p d\mu$$

$$= \int_{[-a,a]^d} \frac{\left|\sum\limits_{p_1,...,p_d=-m}^{m}\Big(H_{p_1...p_m}(\mathbf{x})^\alpha \cdot \big(\sum\limits_{k=0}^{d} b_{k;p_1...p_d} x_k - S(\mathbf{x})\big)\Big)\right|^p}{\left|\sum\limits_{p_1,...,p_d=-m}^{m} H_{p_1...p_d}(\mathbf{x})^\alpha\right|^p} d\mu$$

$$= \int_{[-a,a]^d} \frac{\left|\sum\limits_{(p_1,...,p_d)\in N(\mathbf{x})} H_{p_1...p_d}(\mathbf{x})^\alpha\Big(S\big(\frac{p_1}{m}, ..., \frac{p_d}{m}\big) - S(\mathbf{x})\Big)\right|^p}{\left|\sum\limits_{(p_1,...,p_d)\in N(\mathbf{x})} H_{p_1,...,p_d}(\mathbf{x})^\alpha\right|^p} d\mu$$

$$\leq \int_{[-a,a]^d} \frac{\Big\{\sum\limits_{(p_1,...,p_d)\in N(\mathbf{x})} H_{p_1...p_d}(\mathbf{x})^\alpha \Big|S\big(\frac{p_1}{m}, ..., \frac{p_d}{m}\big) - S(\mathbf{x})\Big|\Big\}^p}{\left|\sum\limits_{(p_1,...,p_d)\in N(\mathbf{x})} H_{p_1,...,p_d}(\mathbf{x})^\alpha\right|^p} d\mu$$

$$\leq \Big(\frac{ac_0}{m}\sum_{i=1}^{d} D_i(S)\Big)^p \mu([-a,a]^d) = \Big(\frac{ac_0}{m}\sum_{i=1}^{d} D_i(S)\Big)^p (2a)^d.$$

Hence if choose $m > c_0 \cdot 2^{d/p} \cdot a^{1+d/p} \sum\limits_{i=1}^{d} D_i(S)/\varepsilon$, we have, $\|T_m - S\|_{\mu,p} < \varepsilon$. □

By above proof procedure the generalized T–S fuzzy systems using for approximating the given functions can be chosen as the simple generalized T–S fuzzy systems. Similarly with Corollary 6.3 and Theorem 6.4, using Theorem 6.1 and Theorem 6.5 easily we can show

Corollary 6.4 *Let μ be Lebesgue measure on $\mathbb{R}^d$. Give arbitrarily $f \in L_p(\mu)$, and $\forall \varepsilon > 0$. For $h > 0$, $a > 0$, suppose $D_H(f)$ is defined by (6.16). Then the following facts hold:*

(i) There is $m \in \mathbb{N}$, so that $\|T_m - f\|_{\mu,p} < \varepsilon$, i.e. with $L_p(\mu)-$norm generalized T–S fuzzy systems are universal approximators;

(ii) When h is sufficiently small, and $m > (2a)^{1+d/p} \cdot d \cdot D_H(f) \cdot c_0/\varepsilon$, we have, $\big\|T_m - f\big\|_{\mu,p} < \varepsilon$.

In Corollary 6.4, if choose $c_0 = 2$, $d = 2$, $p = 1$, $a = 1$, then the corresponding m satisfies: $m \geq 32D_H(f)/\varepsilon$; If $d = 1$, $c_0 = 2$, $p = 1$, $a = 1$, then $m \geq 8D_H(f)/\varepsilon$.

Example 6.2 Let $c_0 = 2$, $p = 1$, $\alpha = 1$. In $\mathbb{R}$, i.e. $d = 1$ we discuss our example. Define $f : \mathbb{R} \longrightarrow \mathbb{R}$ as follows:

$$f(x) = \begin{cases} \dfrac{1}{4}\sin\dfrac{\pi}{4}(1+x), & -1 \leq x \leq 0, \\ \dfrac{1}{8}\cos\dfrac{\pi}{2}x, & 0 < x \leq 1, \\ 0, & \text{otherwise.} \end{cases}$$

Choose error bound $\varepsilon = 0.1$. Easily we know, if partition $[-1, 1]$ identically into $2m$ ($m \geq 10$) parts, a SPLF $S \in \mathcal{D}_1^0$ is defined so that $\|f - S\|_{\mu,p} < \varepsilon/2$. The curves of the functions f, S is shown in Figure 6.6 and Figure 6.7, respectively.

Obviously $D_H(f) \leq \pi/8$. By Corollary 6.4, $m > 8\pi/0.8 = 10\cdot\pi$, let $m = 32$. The size of the fuzzy rule base is $(2 \times 32 + 1)^1 = 65$. By Remark 6.2, $f\big(j/32\big) = S\big(j/32\big)$ $(j = 0, \pm1, ..., \pm32)$. Define $\widetilde{A}_{1j}$ $(j = 0, \pm1, ..., \pm32)$ as the translation of the following triangular fuzzy number $\widetilde{A}$:

$$\widetilde{A}(x) = \begin{cases} 32\Big(x + \dfrac{1}{32}\Big), & -\dfrac{1}{32} \leq x \leq 0, \\ 32\Big(\dfrac{1}{32} - x\Big), & 0 < x \leq \dfrac{1}{32}, \\ 0 & \text{otherwise.} \end{cases}$$

$$\widetilde{A}_{1(32)}(x) = \begin{cases} 32\Big(x - \dfrac{15}{32}\Big), & \dfrac{15}{32} \leq x \leq 1, \\ 0, & \text{otherwise;} \end{cases}$$

$$\widetilde{A}_{1(-32)}(x) = \begin{cases} -32\Big(x + \dfrac{15}{32}\Big), & -1 \leq x \leq -\dfrac{15}{32}, \\ 0, & \text{otherwise.} \end{cases}$$

$\widetilde{A}_{1j}(x) = \widetilde{A}\big(x - j/32\big)$ $(j = 0, \pm1, ..., \pm31)$. Since $d = p = 1$, $H_{p_1}(x) = \widetilde{A}_{1p_1}(x)$.

Thus, by (6.17) it follows that

$$T_m(x) = \frac{\sum\limits_{j=-32}^{32} \widetilde{A}_{1j}(x)^{\alpha} f(\frac{j}{32})}{\sum\limits_{j=-32}^{32} \widetilde{A}_{1j}(x)^{\alpha}},$$

Choose respectively $\alpha = 1$, $\alpha = 1/3$, the I/O relationship curves of the corresponding generalized T–S fuzzy systems are illustrated in Figure 6.8 and Figure 6.9, respectively, by which we can see by comparing that the generalized T–S fuzzy system approximation with $L_1(\mu)$−norm also possesses high accuracy at each point.

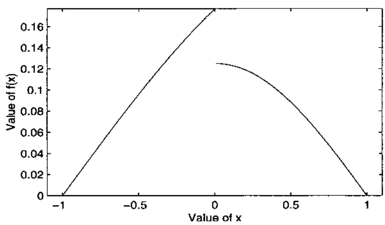


Figure 6.6 The curve of f

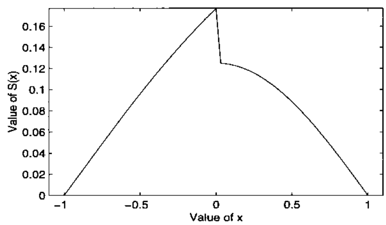


Figure 6.7 The curve of SPLF S

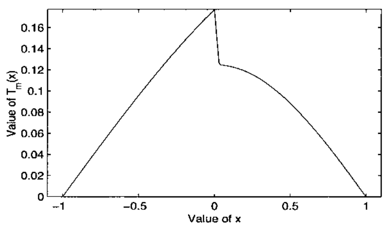


Figure 6.8 I/O of T_m when $\alpha = 1$

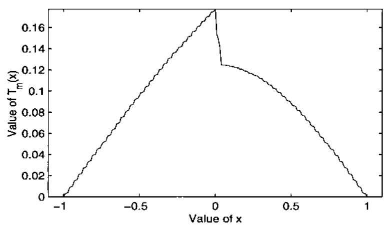


Figure 6.9 I/O of T_m when $\alpha = 1/4$

Example 6.3 Consider two dimensional case, i.e. $d = 2$, and $f : \mathbb{R}^2 \longrightarrow \mathbb{R}$ is defined as follows:

$$f(x) = \begin{cases} \frac{1}{8}(x+y+1)^2, & -1 \le x+y < 0, \\ 1-(x+y), & 0 \le x+y \le 1, \\ 0, & \text{otherwise}. \end{cases}$$

For a given accuracy $\varepsilon = 0.2$, in order to compute the SPLF S, satisfying $\|f - S\|_{[-1,1],p} < \varepsilon/2$, we partition $[-1,1]^2$ identically into sufficiently small squares. So by choosing $h = 0.01$, we can get, $D_H(f) \le 1$. Thus, we may choose $n = (2 \times 2 \times 2 \times 4 \times 1)/0.2 = 160$. The size of the fuzzy rule base is $(2 \times 160 + 1)^2 = 103041$. If let $\varepsilon = 0.1$, by Corollary 6.4 it is necessary to choose

$n = 320$, and in the fuzzy rule base there must be $(2 \times 320 + 1) = 641^2 = 410881$ fuzzy rules, which increases very quickly which n increases (i.e. 'rule explosion'). That will result in much inconvenience when fuzzy systems are employed to deal with high dimensional I/O mappings. So it is of significance in practice how to improve the structures of fuzzy systems to solve such a problem, which is the research subject in §6.3.

In order to utilize the common advantages of generalized T–S fuzzy systems and generalized Mamdani fuzzy systems, let us now synthesize such two systems as one within a general framework, which is called generalized fuzzy system. To this end we at first develop a type of fuzzy inference rules as follows:

$$G_{p_1...p_d}: \text{ IF } x_1 \text{ is } \widetilde{A}_{1p_1} \text{ and } x_2 \text{ is } \widetilde{A}_{2p_2} \text{ and ... and } x_d \text{ is } \widetilde{A}_{dp_d}$$
$$\text{THEN } u \text{ is } \lambda\ \widetilde{V}_{t(p_1...p_d;x_1,...,x_d)} + (1-\lambda)\ \widetilde{U}_{r(p_1,...,p_d)},$$

we let $0/0 = 0$, $\lambda \in [0,1]$ is an adjustable parameter, and

$$t(p_1...p_d;\, x_1, ..., x_d) = \sum_{i=0}^{d} b_{i;p_1...p_d} \cdot x_i,$$

moreover, $\widetilde{V}_x$ is a singleton fuzzy set, that is, $\widetilde{V}_x$ is equivalent to the real number x : $\widetilde{V}_x\ (x) = 1, \widetilde{V}_x\ (x') = 0\ (x \neq x')$. Corresponding to above fuzzy rules we define a generalized fuzzy system [31]

$$F_m(\mathbf{x}) = \frac{\sum\limits_{p_1,...,p_d=-n}^{n} H_{p_1...p_d}(\mathbf{x})^{\alpha}\Big((1-\lambda)\frac{b}{Q_d(r)}r(p_1,...,p_d) + \lambda \sum\limits_{i=0}^{d} b_{i;p_1...p_d}x_i\Big)}{\sum\limits_{p_1,...,p_d=-n}^{n} H_{p_1...p_d}(\mathbf{x})^{\alpha}}, \tag{6.20}$$

where $\mathbf{x} = (x_1, ..., x_d)$, $\lambda \in [0,1]$, $0 \le \alpha \le +\infty$, $x_0 \equiv 1$. If $\lambda = 1$, then (6.20) is a generalized T–S fuzzy system; If $\lambda = 0$, (6.20) is a generalized Mamdani fuzzy system. In practice (6.20) defines a general fuzzy system, and it concludes most of fuzzy systems in application as its special cases [31].

Using Theorem 6.3, Theorem 6.4 and Theorem 6.5 we can show the following conclusion.

Corollary 6.5 *Suppose μ is Lebesgue measure on $\mathbb{R}^d$. For any $f \in L_p(\mu)$, and $\forall \varepsilon > 0$, the following facts hold:*

(i) $\forall S \in \mathcal{D}_d$, the generalized fuzzy system defined by (6.20) is universal approximator to S with $L_p(\mu)$–norm;

(ii) There is $m \in \mathbb{N}$, so that $\|F_m - f\|_{\mu,p} < \varepsilon$, that is, with $L_p(\mu)$–norm generalized fuzzy systems can be universal approximators;

(iii) There is a sufficiently small $h > 0$, if $D_H(f)$ is defined as (6.16), then

$$m > \frac{(2a)^{1+d/p} \cdot d \cdot D_H(f) \cdot c_0}{\varepsilon}, \Longrightarrow \|F_m - f\|_{\mu,p} < \varepsilon.$$

Proof. Let S be determined by (6.12), the d-variate function r be defined by (6.13). Similarly with Theorem 6.3, we define $Q(r)$ and $b > 0$. Using (6.18) we determine the parameter $b_{k;p_1...p_d}$ $(k = 0, 1, ..., d)$. Easily we can show

$$(1-\lambda)\frac{b}{Q_d(r)}r(p_1, ..., p_d) + \lambda\sum_{i=0}^{d} b_{i;p_1...p_d}x_i = S\Big(\frac{p_1}{m}, ..., \frac{p_d}{m}\Big).$$

Also similarly with (6.15) (6.19), we can get

$$m > \frac{c_0 \cdot 2^{d/p} \cdot a^{1+d/p}}{\varepsilon}\sum_{i=1}^{d} D_i(S), \Longrightarrow \|F_m - S\|_{\mu,p} \leq \frac{ac_0}{m}\sum_{i=1}^{d} D_i(S)(2a)^{d/p} < \varepsilon.$$

(i) is proved. (ii) and (iii) are direct results of Corollary 6.3 and Corollary 6.4. □

By comparison between Corollary 6.5 and Corollary 6.4 we get, if the accuracy related is identical, the size of fuzzy rule base of generalized fuzzy system (6.20) is same with ones of generalized T–S fuzzy system and generalized Mamdani fuzzy system, independent of λ.

Remark 6.3 If $\lim\limits_{m\to+\infty} \xi(a, m) = 0$, and $\widetilde{\mathcal{O}}(a, m)$ constitute the antecedent fuzzy set family, similarly we can show, both generalized Mamdani fuzzy systems and generalized T–S fuzzy systems can be universal approximators with $L_p(\mu)$−norm.

In the section we focus on Mamdani fuzzy systems and T–S fuzzy systems with a general framework. Taking SPLF's as bridges we analyze universal approximation of generalized fuzzy systems to p−integrable functions. Thus, by [16, 78] this two generalized fuzzy systems can be universal approximators, respectively with maximum norm and integral norm.

§6.3 Hierarchical system of generalized T–S fuzzy system

In most rule based fuzzy systems, fuzzy rule base consisting of a number of inference rules defined as 'IF...THEN...' is a key part. The size of a complete rule base increases exponentially when the system input variable number increases, which is called 'Rule explosion'. Such a phenomenon is in nature the 'curse of dimensionality' which exists in many fields [10, 14]. That will not only generate complicated system structures, but also cause long computational time, even memory overload of the computer.

To make fuzzy systems usable in dealing with complex systems, we must solve the 'rule explosion' problem [5–7, 40, 49–51]. In the research for fuzzy systems or fuzzy controllers, two classes of such methods are significant. One is based on the equivalence of 'intersection rule configuration' and 'union rule configuration' [5, 6, 40]. That is, if P and Q are two antecedents, and R is a consequent, then

$$[(P \wedge Q) \Longrightarrow R] \Longleftrightarrow [(P \Longrightarrow R) \vee (Q \Longrightarrow R)].$$

Another one is to introduce a hierarchical system configuration [49–51], which we shall focus on in the section and the next section. By this hierarchy we can deal with effectively some large scale systems in application.

In order to analyze a HFS thoroughly, at first we show that the I/O relationships of HFS's may be represented as ones of standard fuzzy systems. The further result is the equivalence between fuzzy system and its HFS. For convenience to define a HFS we give a fuzzy set family as

$$\{\widetilde{A}_{ij} \,|i = 1, ..., d;\ j = -m,\ -m+1, ..., m-1,\ m\} \subset \widetilde{\mathcal{O}}_0(a,\ m),$$

where $a > 0$ is also a given real number.

6.3.1 Hierarchical fuzzy system

In the subsection we introduce a hierarchical structure to solve the 'rule explosion' problem, as shown in Figure 6.10. In this hierarchy, the first level fuzzy system gives an approximate output y_1^m, which is modified by the second level fuzzy system as an input variable; the third level system will modify the output y_2^m of the second level fuzzy system; ... and so on. This process is repeated in succeeding subsystems of the hierarchy.

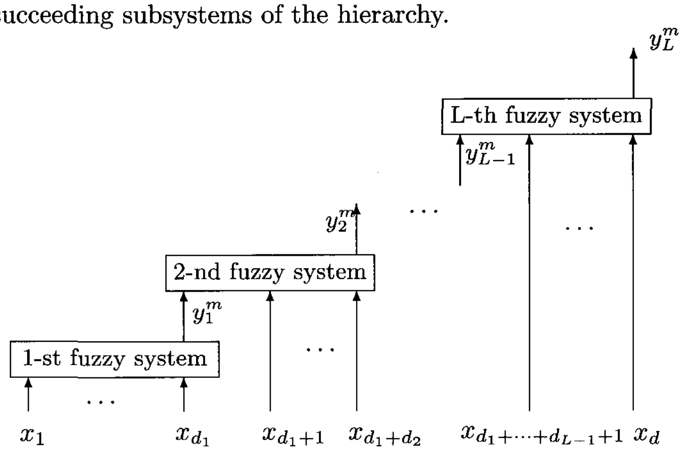


Figure 6.10 Hierarchical fuzzy system

In Figure 6.10 $y_1^m, ..., y_{L-1}^m$ are also employed as the intermediate input variables of the corresponding subsystems. For each $k = 1, ..., L-1$, we introduce fuzzy number $\widetilde{B}_{kj} \in \mathcal{F}(\mathbb{R})$ ($j = 0, \pm1, ..., \pm m$) as the antecedent fuzzy set for the input variable y_k^m, such that $\forall y \in \mathbb{R}$, $\forall k \in \{1, ..., L-1\}$, it follows that

$$\text{Card}\big(\{j \in \{-m, -m+1, ..., m-1, m\} \big| \widetilde{B}_{kj}(y) > 0\}\big) \geq 1. \tag{6.21}$$

In Figure 6.10, The first level and j−th ($j = 2, ..., L$) level are T–S fuzzy systems. There exist d_1 input variables $x_1, ..., x_{d_1}$ in the first level, and there are d_j input variables $x_{d_{j-1}+1}, ..., x_{d_{j-1}+d_j}$ and a intermediate variable y^m_{j-1} that is the output of the $(j-1)$−th level fuzzy system, in the j−th level. The following fuzzy inference rules being used in the first and j−th level fuzzy systems respectively, will be employed to define the I/O relationship of a HFS:

IF x_1 is $\tilde{A}_{1p_1}$ and $\cdots$ and x_{d_1} is $\tilde{A}_{d_1p_{d_1}}$ THEN y^m_1 is $b^1_{0;p_1\cdots p_{d_1}} + \sum\limits_{i=1}^{p_1} b^1_{i;p_1\cdots p_{d_1}} x_i$;

IF $x_{d_{j-1}+1}$ is $\tilde{A}_{(d_{j-1}+1)p_1}$ and $\cdots$ and $x_{d_{j-1}+d_j}$ is $\tilde{A}_{(d_{j-1}+d_j)p_{d_j}}$ and y^m_{j-1} is $\tilde{B}_{j-1}$

THEN y_j is $b^j_{0;p_1\cdots p_{d_j}} + c^j_q y^m_{j-1} + \sum\limits_{i=1}^{d_j} b^j_{i;p_1\cdots p_{d_j}} x_{d_{j-1}+i}$,

where $j = 2, ..., L$; $\sum\limits_{j=1}^{L} d_j = d$; $q, p_1, p_2, ... \in \{-m, -m+1, ..., m-1, m\}$, and $b^j_{i;p_1\cdots p_j}$, c^j_q are adjustable parameters.

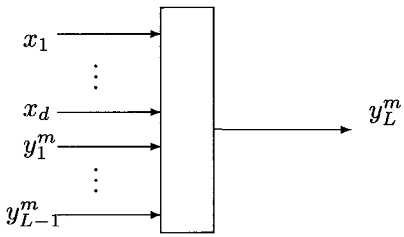


Figure 6.11 Fuzzy system with intermediate input variables

Through the hierarchy as in Figure 6.10 the size of fuzzy rule base of a fuzzy system can increase as the linear function of input variable number [63].

Proposition 6.1 *Suppose there are L levels in a HFS, in which exist d_1 input variables in the first level and exist $d_j + 1$ input variables in which the intermediate variable y^m_{j-1} is included, in the $j-th$ level ($j = 2, ..., L$). If $d_1 = d_j + 1 = c$, then the size of the fuzzy rule base related to this HFS is $(2m+1)^c(d-1)/(c-1)$.*

Proof. Let s be the size of the rule base, i.e. the total number of fuzzy rules related to the HFS as Figure 6.10. Obviously we have

$$s = (2m+1)^{d_1} + \sum_{j=2}^{L} (2m+1)^{d_j+1} = (2m+1)^c + (L-1)(2m+1)^c = L(2m+1)^c.$$

Since $d = \sum\limits_{j=1}^{L} d_j = c + \sum\limits_{j=1}^{L} (c-1) = Lc - L + 1$, thus, $L = (d-1)/(c-1)$. Hence $s = (d-1)(2m+1)^c/(c-1)$. The proposition is therefore proved. □

In Proposition 6.1, let $c = 2$, then $s = (d-1)(2m+1)^2$. We introduce intermediate variables as $y_1^m, ..., y_{L-1}^m$ in HFS shown in Figure 6.10, and obtain a fuzzy system whose input variables are $x_1, ..., x_d$; $y_1^m, ..., y_{L-1}^m$, as shown in Figure 6.11.

Next let us prove the equivalence between the I/O relationship of the HFS as Figure 6.10 and one of the fuzzy system as Figure 6.11. To this end we at first analyze I/O relationships of generalized hierarchical T–S fuzzy systems, thoroughly.

6.3.2 Generalized hierarchical T–S fuzzy system

In the following we introduce the notation: $\mathbf{x}_k^n = (x_{k+1}, ..., x_{k+n}) \in \mathbb{R}^n$ for $k, n \in \mathbb{N}$. For $\mathbf{x} = (x_1, ..., x_d) = \mathbf{x}_0^d \in \mathbb{R}^d$, give the indices j and $k \in \mathbb{N}$, satisfying $0 \le j < j+k \le d$. If $p_1, ..., p_k \in \{-m, -m+1, ..., m-1, m\}$, denote

$$H_{p_1 \dots p_k}(x_{j+1}, ..., x_{j+k}) = \widetilde{A}_{(j+1)p_1}(x_{j+1}) \times \cdots \times \widetilde{A}_{(j+k)p_k}(x_{j+k}) \tag{6.22}$$

If $j = 0$, $j+k = d$, the relation structure as shown in Figure 6.10 is a T–S fuzzy system, whose I/O relationship $(x_1, ..., x_d) \longrightarrow T_m(x_1, ..., x_d)$ is determined by (6.17); If $j > 0$, $j+k < d$, then in the HFS shown in Figure 6.10, the I/O relationships of the first level and $j-$th level fuzzy systems are also determined by the generalized T–S fuzzy systems as following:

$$\begin{cases} y_1^m = T_m^1(\mathbf{x}_0^{d_1}) = \dfrac{\sum\limits_{p_1,...,p_{d_1}=-m}^{m} H_{p_1 \dots p_{d_1}}(\mathbf{x}_0^{d_1})^{\alpha}\left(b_{0;p_1\dots p_{d_1}}^1 + \sum\limits_{j=1}^{d_1} b_{j;p_1\dots p_{d_1}}^1 x_j\right)}{\sum\limits_{p_1,...,p_{d_1}=-m}^{m} H_{p_1 \dots p_{d_1}}(\mathbf{x}_0^{d_1})^{\alpha}}, \\ y_j^m = T_m^j(\mathbf{x}_{l_j}^{d_j}, y_{j-1}^m) = \dfrac{\sum\limits_{q,p_1,...,p_{d_j}=-m}^{m} \left[H_{p_1 \dots p_{d_j}}(\mathbf{x}_{l_j}^{d_j})\widetilde{B}_q(y_{j-1}^m)\right]^{\alpha} Z(j, y_{j-1}^m)}{\sum\limits_{q,p_1,...,p_{d_j}=-m}^{m} \left[H_{p_1 \dots p_{d_j}}(\mathbf{x}_{l_j}^{d_j})\widetilde{B}_q(y_{j-1}^m)\right]^{\alpha}}, \end{cases} \tag{6.23}$$

where we let $0/0 = 0$; $j = 2, ..., L$, $l_j = \sum\limits_{k=1}^{j-1} d_k$. The I/O relationship (6.23) is called a generalized hierarchical fuzzy system (generalized HFS), where

$$Z(j, y_{j-1}^m) = b_{0;p_1\dots p_{d_j}}^j + c_q^j y_{j-1}^m + \sum_{i=1}^{d_j} b_{i;p_1\dots p_{d_j}}^j x_{l_j+i}. \tag{6.24}$$

In order to analyze the I/O relationship of the generalized HFS we firstly present the following lemma.

Lemma 6.3 *Suppose* $\mathbf{x} = \mathbf{x}_0^d = (x_1, ..., x_d) \in \mathbb{R}^d$, *and* $i, j, k_1, k_2 \in \mathbb{N}$ *satisfy:* $1 \le i < j < i+k_1+k_2 \le d$, $i+k_1 = j$. *Then for any* $\alpha :\ 0 < \alpha \le +\infty$, *it follows that*

$$\Big(\sum_{p_1,\dots,p_{k_1}=-m}^{m} H_{p_1\dots p_{k_1}}\big(\mathbf{x}_i^{k_1}\big)^{\alpha}\Big)\cdot\Big(\sum_{q_1,\dots,q_{k_2}=-m}^{m} H_{q_1\dots q_{k_2}}\big(\mathbf{x}_j^{k_2}\big)^{\alpha}\Big)$$
$$=\sum_{p_1,\dots,p_{k_1+k_2}=-m}^{m} H_{p_1\dots p_{k_1+k_2}}\big(\mathbf{x}_i^{k_1+k_2}\big)^{\alpha}.$$

Proof. By (6.22), it follows that the following inequalities hold:

$$H_{p_1\dots p_{k_1}}\big(\mathbf{x}_i^{k_1}\big)^{\alpha}=\widetilde{A}_{(i+1)p_1}(x_{i+1})^{\alpha}\times\cdots\times\widetilde{A}_{(i+k_1)p_{k_1}}(x_{i+k_1})^{\alpha},$$
$$H_{q_1\dots q_{k_2}}\big(\mathbf{x}_j^{k_2}\big)^{\alpha}=\widetilde{A}_{(j+1)q_1}(x_{j+1})^{\alpha}\times\cdots\times\widetilde{A}_{(j+k_2)q_{k_2}}(x_{j+k_2})^{\alpha}.$$

Considering $i+k_1=j$, we get

$$\Big(\sum_{p_1,\dots,p_{k_1}=-m}^{m} H_{p_1\dots p_{k_1}}\big(x_i^{k_1}\big)^{\alpha}\Big)\cdot\Big(\sum_{q_1,\dots,q_{k_2}=-m}^{m} H_{q_1\dots q_{k_2}}(\mathbf{x}_j^{k_2})^{\alpha}\Big)$$
$$=\Big(\sum_{p_1,\dots,p_{k_1}=-m}^{m}\big(\widetilde{A}_{(i+1)p_1}(x_{i+1})^{\alpha}\times\cdots\times\widetilde{A}_{(i+k_1)p_{k_1}}(x_{i+k_1})^{\alpha}\big)\Big)\cdot$$
$$\cdot\Big(\sum_{q_1,\dots,q_{k_2}=-m}^{m}\widetilde{A}_{(j+1)q_1}(x_{j+1})^{\alpha}\times\cdots\times\widetilde{A}_{(j+k_2)q_{k_2}}(x_{j+k_2})^{\alpha}\Big)$$
$$=\sum_{p_1,\dots,p_{k_1},q_1,\dots,q_{k_2}=-m}^{m}\Big(\widetilde{A}_{(i+1)p_1}(x_{i+1})^{\alpha}\times\cdots\times\widetilde{A}_{(i+k_1)p_{k_1}}(x_{i+k_1})^{\alpha}\times$$
$$\widetilde{A}_{(i+k_1+1)q_1}(x_{i+k_1+1})^{\alpha}\times\cdots\times\widetilde{A}_{(i+k_1+k_2)q_{k_2}}(x_{i+k_1+k_2})^{\alpha}\Big)$$
$$=\sum_{p_1,\dots,p_{k_1+k_2}=-m}^{m}\Big(\widetilde{A}_{(i+1)p_1}(x_{i+1})^{\alpha}\times\cdots\times\widetilde{A}_{(i+k_1+k_2)p_{k_1+k_2}}(x_{i+k_1+k_2})^{\alpha}\Big)$$
$$=\sum_{p_1,\dots,p_{k_1+k_2}=-m}^{m} H_{p_1\dots p_{k_1+k_2}}(x_{i+1},\dots,x_{i+k_1+k_2})^{\alpha}.$$

Considering $\mathbf{x}_i^{k_1+k_2}=(x_{i+1},\dots,x_{i+k_1+k_2})$, we can prove the lemma. □

Theorem 6.6 *Suppose $y_1^m,\dots,y_L^m$ are defined by (6.23). Then for any $j=2,\dots,L$, there exist $a_0(I^j,\,P^j)$, $a_i(I^j,\,P^j)$ for $i=1,\dots,l_j+d_j$, such that if we denote*

$$I^j=(i_1,\dots,i_{j-1}),\ P^j=(p_1,\dots,p_{l_j+d_j}),$$
$$J(I^j;y_1^m,\dots,y_{j-1}^m)\triangleq\widetilde{B}_{i_1}(y_1^m)\times\cdots\times\widetilde{B}_{i_{j-1}}(y_{j-1}^m),$$
$$O\big(\mathbf{x}_0^{l_{j+1}}\big)=O(x_1,\dots,x_{l_{j+1}})=a_0(I^j,P^j)+\sum_{i=1}^{d_j+l_j}a_i(I^j,P^j)x_i,$$

we can conclude that

$$y_j^m = \frac{\sum\limits_{i_1,\dots,i_{j-1};p_1,\dots,p_{l_j+d_j}=-m}^{m} \left[H_{p_1\dots p_{d_j+l_j}}\left(\mathbf{x}_0^{d_j+l_j}\right)J(I^j;y_1^m,\dots,y_{j-1}^m)\right]^{\alpha} O\left(\mathbf{x}_0^{l_j+1}\right)}{\sum\limits_{i_1,\dots,i_{j-1};p_1,\dots,p_{d_j+l_j}=-m}^{m} \left[H_{p_1\dots p_{d_j+l_j}}\left(\mathbf{x}_0^{d_j+l_j}\right)J(I^j;y_1^m,\dots,y_{j-1}^m)\right]^{\alpha}}, \tag{6.25}$$

Proof. We employ induction to show the conclusion. At first when $L=2$, we have, $l_L = l_2 = d_1$. By (6.23) it follows that

$$y_2^m = \frac{\sum\limits_{i_1,p_1,\dots,p_{d_2}=-m}^{m} \left[H_{p_1\dots p_{d_2}}\left(\mathbf{x}_{d_1}^{d_2}\right)^{\alpha} \widetilde{B}_{i_1}(y_1^m)^{\alpha}\right] Z(2,y_1^m)}{\sum\limits_{i_1,p_1,\dots,p_{d_2}=-m}^{m} \left[H_{p_1\dots p_{d_2}}\left(\mathbf{x}_{d_1}^{d_2}\right)^{\alpha} \widetilde{B}_{i_1}(y_1^m)^{\alpha}\right]}. \tag{6.26}$$

Using (6.23) (6.26) we can conclude that

$$Z(2,y_1^m) = \frac{\sum\limits_{q_1,\dots,q_{d_1}=-m}^{m} H_{q_1\dots q_{d_1}}\left(\mathbf{x}_0^{d_1}\right)^{\alpha}\left(b_{0;q_1\dots q_{d_1}}^1 c_{i_1}^2 + b_{0;p_1\dots p_{d_2}}^2\right)}{\sum\limits_{q_1,\dots,q_{d_1}=-m}^{m} H_{q_1\dots q_1}\left(\mathbf{x}_0^{d_1}\right)^{\alpha}}$$
$$+\frac{\sum\limits_{q_1,\dots,q_{d_1}=-m}^{m} H_{q_1\dots q_{d_1}}\left(\mathbf{x}_0^{d_1}\right)^{\alpha}\left(\sum\limits_{i=1}^{d_1} c_{i_1}^2 b_{i;q_1\dots q_{d_1}}^1 x_i + \sum\limits_{i=1}^{d_2} b_{i;p_1\dots p_{d_2}}^2 x_{d_1+i}\right)}{\sum\limits_{q_1,\dots,q_{d_1}=-m}^{m} H_{q_1\dots q_1}\left(\mathbf{x}_0^{d_1}\right)^{\alpha}}. \tag{6.27}$$

Substituting (6.27) for the corresponding term of (6.26) and using Lemma 6.3 we get

$$y_2^m = \frac{\sum\limits_{i_1,p_1,\dots,p_{d_1+d_2}=-m}^{m} \left(H_{p_1\dots p_{d_1+d_2}}\left(\mathbf{x}_0^{d_1+d_2}\right)\widetilde{B}_{i_1}(y_1^m)\right)^{\alpha} C\left(\mathbf{x}_0^{d_1+d_2}\right)}{\sum\limits_{i_1,p_1,\dots,p_{d_1+d_2}=-m}^{m} \left(H_{p_1\dots p_{d_1+d_2}}\left(\mathbf{x}_0^{d_1+d_2}\right)\widetilde{B}_{i_1}(y_1^m)\right)^{\alpha}},$$

where $C\left(\mathbf{x}_0^{d_1+d_2}\right) = C(x_1,\dots,x_{d_1+d_2})$ is a linear of $x_1,\dots,x_{d_1+d_2}$:

$$C\left(\mathbf{x}_0^{d_1+d_2}\right) = \sum_{i=1}^{d_1} c_{i_1}^2 b_{i;p_1\dots p_{d_1}}^1 x_i + \sum_{i=1}^{d_2} b_{i;p_{d_1+1}\dots p_{d_1+d_2}}^2 x_{d_1+i} + b_{0;p_1\dots p_{d_1}}^1 c_{i_1}^2 + b_{0;p_{d_1+1}\dots p_{d_1+d_2}}^2.$$

If denote

$$a_0(I^2,\,P^2) = b_{0;p_1\dots p_{d_1}}^1 c_{i_1}^2 + b_{0;p_{d_1+1}\dots p_{d_1+d_2}}^2;$$

$$a_i(I^2,\,P^2)=\begin{cases} b^1_{i;p_1\dots p_{d_1}}c^2_{i_1}, & 1\le i\le d_1,\\ b^2_{i;p_{d_1+1}\dots p_{d_1+d_2}}, & d_1+1\le i\le d_1+d_2.\end{cases}$$

we can conclude that the following fact holds:

$$y_2^m=\frac{\sum\limits_{i_1;p_1,\dots,p_{l_2+d_2}=-m}^{m}\left(H_{p_1\dots p_{d_2+l_2}}\left(\mathbf{x}_0^{d_2+l_2}\right)J(I^2;y_1^m)\right)^{\alpha}O\left(\mathbf{x}_0^{l_2+d_2}\right)}{\sum\limits_{i_1;p_1,\dots,p_{d_2+l_2}=-m}^{m}\left(H_{p_1\dots p_{l_2+d_2}}\left(\mathbf{x}_0^{d_2+l_2}\right)J(I^2;y_1^m)\right)^{\alpha}},\tag{6.28}$$

where $O\left(\mathbf{x}_0^{l_2+d_2}\right)=O(x_1,\dots,x_{l_2+d_2})=a_0(I^2,\,P^2)+\sum\limits_{i=1}^{d_2+l_2}a_i(I^2,\,P^2)x_i$. By (6.28) we imply, when $j=2$ (6.25) holds. Assume (6.25) holds when $j=L-1$. Since $l_{L-1}+d_{L-1}=l_L$ it follows that

$$y_{L-1}^m=\frac{\sum\limits_{i_1,\dots,i_{L-2};p_1,\dots,p_{l_L}=-m}^{m}\left[H_{p_1\dots p_{l_L}}\left(\mathbf{x}_0^{l_L}\right)J(I^{L-1};y_1^m,\dots,y_{L-2}^m)\right]^{\alpha}O\left(\mathbf{x}_0^{l_L}\right)}{\sum\limits_{i_1,\dots,i_{L-2};p_1,\dots,p_{l_L}=-m}^{m}\left[H_{p_1\dots p_{l_L}}\left(\mathbf{x}_0^{l_L}\right)J(I^{L-1};y_1^m,\dots,y_{L-2}^m)\right]^{\alpha}},\tag{6.29}$$

also we denote $E\left(\mathbf{x}_0^{l_L}\right)=a_0(I^{L-1},\,P^{L-1})+\sum\limits_{i=1}^{d_{L-1}+l_{L-1}}a_i(I^{L-1},\,P^{L-1})x_i$. By (6.23) we obtain

$$\begin{aligned} y_L^m &= T_L\left(\mathbf{x}_{l_L}^{d_L},\,y_{L-1}^m\right)\\ &=\frac{\sum\limits_{i_L,p_1,\dots,p_{d_L}=-m}^{m}\left[H_{p_1\dots p_{d_L}}\left(\mathbf{x}_{l_L}^{d_L}\right)\widetilde{B}_{i_L}(y_{L-1}^m)\right]^{\alpha}Z(L,y_{L-1}^m)}{\sum\limits_{i_L,p_1,\dots,p_{d_L}=-m}^{m}\left[H_{p_1\dots p_{d_L}}\left(x_{l_L}^{d_L}\right)\widetilde{B}_{i_L}(y_{L-1}^m)\right]^{\alpha}}.\end{aligned}\tag{6.30}$$

Similarly with (6.25) (6.26), substituting (6.29) for corresponding term in (6.30) and letting $O\left(\mathbf{x}_0^d\right)=O(\mathbf{x})=a_0(I^L,\,P^L)+\sum\limits_{i=1}^{d}a_i(I^L,\,P^L)x_i$, where

$$a_0(I^L,P^L)=a_0(I^{L-1},\,P^{L-1})\cdot c^L_{i_L}+b^L_{0;p_{l_L+1}\dots p_{l_L+d_L}};$$

$$a_i(I^L,\,P^L)=\begin{cases} a_i(I^{L-1},P^{L-1})\cdot c^L_{l_L}, & 1\le i\le l_L,\\ b^L_{i;p_{l_L+1}\dots p_{l_L+d_L}}, & l_L+1\le i\le l_L+d_L,\end{cases}$$

and considering $l_L+d_L=d$, we can express the output y_L^m as follows:

$$y_L^m=\frac{\sum\limits_{i_1,\dots,i_{L-1};p_1,\dots,p_d=-m}^{m}\left[H_{p_1\dots p_d}(\mathbf{x})J(I^L;y_1^m,\dots,y_{L-1}^m)\right]^{\alpha}O(\mathbf{x})}{\sum\limits_{i_1,\dots,i_{L-1};p_1,\dots,p_d=-m}^{m}\left[H_{p_1\dots p_d}(\mathbf{x})J(I^L;y_1^m,\dots,y_{L-1}^m)\right]^{\alpha}}.$$

Thus, when $j = L$ (6.25) holds. By the induction principle the theorem is proved. □

As an inverse process of the proof of Theorem 6.6, the following conclusion holds also.

Theorem 6.7 *Suppose* $y_1^m, ..., y_{L-1}^m$ *is intermediate input variables. The I/O relationship* $(x_1, ..., x_d) \longrightarrow y_L^m$ *of the generalized T–S fuzzy system is given as follows:*

$$y_L^m = \frac{\sum\limits_{i_1,...,i_{L-1};p_1,...,p_d=-m}^{m} \big[H_{p_1...p_d}(\mathbf{x}) J(I^L; y_1^m, ..., y_{L-1}^m)\big]^\alpha a_0(I^L,\, P^L)}{\sum\limits_{i_1,...,i_{L-1};p_1,...,p_d=-m}^{m} \big[H_{p_1...p_d}(\mathbf{x}) \cdot J(I^L; y_1^m, ..., y_{L-1}^m)\big]^\alpha}$$

$$+\frac{\sum\limits_{i_1,...,i_{L-1};p_1,...,p_d=-m}^{m} \big[H_{p_1...p_d}(\mathbf{x}) J(I^L; y_1^m, ..., y_{L-1}^m)\big]^\alpha \Big(\sum\limits_{i=1}^{d} a_i(I^L, P^L) x_i\Big)}{\sum\limits_{i_1,...,i_{L-1};p_1,...,p_d=-m}^{m} \big[H_{p_1...p_d}(\mathbf{x}) J(I^L; y_1^m, ..., y_{L-1}^m)\big]^\alpha}.$$

Then there exist $b^j_{0;p_1...p_j}$, c^j_q *and* $b^j_{i;p_1...p_j}$ $(j = 1, ..., L;\ p_1, ..., p_j = -m, -m+1, ..., m-1,\ m)$, *so that*

$$\begin{cases} y_1^m = T_1\big(\mathbf{x}_0^{d_1}\big) = \dfrac{\sum\limits_{p_1,...,p_{d_1}=-m}^{m} H_{p_1...p_{d_1}}\big(x_0^{d_1}\big)^\alpha \Big(b^1_{0;p_1...p_{d_1}} + \sum\limits_{i=1}^{d_1} b^1_{i;p_1...p_{d_1}} x_i\Big)}{\sum\limits_{p_1,...,p_{d_1}=-m}^{m} H_{p_1...p_{d_1}}(x_1, ..., x_{d_1})^\alpha} \\ y_j^m = T_j\big(x_{l_j}^{d_j}, y_{j-1}^m\big) = \dfrac{\sum\limits_{q,p_1,...,p_{d_j}=-m}^{m} \big[H_{p_1...p_{d_j}}\big(\mathbf{x}_{l_j}^{d_j}\big)\widetilde{B}_q(y_{j-1}^m)\big]^\alpha Z(j,\, y_{j-1}^m)}{\sum\limits_{q,p_1,...,p_{d_j}=-m}^{m} \big[H_{p_1...p_{d_j}}\big(\mathbf{x}_{l_j}^{d_j}\big)\widetilde{B}_q(y_{j-1}^m)\big]^\alpha}, \end{cases}$$

where $Z(j,\, y_{j-1}^m)$ is also defined as (6.24):

$$Z(j,\, y_{j-1}^m) = b^j_{0;p_1...p_{d_j}} + c^j_q y^m_{j-1} + \sum_{i=1}^{d_j} b^j_{i;p_1...p_{d_j}} x_{l_j+i}.$$

In application we can simplify the HFS (6.23), and obtain a simple HFS, i.e. for $j = 2, ..., L$, let

$$b^1_{i;p_1...p_{d_1}} = 0\ (i = 1, ..., d_1);\ b^j_{i;p_1...p_{d_j}} = 0\ (i = 1, ..., d_j).$$

Then HFS (6.23) is a simple HFS as

$$\begin{cases} y_1^m = T_1^m(\mathbf{x}_0^{d_1}) = \dfrac{\sum\limits_{p_1,\dots,p_{d_1}=-m}^{m} H_{p_1\dots p_{d_1}}(x_0^{d_1})^{\alpha} \cdot b_{0;p_1\dots p_{d_1}}^1}{\sum\limits_{p_1,\dots,p_{d_1}=-m}^{m} H_{p_1\dots p_{d_1}}(\mathbf{x}_0^{d_1})^{\alpha}}; \\ y_j^m = T_j^m(\mathbf{x}_{l_j}^{d_j}, y_{j-1}^m) \\ \quad = \dfrac{\sum\limits_{i_{j-1},p_{l_j+1},\dots,p_{l_j+d_j}=-m}^{m} \left[H_{p_1\dots p_{d_j}}(\mathbf{x}_{l_j}^{d_j})\widetilde{B}_{(j-1)i_{j-1}}(y_{j-1}^m)\right]^{\alpha} Q(y_{j-1}^m)}{\sum\limits_{i_{j-1},p_{l_j+1},\dots,p_{l_j+d_j}=-m}^{m} \left[H_{p_{l_j+1}\dots p_{l_j+d_j}}(\mathbf{x}_{l_j}^{d_j})\, \widetilde{B}_{(j-1)i_{j-1}}(y_{j-1}^m)\right]^{\alpha}}, \end{cases} \tag{6.31}$$

where l_j, $Q(y_{j-1}^m)$ are determined as follows for $j = 2, \dots, L$:

$$l_j = \sum_{k=1}^{j-1} d_k; \quad Q(y_{j-1}^m) = b_{0;p_{l_j+1}\dots p_{l_j+d_j}}^j + c_{i_{j-1}}^j y_{j-1}^m.$$

(6.31) is called a simple generalized hierarchical T–S fuzzy system. As a special case of Theorem 6.6 and Theorem 6.7 we have

Corollary 6.6 *Let $y_1^m, \dots, y_L^m$ be the simple generalized hierarchical T–S fuzzy system defined by (6.31). Then for any $j = 2, \dots, L$, and $I^j = (i_1, \dots, i_{j-1})$, $P^j = (p_1, \dots, p_{l_j+d_j}) : i_1, \dots, i_{j-1}; p_1, \dots, p_{l_j+d_j} \in \{-m, -m+1, \dots, m-1, m\}$, there is a constant $a_0(I^j, P^j)$, so that*

$$y_j^m = \frac{\sum\limits_{i_1,\dots,i_{j-1};p_1,\dots,p_{l_j+d_j}=-m}^{m} \left[H_{p_1\dots p_{d_j+l_j}}(\mathbf{x}_0^{d_j+l_j}) J(I^j; y_1^m, \dots, y_{j-1}^m)\right]^{\alpha} a_0(I^j, P^j)}{\sum\limits_{i_1,\dots,i_{j-1};p_1,\dots,p_{d_j+l_j}=-m}^{m} \left[H_{p_1\dots p_{d_j+l_j}}(\mathbf{x}_0^{d_j+l_j}) J(I^j; y_1^m, \dots, y_{j-1}^m)\right]^{\alpha}}. \tag{6.32}$$

(6.32) is a simple generalized T–S fuzzy system. By Corollary 6.6, the output of a generalized hierarchical T–S fuzzy system defined as (6.31) can be expressed as one of a simple generalized T–S system. The adjustable parameters $a_0(I^2, P^2), \dots, a_0(I^L, P^L)$ can be determined by the following iteration laws:

$$\begin{cases} a_0(I^2,\, P^2) = b_{0;p_1\dots p_{d_1}}^1 c_{i_1}^2 + b_{0;p_{d_1+1}\dots p_{d_1+d_2}}^2; \\ a_0(I^j,\, P^j) = a_0(I^{j-1}, P^{j-1}) c_{i_{j-1}}^j + b_{0;p_{l_j+1}\dots p_{l_j+d_j}}^j, \end{cases} \tag{6.33}$$

where $j = 3, \dots, L$. On the other hand, by (6.33), if $a_0(I^2, P^2), \dots, a_0(I^L, P^L)$ are known, then choose $b_{0;p_1\dots p_{d_1}}^1$, $c_{i_{j-1}}^j, \dots, b_{0;p_1\dots p_{d_j}}^j$ ($j = 2, \dots, L$), so that (6.33) holds, that is, a known simple generalized T–S fuzzy system can determine a corresponding generalized hierarchical T–S fuzzy system.

§6.4 Approximation of hierarchical T–S fuzzy system

By Corollary 6.6 a simple generalized T–S fuzzy system and its simple hierarchical fuzzy system are equivalent, so it seems to be an obvious fact that hierarchical T–S fuzzy systems can be universal approximators. However, how can we construct such an approximating system? For a given accuracy $\varepsilon > 0$, how is the corresponding approximating procedure realized? How can we estimate the size of the fuzzy rule base of the T–S fuzzy system related? and so on. These problems are of much significance in theory and application of fuzzy systems. In the section we employ SPLF's to present systematic research to these subjects.

6.4.1 Universal approximation with maximum norm

Considering that universal approximation is studied on a given compact set $U \subset \mathbb{R}^d$, by Remark 6.1 we may assume $U = [-1,\ 1]^d$. Also in the following we let fuzzy set family $\{\widetilde{A}_{ij}\ |i = 1, ..., d;\ j = -m,\ -m+1, ..., m-1,\ m\}$ satisfy S–L condition, moreover

$$\{\widetilde{A}_{ij}\ |i = 1, ..., d;\ j = -m,\ -m+1, ..., m-1,\ m\} \subset \widetilde{\mathcal{O}}_0(1, m).$$

Theorem 6.8 *Let $S \in \mathcal{D}_d^0$. Then for any $\varepsilon > 0$, there exist $m \in \mathbb{N}$ and a simple generalized hierarchical T–S fuzzy system $\{y_1^m, ..., y_L^m\}$, determined by (6.31), so that $\|y_L^m - S\|_{\infty,[-1,1]^d} < \varepsilon$.*

Proof. Suppose $S \in \mathcal{D}_d^0$ is defined by (6.12), where $a = 1$. Next let us prove, there are $m \in \mathbb{N}$, and the coefficient $a_0(I^L,\ P^L)$, where

$$I^L = (i_1, ..., i_{L-1}),\ p^L = (p_1, ..., p_{l_L+d_L}) = (p_1, ..., p_d):$$
$$i_1, ..., i_{L-1};\ p_1, ..., p_d = 0,\ \pm 1, ..., \pm m,$$

so that if y_L^m is determined by (6.31) then $\|y_L^m - S\|_{\infty,[-1,1]^d} < \varepsilon$.

Define $a_0(I^L,\ P^L)$ $(i_1, ..., i_{L-1};\ p_1, ..., p_{l_L+d_L} \in \{-m,\ -m+1, ..., m-1,\ m\})$ as follows:

$$a_0(I^L,\ P^L) = S\Big(\frac{p_1}{m}, ..., \frac{p_d}{m}\Big). \tag{6.34}$$

By Lemma 6.2, for $\mathbf{x} = (x_1, ..., x_d) \in [-1,\ 1]^d$, and for any $(p_1, ..., p_d) \in N(\mathbf{x}) = N(x_1, ..., x_d)$, let

$$\frac{p_i}{m} = x_i + \frac{\theta_{p_i}}{m}\ (i = 1, ..., d).$$

Then $|\theta_{p_i}| \leq c_0$. Thus

$$(p_1, ..., p_d) \in N(\mathbf{x}) \Longrightarrow \Big|S(\mathbf{x}) - S\Big(\frac{p_1}{m}, ..., \frac{p_d}{m}\Big)\Big| \leq \frac{c_0}{m} \cdot \sum_{i=1}^{d} D_i(S). \tag{6.35}$$

So considering $l_L + d_L = d$, and denoting

$$H_{p_1 \cdots p_d}(x_1, ..., x_d) \cdot J(I^L; y_1^m, ..., y_{L-1}^m) \triangleq \Phi(I^L, P^L),$$

and by (6.34) (6.35) we can conclude that

$$\begin{aligned}
\|y_L^m - S\|_{\infty,[-1,1]^d} &= \bigvee_{\mathbf{x}\in[-1,1]^d} \left\{ \left| y_L^m - S(\mathbf{x}) \right| \right\} \\
&= \bigvee_{\mathbf{x}\in[-1,1]^d} \left\{ \left| \frac{\sum\limits_{i_1,...,i_{L-1};p_1,...,p_d=-m}^{m} \Phi(I^L, P^L)^\alpha \cdot a_0(I^L, P^L)}{\sum\limits_{i_1,...,i_{L-1};p_1,...,p_d=-m}^{m} \Phi(I^L, P^L)^\alpha} - S(\mathbf{x}) \right| \right\} \\
&= \bigvee_{\mathbf{x}\in[-1,1]^d} \left\{ \left| \frac{\sum\limits_{i_1,...,i_{L-1};p_1,...,p_d=-m}^{m} \Phi(I^L, P^L)^\alpha \Big(S\big(\frac{p_1}{m}, ..., \frac{p_d}{m}\big) - S(\mathbf{x}) \Big)}{\sum\limits_{i_1,...,i_{L-1};p_1,...,p_d=-m}^{m} \Phi(I^L, P^L)^\alpha} \right| \right\} \\
&\le \bigvee_{\mathbf{x}\in[-1,1]^d} \left\{ \left| \frac{\sum\limits_{i_1,...,i_{L-1}=-m}^{m} \sum\limits_{(p_1,...,p_d)\in N(\mathbf{x})} \Phi(I^L, P^L)^\alpha \left| S\big(\frac{p_1}{m}, ..., \frac{p_d}{m}\big) - S(\mathbf{x}) \right|}{\sum\limits_{i_1,...,i_{L-1}=-m}^{m} \sum\limits_{(p_1,...,p_d)\in N(\mathbf{x})} \Phi(I^L, P^L)^\alpha} \right| \right\} \\
&\le \bigvee_{\mathbf{x}\in[-1,1]^d} \left\{ \frac{c_0}{m} \cdot \sum_{i=1}^{d} D_i(S) \right\} = \frac{c_0}{m} \cdot \sum_{i=1}^{d} D_i(S).
\end{aligned} \tag{6.36}$$

If let $m > c_0 \cdot \sum\limits_{i=1}^{d} D_i(S)/\varepsilon$, we have, $\left\| y_L^m - S \right\|_{\infty,[-1,1]^d} < \varepsilon$. By (6.33), for $2 \le k \le L$, it follows that

$$a_0\big(I^k, P^k\big) = a_0\big(I^{k-1}, P^{k-1}\big) \cdot c_{i_{k-1}}^k + a_{0;p_{l_k+1} \cdots p_{l_k+d_k}}^k. \tag{6.37}$$

By (6.34) (6.37) we employ the backward induction to establish the parameters: $b_{0;p_1 \cdots p_{d_k}}^k$, $c_{i_{k-1}}^k$ $(k = 2, ..., L)$, $b_{0;p_1 \cdots p_{d_1}}^1$. A simple generalized hierarchical T–S fuzzy system as (6.31) can be established. The theorem is therefore proved. □

By Corollary 6.6, a simple generalized hierarchical T–S fuzzy system defined as (6.31) can be expressed as a generalized T–S fuzzy system, so for any $\varepsilon > 0$, there exist $m \in \mathbb{N}$, and the parameter $b_{0;p_1 \cdots p_d}$ $(p_1, ..., p_d = 0, \pm 1, ..., \pm m)$:

$$b_{0;p_1 \cdots p_d} = S\Big(\frac{p_1}{m}, ..., \frac{p_d}{m}\Big).$$

Thus, the adjustable parameters $b_{0;p_1 \cdots p_{d_1}}^1$, $b_{0;p_1 \cdots p_{d_k}}^k$, $c_{i_{k-1}}^k$ satisfy the following backward iteration formulas, where $i \in \{1, ..., d\}$; $k \in \{2, ..., L\}$ and

i_{k-1}; $p_1, ..., p_d \in \{-m, \ -m+1, ..., m-1, \ m\}$:

$$\begin{cases} a_0(I^L, P^L) = b_{0;p_1\cdots p_d} = S\Big(\frac{p_1}{m}, ..., \frac{p_d}{m}\Big); \\ a_0(I^k, P^k) = a_0(I^{k-1}, P^{k-1}) \cdot c^k_{i_{k-1}} + b^k_{0;p_{l_k+1}\cdots p_{l_k+d_k}} \ (k = L-1, ..., 3); \\ a_0(I^2, P^2) = b^1_{0;p_1\cdots p_{d_1}} c^2_{i_1} + b^2_{0;p_{d_1+1}\cdots p_{d_1+d_2}}. \end{cases} \tag{6.38}$$

Using Theorem 6.2 and Theorem 6.8 we can conclude that

Theorem 6.9 *Let* $f : \ [-1, \ 1]^d \longrightarrow \mathbb{R}$ *be a continuous function. Then for any* $\varepsilon > 0$, *there is* $m \in \mathbb{N}$, *if* $y_1^m, ..., y_L^m$ *are defined by (6.31), it follows that* $\|y_L^m - f\|_{\infty,[-1,1]^d} < \varepsilon$.

By Theorem 6.9 simple generalized hierarchical T–S fuzzy systems can be universal approximators. So such hierarchical systems can be applied, efficiently in designing fuzzy controller and system modeling and so on. For a given error bound $\varepsilon > 0$, we can estimate m, consequently the size of the rule base of a HFS can be estimated.

Theorem 6.10 *Let* $f : \ [-1, \ 1]^d \longrightarrow \mathbb{R}$ *be a continuous function. Then for any* $\varepsilon > 0$, *there is* $h > 0$. *Suppose* $D_H(f)$ *is determined as (6.16), where* $a = 1$. *We have, when* $m > 2D_H(f)c_0d/\varepsilon$, *it follows that* $\|f - y_L^m\|_{\infty,[-1,1]^d} < \varepsilon$.

Proof. By Remark 6.2, for $\varepsilon > 0$, if partitioning $[-1, \ 1]^d$ identically into small cubes, and then dividing these cubes respectively into d dimensional polyhedrons $\Delta_1, ..., \Delta_N$ ($N \in \mathbb{N}$). Thus, we can establish a SPLF S, so that $\|f - S\|_{\infty,[-1,1]^d} < \varepsilon/2$. Let the side length of a cube be $h > 0$, easily we get, $D_H(f) = \bigvee_{i=1}^{d} \{D_i(S)\}$. By Theorem 6.8, if $m > 2D_H(f)c_0d/\varepsilon$, it follows that $m > 2c_0 \cdot \sum_{i=1}^{d} D_i(S)/\varepsilon$, so $\|y_L^m - S\|_{\infty,[-1,1]^d} < \varepsilon/2$. Therefore

$$\|f - y_L^m\|_{\infty,[-1,1]^d} \le \|f - S\|_{\infty,[-1,1]^d} + \|S - y_L^m\|_{\infty,[-1,1]^d} < \frac{\varepsilon}{2} + \frac{\varepsilon}{2} = \varepsilon.$$

The theorem is therefore proved. □

If let $c_0 = 2$, $d = 3$, then by $m > 12D_H(f)/\varepsilon$ we have, $\|f - y_L^m\|_{\infty,[-1,1]^d} < \varepsilon$. Giving a continuous function f and an error bound $\varepsilon > 0$, by the following steps we can realize the approximation of f by a simple generalized hierarchical T–S fuzzy system defined as (6.31):

Step 1. By (6.16) we calculate the supremum $D_H(f)$ and the minimum m;

Step 2. For a given real function f, we establish the antecedent fuzzy sets $\widetilde{A}_{ij} \in \widetilde{\mathcal{O}}(1, \ m)$ ($i = 1, ..., d$; $j = 0, \pm 1, ..., \pm m$), and $\widetilde{B}_{ki_{k-1}}$ ($k = 2, ..., L$; $i_{k-1} = 0, \pm 1, ..., \pm m$). Usually the fuzzy set $\widetilde{A}_{ij}$ is a triangular or trapezoidal fuzzy number;

Step 3. Using (6.34) (6.37) we establish the parameter $a_0(I^L,\ P^L)$ as follows:

$$a_0(I^L,\ P^L)=f\Big(\frac{p_1}{m},...,\frac{p_d}{m}\Big);$$

Step 4. By (6.38) the backward induction is employed to establish the parameters $b^1_{0;p_1\cdots p_{d_1}}$, $b^k_{0;p_{l_k+1}\cdots p_{l_k+d_k}}$ and $c^k_{i_{k-1}}$ related to the HFS;

Step 5. Using (6.31) we obtain the simple generalized hierarchical fuzzy system $\{y_1^m,...,y_L^m\}$.

6.4.2 Realizing procedure of universal approximation

In the subsection we demonstrate a realizing procedure of universal approximation of generalized hierarchical T–S fuzzy system by a real example. let $\alpha=1$, $d=3$, $c_0=2$ and $d_1=2$, $d_2=1$. Thus, the fuzzy system related has three input variables, and there exist two levels in the HFS related. Define a continuous function $f:\ [-1,\ 1]^3\longrightarrow\mathbb{R}$ as follows:

$$f(x_1,\ x_2,\ x_3)=\exp\Big\{-\frac{x_1^2+x_2^2+x_3^2}{25}\Big\}\ \ (|x_1|\le 1,\ |x_2|\le 1,\ |x_3|\le 1).$$

Give an error bound $\varepsilon=0.1$. Since

$$\Big|\exp\Big\{-\frac{x_1^2+y_1^2+z_1^2}{25}\Big\}-\exp\Big\{-\frac{x_2^2+y_2^2+z_2^2}{25}\Big\}\Big|\le\frac{2}{25}(|x_1-x_2|+|y_1-y_2|+|z_1-z_2|),\tag{6.39}$$

and using (6.16) (6.39) we get, $D_H(f)\le 2/25<1/12$. So let $m=12/(12\varepsilon)=1/0.1=10$. Partition $[-1,\ 1]$ identically into 20 parts. By Theorem 6.9 it follows that

$$\|y_L^m-f\|_{\infty,[-1,1]^d}=\|y_2^{10}-f\|_{\infty,[-1,1]^3}<\varepsilon.$$

Using Proposition 6.1, we get the size of the rule base of the HFS related is $(2m+1)^2(d-1)$, i.e. $2(2\times 10+1)^2=882$.

Define the triangular fuzzy number $\widetilde{A}$ as follows:

$$\widetilde{A}(x)=\begin{cases}10\Big(\frac{1}{10}-x\Big), & 0\le x\le\frac{1}{10},\\ 10\Big(\frac{1}{10}+x\Big), & -\frac{1}{10}\le x<0,\\ 0; & \text{otherwise.}\end{cases}$$

Let $\widetilde{A}_{1j}=\widetilde{A}_{2j}=\widetilde{A}_{3j}$ $(j=0,\pm1,...,\pm10)$, and $\widetilde{A}_{1j}$ $(j=0,\pm1,...,\pm9)$ be defined through the translation of $\widetilde{A}$, that is, $\widetilde{A}_{1j}(x)=\widetilde{A}(x-j/10)$, and

$$\widetilde{A}_{1(10)}(x)=\begin{cases}10\Big(x-\frac{9}{10}\Big), & \frac{9}{10}\le x\le 1,\\ 0, & \text{otherwise;}\end{cases}$$

$$\widetilde{A}_{1(-10)}(x)=\begin{cases}-10\Big(x+\dfrac{9}{10}\Big), & -1\le x\le -\dfrac{9}{10},\\ 0, & \text{otherwise.}\end{cases}$$

Define $\widetilde{B}_{1j}$ $(j=0,\ \pm1,...,\pm10)$ as follows:

$$\widetilde{B}_{1j}(y)=\exp\Big\{-\frac{1}{2}\Big(y-\frac{j}{10}\Big)\Big\}.$$

Since $H_{p_1p_2p_3}(x_1,\ x_2,\ x_3)=\widetilde{A}_{1p_1}(x_1)\cdot\ \widetilde{A}_{1p_2}(x_2)\cdot\ \widetilde{A}_{1p_3}(x_3)$, we get the following fact:

$$J(I^2,\ y_1^m)=J(i_1,\ y_1^m)=\widetilde{B}_{1i_1}(y_1^m)=\exp\Big\{-\frac{y_1^m-\frac{i_1}{10}}{2}\Big\},$$

So by (6.31) we can establish a two level simple generalized hierarchical T–S fuzzy system:

$$\begin{cases} y_1^m=\dfrac{\sum\limits_{p_1,p_2=-10}^{10}\widetilde{A}_{1p_1}(x_1)\cdot\ \widetilde{A}_{2p_2}(x_2)\cdot b^1_{0;p_1p_2}}{\sum\limits_{p_1,p_2=-10}^{10}\widetilde{A}_{1p_1}(x_1)\cdot\ \widetilde{A}_{2p_2}(x_2)}; \\ y_2^m=\dfrac{\sum\limits_{i_1;p_3=-10}^{10}\widetilde{A}_{3p_3}(x_3)\cdot\ \widetilde{B}_{1i_1}(y_1^m)\cdot\big(b^2_{0;p_3}+c^2_{i_1}y_1^m\big)}{\sum\limits_{i_1;p_3=-10}^{10}\widetilde{A}_{3p_3}(x_3)\cdot\ \widetilde{B}_{1i_1}(y_1^m)}. \end{cases} \tag{6.40}$$

Table 6.1 Approximating errors at sample points chosen randomly

No.	sample points	$f(x_1,x_2,x_3)$	y_L^m	error
1	$(-0.9,-0.9,-0.9)$	0.9073745131	0.9073745132	1.0×10^{-10}
2	$(-0.7,-0.6,-0.5)$	0.9569539575	0.9569539577	2.0×10^{-10}
3	$(-0.4,-0.3,-0.5)$	0.9801986733	0.9801986734	1.0×10^{-10}
4	$(0.2,0.3,-0.1)$	0.9944156508	0.9944156509	1.0×10^{-10}
5	$(0.4,-0.5,0.6)$	0.9696694876	0.9696694870	6.0×10^{-10}
6	$(0,0.4,0.2)$	0.9920319148	0.9920319151	3.0×10^{-10}
7	$(-0.5,0.2,0.2)$	0.9868867379	0.9868867390	11.0×10^{-10}
8	$(-0.7,0.6,-0.8)$	0.9421413147	0.9421413141	6.0×10^{-10}
9	$(0.6,0.9,0.8)$	0.9301587579	0.9301587580	1.0×10^{-10}
10	(0.9,0.9,0.9)	0.9073745131	0.9073745132	1.0×10^{-10}

By (6.38) and the fact: $f(p_1/10,\, p_2/10,\, p_3/10) = S(p_1/10,\, p_2/10,\, p_3/10)$, we define the parameters $b^1_{0;p_1p_2}$, $c^2_{i_1}$ and $b^2_{0;p_3}$ as (6.40):

$$b^1_{0;p_1p_2}\cdot c^2_{i_1} + b^2_{0;p_3} = f\left(\frac{p_1}{10},\, \frac{p_2}{10},\, \frac{p_3}{10}\right).$$

After choosing

$$c^2_{i_1} = 1,\ b^1_{0;p_1p_2} + b^2_{0;p_3} = f\left(\frac{p_1}{10},\, \frac{p_2}{10},\, \frac{p_3}{10}\right), \tag{6.41}$$

we get a HFS as (6.31). By Theorem 6.6 the I/O relationship related can be expressed as

$$y_2^m = \frac{\sum\limits_{i_1;p_1,p_2,p_3=-10}^{10} \left([\widetilde{A}_{1p_1}(x_1)\cdot \widetilde{A}_{2p_2}(x_2)\cdot \widetilde{A}_{3p_3}(x_3)\cdot \widetilde{B}_{1i_1}(y_1^m)]\cdot f\left(\frac{p_1}{10}, \frac{p_2}{10}, \frac{p_3}{10}\right)\right)}{\sum\limits_{i_1;p_1,p_2,p_3=-10}^{10} \left[\widetilde{A}_{1p_1}(x_1)\cdot \widetilde{A}_{2p_2}(x_2)\cdot \widetilde{A}_{3p_3}(x_3)\cdot \widetilde{B}_{1i_1}(y_1^m)\right]}. \tag{6.42}$$

Table 6.1 demonstrates some approximating errors at some points chosen randomly when use y_2^m to approximate $f(x_1, x_2, x_3)$. From Table 6.1 we can see the high accuracy when HFS's approximating a given continuous function.

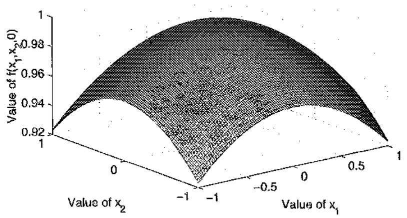


Figure 6.12 Surface of f when $x_3 = 0$

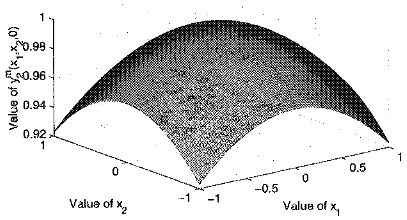


Figure 6.13 Surface of y_2^m when $x_3 = 0$

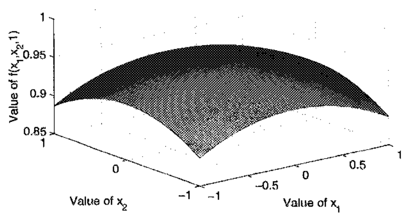


Figure 6.14 Surface of f when $x_3 = 1$

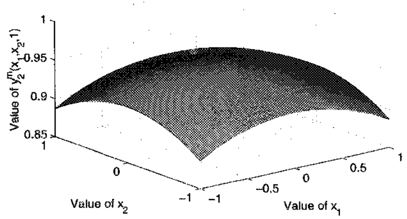


Figure 6.15 Surface of y_2^m when $x_3 = 1$

Let $x_3 = 0$ and $x_3 = 1$, respectively. we can obtain the respective section surfaces of f and y_2^m correspondingly, shown in Figure 6.12, Figure 6.13, Figure 6.14 and Figure 6.15. From table 6.1 and Figures 6.12–6.15 we know, with the approximation sense '$\approx_\varepsilon$' a generalized hierarchical T–S fuzzy system can represent a given continuous function. The realizing procedure is simple and succinct. If we use fuzzy systems without hierarchy to deal with the approximation, the size of the rule base related is $(2m+1)^3 = 21^3 = 9621$, which is much larger than one of a HFS.

6.4.3 Universal approximation with integral norm

By Theorem 6.4 and Corollary 6.6 with integral norm simple generalized hierarchical T–S fuzzy systems can be universal approximators. In the following let us show the conclusion and demonstrate the approximating procedure related.

Theorem 6.11 *Suppose μ is Lebesgue measure on $\mathbb{R}^d$. For any $f \in L_p(\mu)$, and $\forall\, \varepsilon > 0$, let $D_H(f)$ be established by (6.16) for $a > 0$ and $h > 0$. Then*

(i) There is $m \in \mathbb{N}$, so that if $y_1^m, ..., y_L^m$ are defined as (6.31), we have, $\|y_L^m - f\|_{\mu,p} < \varepsilon$, that is, simple generalized hierarchical T–S fuzzy systems are universal approximators with $L_p(\mu)-$norm;

(ii) If $h > 0$ is sufficiently small, and $m > (2a)^{1+d/p} \cdot d \cdot D_H(f) \cdot c_0/\varepsilon$, it follows that $\|y_L^m - f\|_{\mu,p} < \varepsilon$.

Proof. By Theorem 6.1, there exist $a > 0$, and $m_0 \in \mathbb{N}$, so that $\forall m > m_0$, we have, $S \in \mathcal{D}_d$, $\mathrm{Supp}(S) \subset [-a,\, a]^d$. Moreover

$$\int_{\|\mathbf{x}\| \geq m-1} |f(\mathbf{x})|^p \mathrm{d}\mu < \frac{\varepsilon^p}{4};\ \ \|f - S\|_{\mu,p} = \left\{ \int_{\mathbb{R}^d} |f(\mathbf{x}) - S(\mathbf{x})|^p \mathrm{d}\mu \right\}^{\frac{1}{p}} < \frac{\varepsilon}{2}.$$

By remark 6.2 we get, $S\big(ap_1/m, ..., ap_d/m\big) = f\big(ap_1/m, ..., ap_d/m\big)$. For $I^L = (i_1, ..., i_{L-1}) \in \{-m,\ -m+1, ..., m-1,\ m\}^{L-1}$, $P^L = (p_1, ..., p_d) \in \{-m,\ -m+1, ..., m-1,\ m\}^d$, let the constant $a_0(I^L,\ P^L)$ be defined as (6.33). Similarly with (6.36) we can show

$$\begin{aligned}
\|y_L^m - S\|_{\mu,p}^p &= \int_{\mathbb{R}^d} |y_L^m - S(\mathbf{x})|^p \mathrm{d}\mu \\
&\leq \int_{[-a,a]^d} \left| \frac{\sum\limits_{i_1,...,i_{L-1}=-m}^{m} \sum\limits_{(p_1,...,p_d)\in N(\mathbf{x})} \Phi(I^L,\ P^L)^\alpha \big|S(\frac{p_1}{m}, ..., \frac{p_d}{m}) - S(\mathbf{x})\big|}{\sum\limits_{i_1,...,i_{L-1}=-m}^{m} \sum\limits_{(p_1,...,p_d)\in N(\mathbf{x})} \Phi(I^L,\ P^L)^\alpha} \right|^p \mathrm{d}\mu \\
&\leq \Big[\frac{ac_0}{m} \cdot \sum_{i=1}^{d} D_i(S)\Big]^p \cdot (2a)^p \leq \Big[\frac{adc_0 D_H(f)}{m}\Big]^p \cdot (2a)^d.
\end{aligned}$$

let $m > \max\Big\{m_0, (2a)^{d/p+1} \cdot dc_0 D_H(f)/\varepsilon\Big\}$, and $h = a/m$. Then $\|y_L^m - S\|_{\mu,p} < \varepsilon/2$, Therefore

$$\|y_L^m - f\|_{\mu,p} \leq \|y_L^m - S\|_{\mu,p} + \|S - f\|_{\mu,p} < \frac{\varepsilon}{2} + \frac{\varepsilon}{2} = \varepsilon.$$

Thus (i) (ii) hold and the theorem is proved. □

Next let us use a real example to illustrate the realizing procedure of Theorem 6.11, that is, construct a simple generalized hierarchical T–S fuzzy system to approximate a given integrable function with $L_p(\mu)$−norm.

Example 6.4 For a HFS defined as (6.31) we let $\alpha = 1$, $d_1 = 1$, $d_2 = 2$, $c_0 = 2$. Thus a fuzzy system has three input variables x_1, x_2, x_3, and the corresponding HFS has two levels, $L = 2$. Let $f \in L_2(\mu)$ be defined as

$$\forall (x_1, x_2, x_3) \in \mathbb{R}^3,\ f(x_1, x_2, x_3) = \begin{cases} \exp\big(-2|x_1^3| - 2x_2^3 - 2|x_3|^3\big), & x_2 \geq 0, \\ -\exp\big(-2|x_1^3| + 2x_2^3 - 2|x_3|^3\big), & x_2 < 0. \end{cases}$$

Choose error bound $\varepsilon = 0.1$. With the following steps we establish the sample generalized hierarchical T–S system $\{y_1^m,\ y_2^m\}$, and realize the approximating procedure.

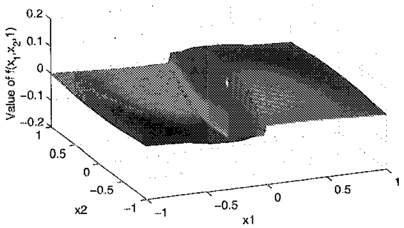


Figure 6.16 Surface of f when $x_3 = 1$

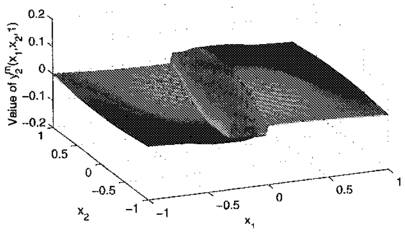


Figure 6.17 Surface of y_2^m when $x_3 = 1$

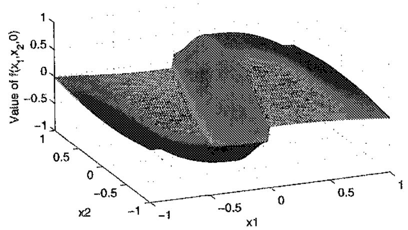


Figure 6.18 Surface of f when $x_3 = 0$

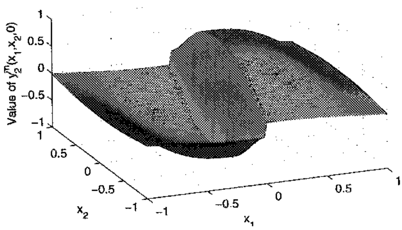


Figure 6.19 Surface of y_2^m when $x_3 = 0$

Step 1. Establish the interval $[-a,\ a]$, i.e. determine $a > 0$. Let $D \triangleq \{(x_1,\ x_2,\ x_3)\big|-1 \leq x_1,\ x_2,\ x_3 \leq 1\}$, then

$$\int_{\mathbb{R}^3 \setminus D} |f(x_1,\ x_2,\ x_3)| \mathrm{d}\mu < \left[2\int_1^{+\infty} \exp(-2x)\mathrm{d}x\right]^3 \mathrm{d}x < \frac{\varepsilon^2}{3}.$$

So choose $a = 1$. By the definition of f, $D_H(f) = 6$.

Step 2. Determine m. By Theorem 6.9 let $m > (2a)^{d/p+1} \cdot dc_0 D_H(f)/\varepsilon = 36\sqrt{32}$. So choose $m = 204$.

Step 3. Define the fuzzy set family $\{\tilde{A}_{ij} \,|i = 1, 2, 3;\ j = 0, \pm 1, ..., \pm m\} \subset \tilde{\mathcal{O}}_0(1, m)$.

$$\tilde{A}(x) = \begin{cases} 204\Big(\dfrac{1}{204} - x\Big), & 0 \le x \le \dfrac{1}{204}, \\ 204\Big(\dfrac{1}{204} + x\Big), & -\dfrac{1}{204} \le x < 0, \\ 0; & \text{otherwise}. \end{cases}$$

Let $\tilde{A}_{1j}=\tilde{A}_{2j}=\tilde{A}_{3j}$ $(j = 0, \pm 1, ..., \pm 204)$, and $\tilde{A}_{1j}$ $(j = 0, \pm 1, ..., \pm 204)$ be determined by the translation of $\tilde{A}$, i.e. $\tilde{A}_{1j}(x) = \tilde{A}\big(x - j/204\big)$. Moreover, define $\tilde{B}_{1j}$ $(j = 0, \pm 1, ..., \pm 204)$ as follows:

$$\tilde{B}_{1j}(y) = \exp\Big\{-\frac{1}{2}\Big(y - \frac{j}{204}\Big)\Big\}.$$

Step 4. Similarly with (6.40) (6.41) (6.42) we get the HFS related as follows:

$$y_2^m = \frac{\sum\limits_{i_1;p_1,p_2,p_3=-240}^{240} \big(\tilde{A}_{1p_1}(x_1)\cdot \tilde{A}_{2p_2}(x_2)\cdot \tilde{A}_{3p_3}(x_3)\cdot \tilde{B}_{1i_1}(y_1^m)\big) f\big(\frac{p_1}{240}, \frac{p_2}{240}, \frac{p_3}{240}\big)}{\sum\limits_{i_1;p_1,p_2,p_3=-240}^{240} \big(\tilde{A}_{1p_1}(x_1)\cdot \tilde{A}_{2p_2}(x_2)\cdot \tilde{A}_{3p_3}(x_3)\cdot \tilde{B}_{1i_1}(y_1^m)\big)}.$$

Choose $x_3 = 1$ and $x_3 = 0$. Figure 6.16 and Figure 6.17 illustrate respectively the section graphs of the function f and the HFS y_2^m when $x_3 = 1$; And Figure 6.18 and Figure 6.19 illustrate the corresponding section graphs when $x_3 = 0$. From Figures 6.16–6.19 we can see the high accuracy at each point.

In the section we use SPLF's as the bridge to prove constructively that sample generalized hierarchical T–S fuzzy systems can be universal approximators with maximum norm or, with integral norm. Also some succinct realizing algorithm of approximating procedures are developed. These results can provide us with the theoretic basis for the wide application of T–S fuzzy systems [64–67]. A meaningful and important problem related to the subject for the future research is how we can study the corresponding problems when processing Mamdani fuzzy systems or, the t-norm being not the product '×'.

References

[1] Azeem M. F., Hanmandlu M. & Ahmad N., Generalization of adaptive neuro-fuzzy inference systems, *IEEE Trans. on Neural Networks,* 2000, **11**: 1332–1346.

[2] Buckley J. J. & Hayashi Y., Fuzzy input-output controllers are universal approximators, *Fuzzy Sets and Systems,* 1993, **58**: 273–278.

[3] Chan L. W., The interpolation property of fuzzy polynomial approximation, *Proc. of IEEE Internat. Symposium on Speech, Image Processing and Neural Networks,* 1994, **2**: 449–452.

[4] Chung F. -L. & Duan J. -C, On multistage fuzzy neural network modeling, *IEEE Trans. on Fuzzy Systems,* 2000, **8**: 125–142.

[5] Combs W. E. & Andrews J. E., Combinatorial rule explosion eliminated by a fuzzy rule configuration, *IEEE Trans. on Fuzzy Systems,* 1998, **6**: 1–11.

[6] Combs W. E. & Andrews J. E., Author's reply, *IEEE Trans. on Fuzzy Systems,* 1999, **7**: (no.3, pp.371–373; no.4, pp.477–479).

[7] Dick S. & Kandel A., Comment on 'Combinatorial rule explosion eliminated by a fuzzy rule configuration, *IEEE Trans. on Fuzzy Systems,* 1999, **7**: 475–477.

[8] Dickerson J. A. & Kosko B., Fuzzy function approximation with ellipsoidal rules, *IEEE Trans. on Systems, Man and Cybernet.-Part B,* 1996, **26**(4): 542–560.

[9] Filev D.P. & Yager R.R.,A generalized defuzzification method via bad distributions, *Internat.J.of Intelligent Systems,*1991,**6**:687–697.

[10] Friedman J. & Stuetzle W., Projection pursuit regression, *J. Amer. Statist. Assoc.,* 1981, **76**: 817–823.

[11] Ghosh S. & Razouqi Q.,et al,A survey of recent advances in fuzzy logic in telecommunication networks and new challenges,*IEEE Trans. on Fuzzy Systems, 1998,* **6**: 443–447.

[12] Ho D. W. C. Zhang P. -A. & Xu J., Fuzzy wavelet networks for function learning, *IEEE Trans. on Fuzzy Systems,* 2001, **9**: 200–211.

[13] Horikawa S., Furuhashi T. & Uchikawa Y., On fuzzy modeling using fuzzy neural networks with the back-propagation algorithm, *IEEE Trans. on Neural Networks,* 1992, **3**: 801–806.

[14] Huber P., Projection pursuit (with discussion), *Ann. Statist.,* 1985, **13**: 435–475.

[15] Hunt K. J. & Haas R., Extending the functional equivalence of radial basis function networks and fuzzy inference systems, *IEEE Trans. on Neural Networks,* 1996, **7**(3): 776–781.

[16] Izquierdo J. M. C., Dimitriadis Y. A. & Sánchez E. G., et al, Learning from noisy information in FasArt and FasBack neuro-fuzzy systems, *Neural Networks,* 2001, **14**: 407–425.

[17] Jang J. S. R., ANFIS: Adaptive-network-based fuzzy inference system, *IEEE Trans. on Systems, Man and Cybernet.,* 1993, **23**: 665–684.

[18] Jang J. S. R., Sun C. T. & Mizutani E., *Neuro-fuzzy and soft computing,* Prentice Hall, 1997.

[19] Kikuchi H., Otake A. & Nakanishi S., Functional completeness of hierarchical fuzzy modeling, *Information Sciences,* 1998, **110**: 51–61.

[20] Kosko B., Fuzzy associative memories, in: Kandel A.(Ed.) *Fuzzy expert systems,* Addison–Weley, Reading, MA, 1987.

[21] Kosko B., Fuzzy systems as universal approximators, *Proc. IEEE Internat. Conf. on Fuzzy Systems,* 1992: 1153–1162.

[22] Kwan H.K. & Cai Y., A fuzzy neural network and its application to pattern recognition, *IEEE Trans. on Fuzzy Systems,* 1994, **2**: 185–193.

[23] Landajo M., Río M. J. & Pérez R., A note on smooth approximation capabilities of fuzzy systems, *IEEE Trans. on Fuzzy Systems,* 2001, **9**: 229–236.

[24] Lee C. C., Fuzzy logic in control systems:fuzzy logic controller,part I, *IEEE Trans. on Systems, and Man,Cybernet.,* 1990, **20**: 419–435.

[25] Lee C. C., Fuzzy logic in control systems:fuzzy logic controller, part II, *IEEE Trans. on Systems, and Man, Cybernet.,*1990,**20**: 404–418.

[26] Li Hongxing, Output–back fuzzy logic systems and equivalence with feedback neural networks, *Chinese Science Bulletin,* 2000, **45**: 592–596.

[27] Li Hongxing, Interpolation mechanism of fuzzy control, *Science in China, Series E,* 1998, **41**: 312–320.

[28] Li Hongxing & Chen C. L. P., The equivalence between fuzzy logic systems and feedforward neural networks, *IEEE Trans. on Neural Networks,* 2000, **11**: 356–365.

[29] Li Hongxing, Chen P. C. P & Huang H. P., *Fuzzy Neural Intelligent Systems,* FL: CRC Press, 2001.

[30] Liang Y. C., Feng D. P. & Lee H. P. et al, Successive approximation training algorithm for feedforward neural networks, *Neurocomputing,* 2002, **42**(1-4): 311-322.

[31] Liu Puyin & Li Hongxing, Approximation of generalized fuzzy system to integrable function, *Science in China, Series E,* 2000, **43**: 618–628.

[32] Liu Puyin & Li Hongxing, Analyses for $L_p(\mu)$-norm approximation capability of generalized Mamdani fuzzy systems, *Information Science,* 2001, **138**: 195–210.

[33] Liu Puyin & Li Hongxing, Hierarchical TS fuzzy systems are universal approximators, *Information Science,* 2004, **155** (accepted for publication).

[34] Liu Puyin & Li Hongxing, Equivalence of generalized Tagaki-Sugeno fuzzy system and its hierarchical system, *J. Beijing Normal University* 2000, **36**(5): 612–618.

[35] Liu Puyin & Wu Mengda, *Fuzzy theory and applications,* Changsha: Press of National University if Defence Tech., 1998.

[36] Liu Puyin & Zhang Hanjiang, Research on fuzzy neural network theory—a survey, *Fuzzy Systems and Math.*, 1998, **10**(1): 77-87.

[37] Mao Z. H., Li Yeda & Zhang X. F., Approximation capability of fuzzy systems using translations and dilatations of one fixed function as membership functions, *IEEE Trans. on Fuzzy Systems,* 1997, **5**(3): 193–197.

[38] Mao Z. H., Zhang X. F. & Li Yeda, Research on fuzzy systems as function universal approximators, *Science in China, Series E,* 1997, **27**(4): 362–371.

[39] Mendel J. M., Uncertainty, fuzzy logic,and signal processing. *Signal Processing,* 2000, **80**: 913–933.

[40] Mendel J. M. & Liang Q., Comment on 'Combinatorial rule explosion eliminated by a fuzzy rule configuration', *IEEE Trans. on Fuzzy Systems,* 1999,**7**: 369–371.

[41] Mitaim S. & Kosko B., Adaptive joint fuzzy sets for function approximation, *Proc. IEEE Internat. Conf. on Neural Networks,* 1997: 537–542.

[42] Mohler R. R., *Nonlinear systems, Vol.1, Dynamics and control,* Prentice Hall, Englewood Cliffs, NJ, 1991.

[43] Nauck D. & Kruse R., A neuro-fuzzy method to learn fuzzy classification rules from data, *Fuzzy Sets and Systems,* 1997, **89**: 277–288.

[44] Nishina T. & Hagiwara M., Fuzzy inference neural network, *Neurocomputing,* 1997, **14**: 223–239.

[45] Nûrnberger A., Radetzky A. & Kruse R., Using recurrent neuro-fuzzy techniques for the identification and simulation of dynamic systems, *Neurocomputing,* 2001, **36**(1-4): 123–147.

[46] Park J. -H., Seo S. -J. & Park G. -T., Robust adaptive fuzzy controller for nonlinear system using estimation of bounds for approximation errors, *Fuzzy Sets and Systems,* 2003, **133**: 19–36.

[47] Pytelkova R. & Husek P., Stability Analysis of Discrete-Time Takagi-Sugeno fuzzy systems, *proc. of Internat. Conf. on Computational Science–ICCS,* 2002, **1**: 604–612.

[48] Quek C. & Tung W. L., A novel approach to the derivation of fuzzy membership functions using the Falcon-MART architecture, *Pattern Recog. Lett.,* 2001, 22: 941–958.

[49] Raju G. V. S., Zhou J. & Kisner R. A., Hierarchical fuzzy control, *Internat. J. of Control,* 1991, **54**: 1201–1216.

[50] Raju G. V. S. & Zhou J., Adaptive hierarchical fuzzy controller, *IEEE Trans. on Systems, Man and Cybernet.,* 1993, **23**: 973–980.

[51] Rattasiri W. & Halgamuge S. K., Computationally advantageous and stable hierarchical fuzzy systems for active suspension, *IEEE Trans. on Industrial Electronics,* 2003, **50**(1): 48–61.

[52] Rovatti R., Fuzzy piecewise multilinear and piecewise linear system as universal approximators in Sobolev norms, *IEEE Trans. on Fuzzy Systems,* 1998, **6**: 235–249.

[53] Rudin W., *Real and complex analysis(third edition),* McGraw-Hill Book Company, 1987.

[54] Runkler T. A. & Bezdek J. C., Function approximation with polynomial membership functions and alternating cluster estimation, *Fuzzy Sets and Systems,* 1999, **101**: 207–218.

[55] Salmeri M., Mencattini A. & Rovatti R., Function approximation using nonnormalized SISO fuzzy systems,*Internat. J. Approx. Reasoning,* 2001, **26**: 211–231.

[56] Stinchcombe M. B., Neural network approximation of continuous functionals and continuous functions on compactifications, *Neural Networks,* 1999, **12**: 467–477.

[57] Takagi T. & Sugeno M.,Fuzzy identification of system and its applications to modeling and control, *IEEE Trans. on Systems, Man, Cybernet.,* 1985, **15**: 116–132.

[58] Tanaka K. & Taniguchi T., Generalized Takagi-Sugeno fuzzy systems: rule reduction and robust control *Proc. of The Ninth IEEE International Conference on Fuzzy systems, FUZZ IEEE 2000,* 2000, **2**: 688–693.

[59] Taniguchi T. & Tanaka K., Nonlinear rule reduction and robust control, *Proc. of Joint 9th IFSA World Congress and 20th NAFIPS Internat. Conf. (IFSA/NAFIPS 2001)*, 2001, **4**: 2050–2055.

[60] Taniguchi T. & Tanaka K., Rule reduction and robust control of generalized Takagi-Sugeno fuzzy systems, *Journal of Advanced Computational Intelligence*, 2000, **4**(5): 373–379.

[61] Wan Feng & Wang L. X., Design of economical fuzzy systems using least fuzzyl rules, *Proc. of 9th IEEE Internat. Conf. on Fuzzy Systems (FUZZ-IEEE 2000)*, 2000, **2**: 1052-1055.

[62] Wang Guojun, Implication 3-I algorithm for fuzzy inference, Since in China, Series E, 1999, **29**: 43–53.

[63] Wang L. X., Universal approximation by hierarchical fuzzy systems, *Fuzzy Sets and Systems*, 1998, **93**: 223–230.

[64] Wang L. X., Analysis and design of hierarchical fuzzy systems, *IEEE Trans. on Fuzzy Systems*, 1999, **7**: 617–624.

[65] Wang L. X., *Adaptive fuzzy systems and control: design stability analysis*, PTR Prentice-Hall, Englewood Cliffs, NJ, 1994.

[66] Wang L. X.& Mendel J. M., Fuzzy basis functions, universal approximation, and orthogonal least-squares learning, *IEEE Trans. on Neural Networks*, 1992, **3**: 807–814.

[67] Wang L. X. & Wan Feng, Structured neural networks for constrained model predictive control, *Automatica*, 2001, **37**(8): 1235–1243.

[68] Wang W. -Y., Lee T. -T. & Liu C. -L., et al, Function approximation using fuzzy neural networks with robust learning algorithm, *IEEE Trans. on Systems, Man and Cybernet. -Part B*, 1997, **27**: 740–747.

[69] Wong C. -C. & Chen C. -C., A GA-based method for constructing fuzzy systems directly from numerical data, *IEEE Trans. on Systems, Man and Cybernet.- Part B*, 2000, **30:** 904–911.

[70] Wu S. & Er M. J., Dynamic fuzzy neural networks— a novel approach to function approximation, *IEEE Trans. on Systems, Man and Cybernet.-Patr B*, 2000, **30**: 358–364.

[71] Yang Dali & Liu Zemin, Analysis for wrong adjustment of BP algorithm in multi-layer feedforward neural networks and its improvement, *Acta Electron. Sinca*, 1995, **23**(1): 117–120.

[72] Ying H., Sufficient conditions on general fuzzy systems as function approximators, *Automatica*, 1994, **30**: 521–525.

[73] Ying H., General SISO Takagi-Sugeno fuzzy systems with linear rule consequent are universal approximators, *IEEE Trans. on Fuzzy Systems*, 1998, **6**: 582–587.

[74] Ying H., Sufficient conditions on uniform approximation of multivariate functions by general Takagi–Sugeno fuzzy systems with linear rule consequent, *IEEE Trans. on Systems, Man, and Cybernet.*, 1998, **28**: 515–520.

[75] Ying H., General Takagi-Sugeno fuzzy systems with simplified linear rule consequent are universal controllers,models and filters, *Journal of Information Sciences*, 1998, **108**: 91–107.

[76] Ying H., & Chen G. R., Necessary conditions for some typical fuzzy systems as universal approximators, *Automatica*, 1997, **33**: 1333-1338.

[77] Zeng X. J. & Singh M. G., Approximation theory of fuzzy systems–MIMO case, *IEEE Trans. on Fuzzy Systems,* 1995, **3**: 219–235.

[78] Zeng X. J. & Singh M. G., Approximation accuracy analysis of fuzzy systems as function approximators, *IEEE Trans. on Fuzzy Systems,* 1996, **4**: 44–63.

[79] Zhang Y. Q. & Kandel A., Compensatory neurofuzzy systems with fast learning algorithms, *IEEE Trans. on Neural Networks,* 1998, **9**: 83–105.

CHAPTER VII
Stochastic Fuzzy Systems and Approximations

Learning by human knowledge and experience is an essential activity in fuzzy systems for achieving all kinds of information, including natural linguistic information. As in the case of neural networks (see [8, 48]), the learning capability of a fuzzy system is closely related to its approximating ability. The capability of fuzzy systems in approximating arbitrary non-random I/O mappings has recently been demonstrated in the works of Wang [59], Wang and Chen [60], Ding et al [12], Ying and Ding [61–63], Liu and Li [38, 40] and so on for T–S fuzzy systems, the works of Liu and Li [37], Zeng and Singh [64, 65], Abe and Lan [1], etc for Mamdani fuzzy systems.

Since in many practical cases the I/O mappings related are undoubtedly stochastic, the approximate realization of stochastic processes by some known systems, such as artificial neural networks [6], begins to attract many scholars' attention [4, 20, 56]. Of course such neural networks have so far been restricted to a small class which are called the approximation identity networks [10, 11, 55].

As a kind of intelligent systems, fuzzy systems should possess approximating capability to stochastic systems since human speech, human inference and expert knowledge etc. all can be inherently stochastic [7, 42]. So it is natural and important to study in depth the approximate realization of stochastic processes by fuzzy systems. However few achievements have so far been achieved in such a field [35, 39]. This chapter is devoted to the approximating capabilities of Mamdani fuzzy systems and T–S fuzzy systems to a class of stochastic processes. To this end, the two classes of fuzzy systems are extended to stochastic ones. In mean square sense the stochastic fuzzy systems based on the fuzzy operator composition '$\vee - \times$' can approximate a class of stochastic processes including stationary processes and weakly stationary processes to arbitrary degree of accuracy. Furthermore some efficient learning algorithms for the stochastic fuzzy systems is developed.

§7.1 Stochastic process and stochastic integral

As a preliminary for studying stochastic fuzzy system and its approximation to stochastic processes we in the section recall stochastic process and stochastic integral. Let $(\Omega, \mathcal{A}, P)$ be a probability space, and $u: \Omega \longrightarrow \mathbb{R}$ be a random variable with the following conditions: $E(u)=0,\ E(u^2)<+\infty$. All this type of random variables constitute a Hilbert space, which is denoted by $L^2(\Omega)$, where $\langle\cdot,\cdot\rangle$ in inner product, and the corresponding norm is $\|\cdot\|$, $E(u)$ is the expectation of u :

$$\forall u,\ v\in L^2(\Omega),\ \langle u,\ v\rangle= E(u\cdot v),\ \ \|u\|=\{E(u^2)\}^{\frac{1}{2}}.$$

Let $\{u_n,\ n\in\mathbb{N}\}\subset L^2(\Omega)$ be a sequence of random variables, $u\in L^2(\Omega)$. If $\lim\limits_{n\to+\infty} E(|u_n-u|^2)=0$, we call $\{u_n,\ n\in\mathbb{N}\}$ converges to u with mean square sense, which is denoted by $u_n \stackrel{m.s.}{\longrightarrow} u\ (n\to+\infty)$.

Suppose the stochastic process $x=\{x(t),\ t\in\mathbb{R}_+\}$ satisfies the following condition: for any $t\in\mathbb{R}_+,\ x(t)\in L^2(\Omega)$. For $s,\ t\in\mathbb{R}_+$, denote $B_x(s,t)=E\big(x(t)\cdot x(s)\big)$, we call $B_x(\cdot,\cdot)$ the covariance function of x. Set

$$\mathcal{C}(\Omega)=\Big\{x=\{x(t),t\in\mathbb{R}_+\}\,\big|\,\exists\,\psi(\cdot,\cdot):\mathbb{R}_+^2\longrightarrow\mathbb{R},\ \psi(t,\cdot)\in L^2(\mathbb{R}_+,\mathcal{B},F),$$
$$B_x(s,t)=\int_0^{+\infty}\psi(s,\theta)\psi(t,\theta)F(\mathrm{d}\theta)\Big\},$$

where $\mathcal{B}$ is Borel algebra on $\mathbb{R}_+$, and F is a finite measure on $\mathcal{B}$. If $x\in\mathcal{C}(\Omega)$, and $B_x(t,\ t+\delta t)$ is a function of δt, independent of t, we call x a weak stationary process.

7.1.1 Stochastic measure and stochastic integral

In order obtain the canonical representation of each process in $\mathcal{C}(\Omega)$, we at first recall stochastic measure and its integral in a unifying framework. Let $\eta:\mathcal{B}\longrightarrow L^2(\Omega)$ satisfy the following condition: $\forall C_1,\ C_2\in\mathcal{B},\ E(\eta(C_1)\eta(C_2))=F(C_1\cap C_2)$. Then η is called a stochastic measure based on F. $\eta:\mathcal{B}\longrightarrow L^2(\Omega)$ is a stochastic measure if and only if the following conditions hold [15, 19]:

(i) $\forall A,\ A_1,\ A_2\in\mathcal{B},\ \ A_1\cap A_2=\emptyset,\Longrightarrow\ \langle\eta(A_1),\ \eta(A_2)\rangle=0$, moreover $\|\eta(A)\|=F(A)$;

(ii) $\{B_0,\ B_1,\ B_2,...\}\subset\mathcal{B}:\ \bigcup\limits_{i=1}^{+\infty}B_i=B_0,\ \ B_i\cap B_j=\emptyset\ (i\neq j),\Longrightarrow \sum\limits_{i=1}^{q}\eta(B_i)\stackrel{m.s.}{\longrightarrow}\eta(B_0)\ (q\to+\infty)$;

Suppose $f\in L^2(\mathbb{R}_+,\ \mathcal{B},\ F)$. In the following we present the integral of f with respect to $\eta(\cdot)$. Firstly assume f is a simple function:

$$f(x)=\sum_{i=1}^{n}a_i\chi_{B_i}(x)\ (B_i\in\mathcal{B},\ a_i\in\mathbb{R},\ i=1,...,n).$$

Define $I(f)=\sum\limits_{i=1}^{n}a_i\eta(B_i)$. we can conclude the following facts [15]:

(i) $I(f)$ is independent of the expressing form of the simple function f;

(ii) If f, g are simple functions, a, $b \in \mathbb{R}$, then $I(af + bg) = aI(f) + bI(g)$;

(iii) Provided f, g are simple functions, we have, $\|I(f)\| = \|f\|_{L_2(\mathbb{R}_+)}$, moreover $\langle I(f), I(g)\rangle = \int_{\mathbb{R}_+} f(x)g(x)F(\mathrm{d}x)$.

For any $f \in L^2(\mathbb{R}_+, \mathcal{B}, F)$, there exist a sequence of simple functions $\{f_n | n \in \mathbb{N}\} : \|f - f_n\|_{L_2(\mathbb{R}_+)} \to 0$ $(n \to +\infty)$. Define $I(f) = \lim_{n\to+\infty} I(f_n)$. $I(f)$ is the stochastic integral of f with respect to η : $I(f) = \int_{\mathbb{R}_+} f(\theta)\eta(\mathrm{d}\theta)$.

By Karhunen Theorem [15, 22], $\forall x \in \mathcal{C}(\Omega)$, there is a stochastic measure η on $(\mathbb{R}_+, \mathcal{B}, F)$ satisfying

$$\forall t \in \mathbb{R}_+,\ x(t) = \int_{\mathbb{R}_+} \psi(t,\theta)\eta(\mathrm{d}\theta), \tag{7.1}$$

moreover, $E(|\eta(I)|^2) = F(I)$ $(I \in \mathcal{B})$. Suppose $\gamma = \{\gamma(\theta),\ \theta \in \mathbb{R}_+\}$ is a stochastic process with orthogonal increments, that is, $\forall\theta_1, \theta_2, \theta_3, \theta_4 \in \mathbb{R}_+$, it yields that

$$\theta_1 < \theta_2 \le \theta_3 < \theta_4, \Longrightarrow E\big((\gamma(\theta_2) - \gamma(\theta_1))(\gamma(\theta_4) - \gamma(\theta_4))\big) = 0. \tag{7.2}$$

Stochastic measure η may be generated by an orthogonal increment process as $\gamma = \{\gamma(\theta),\ \theta \in \mathbb{R}_+\}$. In fact, if $I = [\theta_1, \theta_2) \in \mathcal{B}$, let $\eta(I) = \gamma(\theta_2) - \gamma(\theta_1)$. Thus, (7.1) may be expressed as

$$\forall t \in \mathbb{R}_+,\ x(t) = \int_{\mathbb{R}_+} \psi(t,\theta)\mathrm{d}\gamma(\theta). \tag{7.3}$$

Define a real function $f : \mathbb{R}_+ \longrightarrow \mathbb{R}_+$ as following: $f(\theta) \triangleq E\big(|\gamma(\theta) - \gamma(0)|^2\big)$ for each $\theta \in \mathbb{R}_+$. By (7.2) easily we can show, $f(\cdot)$ is increasing and left continuous. For any semi-closed interval $I = [\theta_1, \theta_2)$, define $F(I) = f(\theta_2) - f(\theta_1)$. By the measure extension theorem [15], we can establish a measure on $\mathcal{B}$ determined by F, which is also denoted by F.

Using the finite measure F on $\mathcal{B}$ we can construct an isometric mapping of $L^2(\mathbb{R}_+, \mathcal{B}, F)$ to $L^2(\Omega)$. In fact, for any $g(\cdot) \in L^2(\mathbb{R}_+, \mathcal{B}, F)$, by the stochastic integral $J(g) = \int_{\mathbb{R}_+} g(\theta)\mathrm{d}\gamma(\theta)$, we may define a random variable $J(g) \in L^2(\Omega)$. Given $g_1, g_2 \in L^2(\mathbb{R}_+, \mathcal{B}, F)$, it follows by [15, 22] that

$$E\Big(\Big[\int_{\mathbb{R}_+} g_1(\theta)\mathrm{d}\gamma(\theta)\Big]\cdot\Big[\int_{\mathbb{R}_+} g_2(\theta)\mathrm{d}\gamma(\theta)\Big]\Big) = \int_{\mathbb{R}_+} g_1(\theta)g_2(\theta)F(\mathrm{d}\theta). \tag{7.4}$$

For $f_1, f_2 \in L^2(\mathbb{R}_+, \mathcal{B}, F)$, put $u_k = J(f_k)$ $(k = 1,2)$. In (7.4) letting $g_1 = g_2 = f_1 - f_2$ we get

$$\begin{aligned}\|f_1 - f_2\|^2_{L_2(F)} &= \int_{\mathbb{R}_+} |f_1(\theta) - f_2(\theta)|^2 F(\mathrm{d}\theta) = E\Big(\Big[\int_{\mathbb{R}_+} (f_1(\theta) - f_2(\theta))\mathrm{d}\gamma(\theta)\Big]^2\Big)\\ &= E([J(f_1) - J(f_2)]^2) = E([u_1 - u_2]^2) = \|u_1 - u_2\|^2.\end{aligned} \tag{7.5}$$

Therefore $J: L^2(\mathbb{R}_+, \mathcal{B}, F) \longrightarrow L^2(\Omega)$ is an isometric mapping.

Also by [15, 22], for any $x \in \mathcal{C}(\Omega)$, (7.1) (7.3) can be extended to the vector case, and consequently, the covariance function of x is

$$B_x(s,t) = \int_{\mathbb{R}_+} \Big\langle \Psi^{\mathrm{T}}(s,\theta),\ \Psi(t,\theta) \Big\rangle F(\mathrm{d}\theta),$$

where $\langle \cdot,\ \cdot \rangle$ also denotes the inner product of the vector valued function $\Psi^{\mathrm{T}} = (\psi_1,\ \psi_2, \dots)$, where Ψ^{T} means the transpose of Ψ. Thus, a canonical representation of $x \in \mathcal{C}(\Omega)$ is as follows:

$$\forall t \in \mathbb{R}_+,\ x(t) = \int_{\mathbb{R}_+} \langle \Psi^{\mathrm{T}}(t,\theta), \Upsilon(\mathrm{d}\theta) \rangle, \tag{7.6}$$

where $\Upsilon(\cdot) = \big(\eta_1(\cdot),\ \eta_2(\cdot), \dots\big)^{\mathrm{T}}$ is a vector valued measure, satisfying

$$\forall I \in \mathcal{B},\ E(|\eta_1(I)|^2) = E(|\eta_2(I)|^2) = \cdots = F(I),\quad E(\eta_i(I)\eta_j(I)) = 0\ (i \neq j).$$

If $\Gamma = (\gamma_1, \gamma_2, \dots)^{\mathrm{T}}$, and $\gamma_i = \{\gamma_i(\theta),\ \theta \in \mathbb{R}_+\}$ $(i = 1, 2, \dots)$ is a process with orthogonal increments, moreover

$$\forall I = [\theta_1,\ \theta_2) \in \mathcal{B}, \eta_i(I) = \gamma_i(\theta_2) - \gamma_i(\theta_1)\ (i = 1, 2, \dots).$$

Hence (7.6) can be expressed as

$$\forall t \in \mathbb{R}_+,\ x(t) = \int_{\mathbb{R}_+} \langle \Psi^{\mathrm{T}}(t,\theta), \mathrm{d}\Gamma(\theta) \rangle, \tag{7.7}$$

where $\Psi^{\mathrm{T}}(t,\ \theta) = \big(\phi_1(t,\theta),\ \phi_2(t,\theta), \dots\big)$, $\mathrm{d}\Gamma(\theta) = \big(\mathrm{d}\gamma_1(\theta),\ \mathrm{d}\gamma_2(\theta), \dots\big)^{\mathrm{T}}$. We call (7.7) a canonical representation of the stochastic process $x = \{x(t),\ t \in \mathbb{R}_+\}$. In (7.7) $\gamma_i(\cdot)$ $(i = 1, 2, \dots)$ is an orthogonal increment process with the following conditions:

$$E(|\mathrm{d}\gamma_1(\theta)|^2) = E(|\mathrm{d}\gamma_2(\theta)|^2) = \cdots = F(\mathrm{d}\theta),\quad E(\mathrm{d}\gamma_i(\theta)\mathrm{d}\gamma_j(\theta)) = 0\ (i \neq j). \tag{7.8}$$

7.1.2 Canonical representation of Brownian motion

For applying convenience in the following we transform the vector valued stochastic process $\Gamma = \{\Gamma(t),\ t \in \mathbb{R}_+\}$ in (7.7) into Brownian motion. To this end we introduce the definition of the standard Brownian motion. $b = \{b(t),\ t \in \mathbb{R}_+\}$ is called a standard Brownian motion, if

(i) $\forall \theta_1 < \theta_2$, the increment $b(\theta_2) - b(\theta_1)$ is a real random variable whose distribution is standard normal;

(ii) $\forall \theta_1,\ \theta_2,\ \theta_3,\ \theta_4:\ \theta_1 < \theta_2 \leq \theta_3 < \theta_4 \Longrightarrow b(\theta_2) - b(\theta_1)$ and $b(\theta_4) - b(\theta_3)$ are independent;

(iii) $E(|b(\theta_2)-b(\theta_1)|^2)=|\theta_2-\theta_1|$. Consequently $E(|\mathrm{d}b(\theta)|^2)=\mathrm{d}\theta$.

Let $B^{\mathrm{T}}(\theta)=\big(b_1(\theta),\,b_2(\theta),...\big)$, where $b_1,\,b_2,...$, are standard Brownian motions that are independent, satisfying: $E(|b_i(\theta)|^2)=\mathrm{d}\theta\triangleq M(\mathrm{d}\theta)$. Let $M(\cdot)$ is Lebesgue-Stieltjes measure on $\mathcal{B}$. Then for $I=[\theta_1,\,\theta_2)\in\mathcal{B},\ M(I)=E\big(|b_i(\theta_2)-b_i(\theta_1)|^2\big)$. Suppose $C(\cdot)$ is a nonnegative function with the following condition: $\gamma_i(\theta_2)-\gamma_i(\theta_1)=\int_{\theta_1}^{\theta_2}C(\theta)\mathrm{d}b_i(\theta)$. Then by (7.8) and the fact that b_i is an orthogonal increment process, it follows that

$$E\big(|\gamma_i(\theta_2)-\gamma_i(\theta_1)|^2\big)=\int_{\theta_1}^{\theta_2}C(\theta)^2M(\mathrm{d}\theta)\Longrightarrow F(\mathrm{d}\theta)=C(\theta)^2\mathrm{d}\theta=C(\theta)^2M(\mathrm{d}\theta). \tag{7.9}$$

If there is a density function $f(\cdot)$ of F, then $f(\theta)=C(\theta)^2$. Let $\Phi^{\mathrm{T}}(t,\theta)=C(\theta)\Psi^{\mathrm{T}}(t,\theta)$, then

$$\forall t,\,s\in\mathbb{R}_+,\ \ B_x(t,s)=\int_{\mathbb{R}_+}\Big\langle\Phi^{\mathrm{T}}(t,\theta),\ \Phi(s,\theta)\Big\rangle M(\mathrm{d}\theta).$$

Thus, the canonical representation of (7.7) x can be written as

$$\forall t\in\mathbb{R}_+,\ \ x(t)=\int_{\mathbb{R}_+}\Big\langle\Phi^{\mathrm{T}}(t,\theta),\ \mathrm{d}B(\theta)\Big\rangle, \tag{7.10}$$

where $\mathrm{d}B(\theta)=\big(\mathrm{d}b_1(\theta),\,\mathrm{d}b_2(\theta),...\big)^{\mathrm{T}}$, and

$$\Phi^{\mathrm{T}}(t,\,\theta)=C(\theta)\Psi^{\mathrm{T}}(t,\,\theta)=\big(C(\theta)\phi_1(\theta),\,C(\theta)\phi_2(\theta),...\big).$$

Let $(\mathbb{R}_+,\,\mathcal{B},\,G)$ be a finite measure space, that is, $G(\mathbb{R}_+)<+\infty$. So in the following we may assume $G(\mathrm{d}t)=\exp(-10t)\mathrm{d}t$. Considering

$$L^2(\mathbb{R}_+,\,\mathcal{B},\,G)=\Big\{f:\ \mathbb{R}_+\longrightarrow\mathbb{R}\,\Big|\int_{\mathbb{R}_+}|f(t)|^2G(\mathrm{d}t)<+\infty\Big\},$$

we can rewrite the finite product measure space $L^2\big(\mathbb{R}_+^2,\,\mathcal{B}\times\mathcal{B},\,G\times F\big)$ as

$$L^2(\mathbb{R}_+^2,\,\mathcal{B}\times\mathcal{B},\,G\times F)\triangleq\Big\{f:\ \mathbb{R}_+^2\longrightarrow\mathbb{R}\Big|\int_{\mathbb{R}_+^2}\big|f(t,s)\big|^2F(\mathrm{d}\theta)G(\mathrm{d}t)<+\infty\Big\}.$$

For $f\in L^2(\mathbb{R}_+^2,\,\mathcal{B}\times\mathcal{B},\,G\times F)$, let $\|f\|_{L_2(G\times F)}=\Big\{\int_{\mathbb{R}_+^2}\big|f(t,s)\big|^2F(\mathrm{d}\theta)G(\mathrm{d}t)\Big\}^{\frac{1}{2}}$.

By Theorem 6.3, Theorem 6.4 and Corollary 6.4, to the generalized Mamdani fuzzy systems and generalized T–S fuzzy systems as (6.11) (6.17), respectively letting $\alpha=1,\,d=2$, easily we have the following conclusion.

Theorem 7.1 *Suppose* $g\in L^2(\mathbb{R}_+^2,\,\mathcal{B}\times\mathcal{B},\,G\times F)$. *For any* $\varepsilon>0$, *choose* $a>0$ *so that*

$$\int_{t>a,\theta>a}\big|g(t,\theta)\big|^2F(\mathrm{d}\theta)G(\mathrm{d}t)<\frac{\varepsilon^2}{8}.$$

Then there is a sufficiently small $h > 0$, if let

$$D_H(g) = \bigvee_{t,\theta,t+h,\theta+h\in[0,m]} \left\{ \frac{|g(t,\theta+h)-g(t,\theta)|}{h} \vee \frac{|g(t+h,\theta)-g(t,\theta)|}{h} \right\}.$$

there exists a fuzzy system defined as following:

$$M_{a,m}(t,\theta) = \frac{\sum_{p_1,p_2=0}^{m} H_{p_1p_2}(t,\theta)\cdot g\left(\frac{ap_1}{m}, \frac{ap_2}{m}\right)}{\sum_{p_1,p_2=0}^{m} H_{p_1p_2}(t,\theta)},$$

so that provided $m > 4D_H(g)ac_0 \cdot \sqrt{(G\times F)\big([0,a]^2\big)}/\varepsilon$, it follows that,

$$\left\{\int_{\mathbb{R}_+^2} \big|g(t,\theta) - M_{a,m}(t,\theta)\big|^2 F(\mathrm{d}\theta)G(\mathrm{d}t)\right\}^{\frac{1}{2}} < \varepsilon.$$

§7.2 Stochastic fuzzy systems

Fuzzy systems can deal with linguistic and numerical information, simultaneously [3, 26, 32]. So it is undoubtedly very important to study their approximation in stochastic environment, that is, the approximation capabilities of fuzzy systems to stochastic processes. To this end in this section we introduce stochastic fuzzy systems, and study stochastic integral of a fuzzy system.

Since one of the variable sets related to a stochastic process related is time parameters, we can discuss our subjects in $\mathbb{R}_+ \times \Omega$. Choose the antecedent fuzzy set family $\{\tilde{A}_{ij}\ |i=1,2;\ j=0,1,...,m\} \subset \tilde{\mathcal{O}}_0(a,\ m)$ on $\mathbb{R}_+$, which satisfies the following condition: $t \notin [0,\ a]$, $\tilde{A}_{ij}(t) = 0$; $\forall t \in [0,\ a]$, $i = 1,\ 2$, $\sum_{j=0}^{m} \tilde{A}_{ij}(t) \equiv 1$. Suppose the continuous t−norm T is '$\times$', which is also written as '$\cdot$'. Then $N(t_1,\ t_2) = \{(p_1,p_2)|\tilde{A}_{1p_1}(t_1)\cdot \tilde{A}_{2p_2}(t_2) > 0\}$, $N(t) = \{p|\ \tilde{A}_{1p}(t) > 0\}$.

7.2.1 Stochastic T–S fuzzy system

Using the antecedent fuzzy set $\tilde{A}_{ij}$ $(i=1,2;\ j=0,1,...,m)$ we can obtain the T–S fuzzy inference rule with two input variables:

$TR_{p_1p_2}$: IF t_1 is $\tilde{A}_{1p_1}$ and t_2 is $\tilde{A}_{2p_2}$ THEN u is $b_{0;p_1p_2} + b_{1;p_1p_2}t_1 + b_{2;p_1p_2}t_2$,

where $b_{0;p_1p_2}$, $b_{1;p_1p_2}$, $b_{2;p_1p_2}$ $(p_1,\ p_2 = 0,1,...,m)$ are adjustable real parameters. Corresponding to above T–S inference rule, and letting $\alpha = 1$ in (6.17) we express the I/O relationship of T–S fuzzy system as follows [17, 23, 65]:

$$S_{a,m}(t_1,\ t_2) = \frac{\sum_{p_1,p_2=0}^{m}\Big(H_{p_1p_2}(t_1,t_2)\cdot\big(b_{0;p_1p_2} + b_{1;p_1p_2}t_1 + b_{2;p_1p_2}t_2\big)\Big)}{\sum_{p_1,p_2=0}^{m} H_{p_1p_2}(t_1,t_2)}, \quad (7.11)$$

where $t_1,\, t_2 \in \mathbb{R}_+$, and let $0/0 \equiv 0$. And $H_{p_1p_2}(t_1, t_2) = \widetilde{A}_{1p_1}(t_1)\cdot \widetilde{A}_{2p_2}(t_2)$. Since $\sum\limits_{p=0}^{m} \widetilde{A}_{ij}(t) \equiv 1\ (i = 1, 2)$, we get

$$\begin{aligned}\sum_{p_1,p_2=0}^{m} H_{p_1p_2}(t_1,\, t_2) &= \sum_{p_1,p_2=0}^{m} \big[\widetilde{A}_{1p_1}(t_1)\big]\big[\widetilde{A}_{2p_2}(t_2)\big] \\ &= \Big(\sum_{p_1=0}^{m} \widetilde{A}_{1p_1}(t_1)\Big)\Big(\sum_{p_2=0}^{m} \widetilde{A}_{2p_2}(t_2)\Big) = 1.\end{aligned}$$

So (7.11) can be rewritten as

$$\begin{aligned}S_{a,m}(t_1,t_2) &= \sum_{p_1,p_2=0}^{m} H_{p_1p_2}(t_1,t_2)\big(b_{0;p_1p_2} + b_{1;p_1p_2}t_1 + b_{2;p_1p_2}t_2\big) \\ &= \sum_{p_1,p_2=0}^{m} \widetilde{A}_{1p_1}(t_1)\cdot \widetilde{A}_{2p_2}(t_2)\big(b_{0;p_1p_2} + b_{1;p_1p_2}t_1 + b_{2;p_1p_2}t_2\big)\end{aligned} \tag{7.12}$$

for $t_1, t_2 \in \mathbb{R}_+$. Similarly we can express an one dimensional T–S fuzzy system as follows:

$$S_{a,m}(t) = \sum_{p=0}^{m} T_p(t)(b_{0;p} + b_{1;p}t) = \sum_{p=0}^{m} \widetilde{A}_{1p}(t)(b_{0;p} + b_{1;p}t)\ (t \in \mathbb{R}_+). \tag{7.13}$$

If the adjustable parameters in (7.11) (7.12) are random variables, the corresponding systems are called stochastic T–S fuzzy systems.

Next let us represent approximately the stochastic integral of $S_{a,m}(\cdot,\cdot)$ as the sum of one dimensional stochastic T–S fuzzy systems.

Theorem 7.2 *For any $a > 0$ and $m \in \mathbb{N}$, let the T–S fuzzy system $S_{a,m}(\cdot,\,\cdot)$ be defined as (7.12), and $\gamma = \{\gamma(\theta),\ \theta \in \mathbb{R}_+\}$ be a stochastic process with orthogonal increments. Then for arbitrary $t \in \mathbb{R}_+$, the stochastic integral $\int_{\mathbb{R}_+} S_{a,m}(t,\,\theta)\mathrm{d}\gamma(\theta)$ exists, moreover for any $\varepsilon > 0$, there exist $\theta_1, ..., \theta_q:\ 0 = \theta_0 < \theta_1 < \cdots < \theta_q$, independently of t satisfying: $\forall t \in \mathbb{R}_+$, the following fact holds:*

$$E\Big(\Big|\int_{\mathbb{R}_+} S_{a,m}(t,\theta)\mathrm{d}\gamma(\theta) - \sum_{j=1}^{q} S_{a,m}(t,\theta_j)\Delta\gamma_j\Big|^2\Big)^{\frac{1}{2}} < \varepsilon,$$

where $\Delta\gamma_j = \gamma(\theta_j) - \gamma(\theta_{j-1})\ (j = 1, ..., q)$.

Proof. By (7.12), $\theta > a \Longrightarrow S_{a,m}(t,\,\theta) \equiv 0$, and when $t \leq a$, we get

$$S_{a,m}(t,\,\theta) = \sum_{p_1=0}^{m} \widetilde{A}_{1p_1}(t)\Big(\sum_{p_2=0}^{m} \big(b_{0;p_1p_2} + b_{1;p_1p_2}t\big)\widetilde{A}_{2p_2}(\theta) + \sum_{p_2=0}^{m} \widetilde{A}_{2p_2}(\theta)b_{2;p_1p_2}\theta\Big).$$

Since $\widetilde{A}_{1p_1}(\cdot)$, $\widetilde{A}_{2p_2}(\cdot)$ are Riemann integrable on $[0, a]$, we imply, stochastic integral $\int_0^a S_{a,m}(t, \theta)\mathrm{d}\gamma(\theta)$ exists. So $\int_{\mathbb{R}_+} S_{a,m}(t, \theta)\mathrm{d}\gamma(\theta)$ also exists. Moreover

$$\int_{\mathbb{R}_+} S_{a,m}(t,\theta)\mathrm{d}\gamma(\theta) = \sum_{p_1=0}^{m} \widetilde{A}_{1p_1}(t)\bigg(\sum_{p_2=0}^{m}\big(b_{0;p_1,p_2} + b_{1;p_1p_2}t\big)\int_0^a \widetilde{A}_{2p_2}(\theta)\mathrm{d}\gamma(\theta) + \sum_{p_2=0}^{m} b_{2;p_1,p_2}\int_0^a \theta\cdot \widetilde{A}_{2p_2}(\theta)\mathrm{d}\gamma(\theta)\bigg). \tag{7.14}$$

For the stochastic integrals $\int_0^a \widetilde{A}_{2p_2}(\theta)\mathrm{d}\gamma(\theta)$ and $\int_0^a \theta\cdot \widetilde{A}_{2p_2}(\theta)\mathrm{d}\gamma(\theta)$, there are $\theta_1, ..., \theta_q \in \mathbb{R}_+$, independently of t so that $0 = \theta_0 < \theta_1 < \cdots < \theta_q$, and we have

$$\begin{cases} E\bigg(\bigg|\displaystyle\int_0^a \widetilde{A}_{2p_2}(\theta)\mathrm{d}\gamma(\theta) - \sum_{j=1}^{q}\widetilde{A}_{2p_2}(\theta_j)\Delta\gamma_j\bigg|^2\bigg)^{\frac{1}{2}} < \dfrac{2^{-\frac{m+4}{2}}\cdot\varepsilon}{\sum\limits_{p_1,p_2=0}^{m}\big(b_{0;p_1p_2}^2 + a^2\cdot b_{1;p_1p_2}^2\big)}, \\ E\bigg(\bigg|\displaystyle\int_0^a \theta\cdot \widetilde{A}_{2p_2}(\theta)\mathrm{d}\gamma(\theta) - \sum_{j=1}^{q}\theta_j\cdot \widetilde{A}_{2p_2}(\theta_j)\Delta\gamma_j\bigg|^2\bigg)^{\frac{1}{2}} < \dfrac{2^{-\frac{m+4}{2}}\cdot\varepsilon}{\sum\limits_{p_1,p_2=0}^{m} b_{2;p_1p_2}^2}. \end{cases} \tag{7.15}$$

Easily we can show, $\forall t > a$, $S_{a,m}(t, \theta) \equiv 0$, and for $t \le a$, the following fact holds:

$$\begin{aligned} &\int_{\mathbb{R}_+} S_{a,m}(t,\theta)\mathrm{d}\gamma(\theta) - \sum_{j=1}^{q} S_{a,m}(t,\theta_j)\Delta\gamma_j \\ &= \sum_{p_1=0}^{m}\widetilde{A}_{1p_1}(t)\bigg(\sum_{p_2=0}^{m}\big(b_{0;p_1p_2}+b_{1;p_1p_2}t\big)\bigg[\int_0^m \widetilde{A}_{2p_2}(\theta)\mathrm{d}\gamma(\theta) - \sum_{j=1}^{q}\widetilde{A}_{2p_2}(\theta_j)\Delta\gamma_j\bigg] \\ &\quad + \sum_{p_2=0}^{m} b_{2;p_1p_2}\bigg[\int_0^m \theta\cdot \widetilde{A}_{2p_2}(\theta)\mathrm{d}\gamma(\theta) - \sum_{j=1}^{q}\theta_j\, \widetilde{A}_{2;p_2}(\theta_j)\Delta\gamma_j\bigg]\bigg). \end{aligned} \tag{7.16}$$

For any $t \in [0, a]$, $\sum\limits_{p_1=0}^{m}[\widetilde{A}_{1p_1}(t)]^2 \le \sum\limits_{p_1=0}^{m}\widetilde{A}_{1p_1}(t) = 1$, by (7.14)–(7.16) it follows that

$$\begin{aligned} &\bigg\{E\bigg(\bigg|\int_{\mathbb{R}_+} S_{a,m}(t,\theta)\mathrm{d}\gamma(\theta) - \sum_{j=1}^{q} S_{a,m}(t,\theta_j)\Delta\gamma_j\bigg|^2\bigg)\bigg\}^{\frac{1}{2}} \\ &\le \bigg\{E\bigg(\sum_{p_1=0}^{m}\bigg|\sum_{p_2=0}^{m}\big(b_{0;p_1p_2}+b_{1;p_1p_2}t\big)\bigg\{\int_0^a \widetilde{A}_{2p_2}(\theta)\mathrm{d}\gamma(\theta) - \sum_{j=1}^{q}\widetilde{A}_{2p_2}(\theta_j)\Delta\gamma_j\bigg\} \\ &\quad + \sum_{p_2=0}^{m} b_{2;p_1p_2}\bigg\{\int_0^a \theta\cdot \widetilde{A}_{2p_2}(\theta)\mathrm{d}\gamma(\theta) - \sum_{j=1}^{q}\theta_j\cdot \widetilde{A}_{2;p_2}(\theta_j)\Delta\gamma_j\bigg\}\bigg|^2\bigg)\bigg\}^{\frac{1}{2}} \end{aligned}$$

$$\leq \sqrt{2}\Bigg\{\sum_{p_1=0}^{m} E\Big(\Big|\sum_{p_2=0}^{m}(b_{0;p_1p_2}+b_{1;p_1p_2}t)\Big\{\int_0^a \widetilde{A}_{2p_2}(\theta)\mathrm{d}\gamma(\theta)-\sum_{j=1}^{q}\widetilde{A}_{2p_2}(\theta_j)\Delta\gamma_j\Big\}\Big|^2\Big)$$

$$+\sum_{p_1=0}^{m} E\Big(\Big|\sum_{p_2=0}^{m} b_{2;p_1p_2}\Big\{\int_0^a \theta\cdot \widetilde{A}_{2p_2}(\theta)\mathrm{d}\gamma(\theta) - \sum_{j=1}^{q}\theta_j\cdot \widetilde{A}_{2p_2}(\theta_j)\Delta\gamma_j\Big\}\Big|^2\Big)\Bigg\}^{\frac{1}{2}}$$

$$\leq 2^{\frac{m+2}{2}}\Bigg\{\sum_{p_1,p_2=0}^{m}\Big[(b_{0;p_1p_2}^2+b_{1;p_1p_2}^2a^2)E\Big(\Big|\int_0^a \widetilde{A}_{2p_2}(\theta)\mathrm{d}\gamma(\theta)-\sum_{j=1}^{q}\widetilde{A}_{2p_2}(\theta_j\Delta\gamma_j)\Big|^2\Big)$$

$$+b_{2;p_1p_2}^2\cdot E\Big(\Big|\int_0^a \theta\cdot \widetilde{A}_{2p_2}(\theta)\mathrm{d}\gamma(\theta) - \sum_{j=1}^{q}\theta_j\cdot \widetilde{A}_{2p_2}(\theta_j\Delta\gamma_j)\Big|^2\Big)\Big]\Bigg\}^{\frac{1}{2}}$$

$$< 2^{\frac{m+2}{2}}\cdot\Big(\frac{\varepsilon}{2^{\frac{m+4}{2}}}+\frac{\varepsilon}{2^{\frac{m+4}{2}}}\Big)=\varepsilon.$$

The theorem is therefore proved. □

7.2.2 Stochastic Mamdani fuzzy system

Using fuzzy set $\widetilde{A}_{ij}$ $(i=1,2;\ j=0,1,...,m)$ we can design Mamdani inference rule. Taking two special cases, we can establish the related rules with one input variable, whose $p-$th fuzzy rule for $p=0,\ 1,...,\ m)$ as follows:

$$R_p: \text{IF } t \text{ is } \widetilde{A}_{1p} \text{ THEN } v \text{ is } \widetilde{V}_{O(p)},$$

where $O(\cdot)$ is a real function, and the consequent fuzzy set is a fuzzy number $\widetilde{V}_{O(p)}\in \mathcal{F}(\mathbb{R})$: $\mathrm{Ker}(\widetilde{V}_{O(p)}) = \{O(p)\}$; and with two input variables, whose p_1p_2-inference rule for $p_1,\ p_2=0,\ 1,...,m$ as

$$R_{p_1p_2}: \text{IF } t_1 \text{ is } \widetilde{A}_{1p_1} \text{ and } t_2 \text{ is } \widetilde{A}_{2p_2} \text{ THEN } u \text{ is } \widetilde{U}_{r(p_1,p_2)},$$

where r is a real function $r:\ \mathbb{R}^2 \longrightarrow \mathbb{R}$, and $\widetilde{U}_{r(p_1,p_2)}\in \mathcal{F}(\mathbb{R})$ is a fuzzy number whose kernel is $\mathrm{Ker}(\widetilde{U}_{r(p_1,p_2)})= \{r(p_1,\ p_2)\}$. By (6.11), we use the centroid defuzzification to get the corresponding I/O relationship:

$$M_{a,m}(t_1,\ t_2)=\frac{\sum_{p_1,p_2=0}^{m}\overline{y}_{p_1p_2}\cdot(\widetilde{A}\circ\widetilde{R}_{p_1p_2})(\overline{y}_{p_1p_2})}{\sum_{p_1,p_2=0}^{m}(\widetilde{A}\circ\widetilde{R}_{p_1p_2})(\overline{y}_{p_1p_2})}. \tag{7.17}$$

where $\overline{y}_{p_1p_2} = r(p_1,p_2)$ is a maximum value point of $\widetilde{U}_{r(p_1,p_2)}$ in $\mathbb{R}$, that is, $\widetilde{U}_{r(p_1,p_2)}(\overline{y}_{p_1p_2}) = 1$. Therefore, $(\widetilde{A}\ \circ W_{p_1p_2})(r(p_1,\ p_2)) = H_{p_1p_2}(t_1,\ t_2)$. Using

(6.11) we can get from (7.17) a Mamdani fuzzy system with two input variables:

$$M_{a,m}(t_1,\, t_2) = \frac{\sum\limits_{p_1,p_2=0}^{m}\Big(H_{p_1p_2}(t_1,t_2)\cdot r(p_1,p_2)\Big)}{\sum\limits_{p_1,p_2=0}^{m} H_{p_1p_2}(t_1,t_2)},\ (t_1,t_2\in\mathbb{R}_+), \tag{7.18}$$

where let $0/0 \equiv 0$. Similarly with (7.12), the Mamdani fuzzy system can expressed as

$$M_{a,m}(t_1,\, t_2) = \sum_{p_1,p_2=0}^{m} H_{p_1p_2}(t_1,t_2)\cdot r(p_1,p_2), \quad t_1,t_2\in\mathbb{R}_+. \tag{7.19}$$

With the same reason we can establish one dimensional Mamdani fuzzy system:

$$M_{a,m}(t) = \sum_{p=0}^{m} H_p(t)\cdot O(p), \ \ t\in\mathbb{R}_+. \tag{7.20}$$

If in (7.19) (7.20), $r(\cdot,\,\cdot)$ and $O(\cdot)$ are random functions, i.e. for each p, p_1, $p_2 \in \{0,1,...,m\}$, the adjustable parameters $O(p)$, $r(p_1,p_2)$ are random variables, then the corresponding I/O relationships are called stochastic Mamdain fuzzy systems.

Similarly with Theorem 7.2, the stochastic integral of $M_{a,m}(\cdot,\,\cdot)$ can be also represented as the sum of some one dimensional stochastic Mamdani fuzzy systems.

Theorem 7.3 *For any $a>0$, and $m\in\mathbb{N}$, let Mamdani fuzzy system $M_{a,m}(\cdot,\,\cdot)$ is defined as (7.19). Suppose $\gamma=\{\gamma(\theta),\,\theta\in\mathbb{R}_+\}$ is a stochastic process with orthogonal increments, Then it follows that*

(i) For any $t\in\mathbb{R}_+$, the stochastic integral $\int_{\mathbb{R}_+} M_{a,m}(t,\,\theta)\mathrm{d}\gamma(\theta)$ exists;

(ii) $\forall\varepsilon>0$, there exist $\theta_1,...,\theta_q$ independently of t, and $0=\theta_0<\theta_1<\cdots<\theta_q$, satisfying

$$\forall t\in\mathbb{R}_+,\ E\bigg(\Big|\int_{\mathbb{R}_+} M_{a,m}(t,\theta)\mathrm{d}\gamma(\theta) - \sum_{j=1}^{q} M_{a,m}(t,\theta_j)\Delta\gamma_j\Big|^2\bigg)^{\frac{1}{2}} < \varepsilon,$$

where $\Delta\gamma_j=\gamma(\theta_j)-\gamma(\theta_{j-1})$ $(j=1,...,q)$.

Proof. By (7.19) it follows that $\theta>a$, $\Longrightarrow M_{a,m}(t,\,\theta)\equiv 0$, and when $t\le a$, we have

$$\begin{aligned} M_{a,m}(t,\,\theta) &= \sum_{p_1,p_2=0}^{m} H_{p_1p_2}(t,\,\theta)\cdot r(p_1,p_2) \\ &= \sum_{p_1=0}^{m} \widetilde{A}_{1p_1}(t)\Big(\sum_{p_2=0}^{m} \widetilde{A}_{2p_2}(\theta)\cdot r(p_1,\,p_2)\Big). \end{aligned} \tag{7.21}$$

Since $\widetilde{A}_{ip_i}$ $(i = 1, 2)$ is Riemann integrable we imply, the stochastic integral $\int_0^a M_{a,m}(t,\,\theta)\mathrm{d}\gamma(\theta)$ exist, hence $\int_{\mathbb{R}_+} M_{a,m}(t,\,\theta)\mathrm{d}\gamma(\theta)$ also exists. So we can obtain (i). Moreover

$$\int_{\mathbb{R}_+} M_{a,m}(t,\theta)\mathrm{d}\gamma(\theta) = \sum_{p_1=0}^{m} \widetilde{A}_{1p_1}(t)\Big(\sum_{p_2=0}^{m} r(p_1,p_2)\cdot\int_0^a \widetilde{A}_{2p_2}(\theta)\mathrm{d}\gamma(\theta)\Big). \quad (7.22)$$

By the definition of stochastic integral [15, 30], there exist $\theta_1, ..., \theta_q \in \mathbb{R}_+ :\ 0 = \theta_0 < \theta_1 < \cdots < \theta_q$, independently of t. For any $p_2 \in \{0, 1, ..., m\}$, it follows that

$$E\Big(\Big|\int_0^a \widetilde{A}_{2p_2}(\theta)\mathrm{d}\gamma(\theta) - \sum_{j=1}^{q} \widetilde{A}_{2p_2}(\theta_j)\Delta\gamma_j\Big|^2\Big)^{\frac{1}{2}} < \frac{\varepsilon\cdot\sqrt{(m+1)^{-1}}}{\sqrt{\sum\limits_{p_1,p_2=0}^{m}(r(p_1,p_2))^2}}. \quad (7.23)$$

Obviously $\forall t > a,\ M_{a,m}(t,\theta) \equiv 0$; when $t \le a$, we have

$$\begin{aligned}&\int_{\mathbb{R}_+} M_{a,m}(t,\,\theta)\mathrm{d}\gamma(\theta) - \sum_{j=1}^{q} M_{a,m}(t,\,\theta_j)\Delta\gamma_j\\ &= \sum_{p_1=0}^{m} \widetilde{A}_{1p_1}(t)\Big(\sum_{p_2=0}^{m} r(p_1,p_2)\cdot\Big[\int_0^a \widetilde{A}_{2p_2}(\theta)\mathrm{d}\gamma(\theta) - \sum_{j=1}^{q}\widetilde{A}_{2p_2}(\theta_j)\Delta\gamma_j\Big]\Big).\end{aligned} \quad (7.24)$$

Since $\sum\limits_{p_1=0}^{m}[\widetilde{A}_{1p_1}(t)]^2 \le \sum\limits_{p_1=0}^{m}\widetilde{A}_{1p_1}(t) = 1$ $(t \in [0,\,a])$, by (7.21)–(7.24) we get

$$\begin{aligned}&\Big\{E\Big(\Big|\int_{\mathbb{R}_+} M_{a,m}(t,\theta)\mathrm{d}\gamma(\theta) - \sum_{j=1}^{q} M_{a,m}(t,\theta_j)\Delta\gamma_j\Big|^2\Big)\Big\}^{\frac{1}{2}}\\ &\le \Big\{E\Big(\sum_{p_1=0}^{m}\Big|\sum_{p_2=0}^{m} r(p_1,\,p_2)\Big[\int_0^a \widetilde{A}_{2p_2}(\theta)\mathrm{d}\gamma(\theta) - \sum_{j=1}^{q}\widetilde{A}_{2p_2}(\theta_j)\Delta\gamma_j\Big]\Big|^2\Big)\Big\}^{\frac{1}{2}}\\ &\le \Big\{\sum_{p_1,p_2=0}^{m}(r(p_1,\,p_2))^2\sum_{p=0}^{m}E\Big[\Big|\int_0^a \widetilde{A}_{2p}(\theta)\mathrm{d}\gamma(\theta) - \sum_{j=1}^{q}\widetilde{A}_{2p}(\theta_j)\Delta\gamma_j\Big|^2\Big]\Big\}^{\frac{1}{2}}\\ &\le \Big\{\sum_{p_1=0}^{m}\Big(\sum_{p_2=0}^{m}(r(p_1,p_2))^2\cdot\sum_{p_2=0}^{m}\frac{\varepsilon^2}{\cdot(m+1)\cdot\sum\limits_{p_1,p_2=0}^{m}(r(p_1,p_2))^2}\Big)\Big\}^{\frac{1}{2}}\\ &\le \sqrt{\sum_{p_1,p_2=0}^{m}(r(p_1,p_2))^2}\cdot\frac{\sqrt{m+1}\cdot\varepsilon}{\sqrt{(m+1)\cdot\sum\limits_{p_1,p_2=0}^{m}(r(p_1,p_2))^2}} = \varepsilon,\end{aligned}$$

The theorem is therefore proved. □

In the section we introduce stochastic fuzzy systems, and establish the approximate representation of the stochastic integrals of a two dimensional stochastic T–S fuzzy system and a stochastic Mamdani fuzzy system. These results will play key roles in the research of the stochastic fuzzy systems related approximating a class of stochastic processes with the mean square sense.

§7.3 Universal approximation of stochastic process by T–S fuzzy system

To apply fuzzy systems widely a necessary and important topic is to study the approximate representations of stochastic processes by fuzzy systems. In the section we focus on the approximating capability of T–S fuzzy systems in stochastic environment under a mean square sense.

7.3.1 Uniform approximation

Suppose $x = \{x(t),\, t \in \mathbb{R}_+\}$ is a stochastic process, and $(\mathbb{R}_+,\, \mathcal{B},\, F)$ is a finite measure space. If $\forall \varepsilon > 0$, there exists a family T–S fuzzy systems $S_{a,m}(\cdot)$ defined as (7.13) so that

$$\Big\{\int_{\mathbb{R}_+} E(|x(t) - S_{a,m}(t)|^2)G(\mathrm{d}t)\Big\}^{\frac{1}{2}} < \varepsilon.$$

Then we call T–S fuzzy systems are universal approximators with mean square sense to the process x. If the trajectory of x is uniformly continuous almost everywhere (a.e.) on $\mathbb{R}_+$, we may employ the property of the process x itself to study the approximation of T–S fuzzy systems to x.

Theorem 7.4 *For any $a > 0$, suppose $x = \{x(t),\, t \in [0, a]\}$ is a stochastic process whose sample trajectory is uniformly continuous a.e. on $[0,\, a]$. Define a finite sum as*

$$F_{a,m}(t) = \frac{\sum\limits_{p=0}^{m} [\widetilde{A}_{1p}(t)]^{\alpha}(b_{0;p} + b_{1;p}t)}{\sum\limits_{p=0}^{m} [\widetilde{A}_{1p}(t)]^{\alpha}} \tag{7.25}$$

where $0 \le \alpha \le +\infty$. Then there exist $b_{0;p},\, b_{1;p} \in L^2(\Omega)$ $(p = 0, 1, ..., , m)$, so that $F_{a,m}(t) \xrightarrow{m.s.} x(t)$ $(n \to +\infty)$ holds uniformly for $t \in [0,\, a]$.

Proof. For given $a > 0$, and for any $\varepsilon > 0$, using the assumption that the sample trajectory of x is uniformly continuous a.e. on $[0,\, a]$, (7.25) and Lemma 6.2, we can find a sufficient large $m \in \mathbb{N}$, satisfying $\forall t \in [0,\, a]$, $p \in N(t)$, $\big|x(t) - x\big(ap/m\big)\big| < \varepsilon$, a.e.. Define the random variables $b_{0;p}$, $b_{1;p}$ $(p = 0, 1, ..., m)$ respectively as follows:

$$b_{0;p}(w) = x\Big(\frac{ap}{m},\; w\Big), \quad b_{1;p}(w) = 0.$$

Obviously $b_{0,p}, b_{1,p} \in L^2(\Omega)$ $(p = 0, 1, ..., n)$. Moreover, $\forall t \in [0, a]$, the following facts hold:

$$
\begin{aligned}
&E\big(\big|x(t) - F_{a,m}(t)\big|^2\big) \\
&= E\Big\{\Big|x(t) - \sum_{p=0}^{m} [\widetilde{A}_{1p}(t)]^{\alpha}\big(b_{0;p} + b_{1;p}t\big)\Big/\Big(\sum_{p=0}^{m} [\widetilde{A}_{1p}(t)^{\alpha}]\Big)\Big|^2\Big\} \\
&= E\Big\{\Big|\sum_{p=0}^{m} [\widetilde{A}_{1p}(t)]^{\alpha}\big(x(t) - b_{0;p} - b_{1;p}t\big)\Big/\Big(\sum_{p=0}^{m} [\widetilde{A}_{1p}(t)]^{\alpha}\Big)\Big|^2\Big\} \\
&= E\Big\{\Big|\sum_{p=0}^{m} [\widetilde{A}_{1p}(t)]^{\alpha}\big(x(t) - x(\tfrac{ap}{m})\big)\Big/\Big(\sum_{p=0}^{m} [\widetilde{A}_{1p}(t)]^{\alpha}\Big)\Big|^2\Big\} \\
&\leq E\Big\{\Big|\sum_{p=0}^{m} [\widetilde{A}_{1p}(t)]^{\alpha}\big|x(t) - x(\tfrac{ap}{m})\big|\Big/\Big(\sum_{p=0}^{m} [\widetilde{A}_{1p}(t)]^{\alpha}\Big)\Big|^2\Big\} \\
&= E\Big\{\Big|\sum_{p\in N(t)} [\widetilde{A}_{1p}(t)]^{\alpha}\big|x(t) - x(\tfrac{ap}{m})\big|\Big/\Big(\sum_{p\in N(t)} [\widetilde{A}_{1p}(t)]^{\alpha}\Big)\Big|^2\Big\} \\
&< E\Big\{\Big|\sum_{p\in N(t)} [\widetilde{A}_{1p}(t)]^{\alpha}\cdot\varepsilon\Big/\Big(\sum_{p\in N(t)} [\widetilde{A}_{1p}(t)]^{\alpha}\Big)\Big|^2\Big\} = \varepsilon^2.
\end{aligned}
$$

Therefore $\lim\limits_{m\to+\infty} E(|F_{a,m}(t) - x(t)|^2) = 0$ holds uniformly for $t \in [0, a]$. The theorem is proved. □

In (7.25) if let $\alpha = 1$, then by $\sum\limits_{p=0}^{m} \widetilde{A}_{1p}(t) \equiv 1$, we get, $F_{a,m}(t) = S_{a,m}(t)$ $(t \in [0, a])$. Thus, (7.25) is an one dimensional T–S fuzzy system. So the following conclusion is obvious.

Corollary 7.1 *Suppose $x = \{x(t),\ t \in \mathbb{R}_+\}$ is a stochastic process with sample trajectory being uniformly continuous on $\mathbb{R}_+$ a.e., and when $t \notin [0, a]$, the random variable $x(t)$ is zero a.e.. Then T–S fuzzy systems are universal approximators to x, that is, for any $\varepsilon > 0$, there is a T–S fuzzy system $S_{a,m}$, so that $\big(\int_{\mathbb{R}_+} E\big(\big|x(t) - S_{a,m}(t)\big|^2\big)G(\mathrm{d}t)\big)^{\frac{1}{2}} < \varepsilon$.*

Proof. Given arbitrarily $\varepsilon > 0$, since $0 < G(\mathbb{R}_+) \leq 1$, let $\varepsilon' = \varepsilon/(2G(\mathbb{R}_+))$. By Theorem 7.2 it follows that there is $m \in \mathbb{N}$ satisfying, $E\big(\big|S_{a,m}(t) - x(t)\big|^2\big) < \varepsilon'$ for all $t \in [0, a]$. So using the fact: $\forall t \notin [0, a]$, $x(t) = 0$ a.e., we get

$$
\begin{aligned}
\int_{\mathbb{R}_+} E\big(\big|S_{n,m}(t) - x(t)\big|^2\big)G(\mathrm{d}t) &= \int_0^a E\big(\big|S_{a,m}(t) - x(t)\big|^2\big)G(\mathrm{d}t) \\
&\leq \int_{\mathbb{R}_+} \varepsilon' G(\mathrm{d}t) = G(\mathbb{R}_+)\cdot\varepsilon' \leq \frac{\varepsilon}{2} < \varepsilon,
\end{aligned}
$$

The corollary is therefore proved. □

7.3.2 Approximation with mean square sense

If the sample trajectory of the stochastic process $x = \{x(t),\, t \in \mathbb{R}_+\}$ is not uniformly continuous on $\mathbb{R}_+$, we may analyze the universal approximation of stochastic T–S fuzzy systems under the mean square sense.

Theorem 7.5 *Suppose $x \in \mathcal{C}(\Omega)$. Then for arbitrary $\varepsilon > 0$, there is an one dimensional T–S fuzzy system $S_{a,m}(\cdot)$ defined as (7.12), such that $\left\{\int_{\mathbb{R}_+} E\left(\left|x(t) - S_{a,m}(t)\right|^2\right)G(\mathrm{d}t)\right\}^{\frac{1}{2}} < \varepsilon$, that is, T–S fuzzy systems are universal approximators of each process in $\mathcal{C}(\Omega)$.*

Proof. By the assumption and (7.7) we can get the spectrum representation of the process x : $\forall t \in \mathbb{R}_+$, it follows that

$$x(t) = \int_0^{+\infty} \psi_1(t,\,\theta)\mathrm{d}\gamma_1(\theta) + \psi_2(t,\,\theta)\mathrm{d}\gamma_2(\theta) + \cdots \triangleq \int_0^{+\infty} \left\langle \Psi^{\mathrm{T}}(t,\,\theta),\, \mathrm{d}\Gamma(\theta)\right\rangle,$$

where $\Psi^{\mathrm{T}}(t,\,\theta) = \left(\psi_1(t,\theta),\, \psi_2(t,\theta), ...\right)$, $\mathrm{d}\Gamma(\theta) = \left(\mathrm{d}\gamma_1(\theta),\, \mathrm{d}\gamma_2(\theta), ...\right)^{\mathrm{T}}$. Moreover, $\gamma_1 = \{\gamma_1(\theta),\ \theta \in \mathbb{R}_+\}$, $\gamma_2 = \{\gamma_2(\theta),\, \theta \in \mathbb{R}_+\}, ...$ are orthogonal increment process with the condition (7.8). By Theorem 7.1 it follows that the T–S fuzzy system $S_{a,m}(\cdot,\,\cdot)$ is universal approximator of each function in $L^2(\mathbb{R}_+^2,\, \mathcal{B} \times \mathcal{B},\, G \times F)$. So there exist constant vectors as

$$A_{0;p_1p_2}^{\mathrm{T}} = \left(b_{0;p_1p_2}^1,\, b_{0;p_1p_2}^2, ...\right),\quad A_{1;p_1p_2}^{\mathrm{T}} = \left(b_{1;p_1p_2}^1,\, b_{1;p_1p_2}^2, ...\right),$$
$$A_{2;p_1p_2}^{\mathrm{T}} = \left(b_{2;p_1p_2}^1,\, b_{2;p_1p_2}^2, ...\right),$$

where $p_1,\, p_2 = 0, 1, ..., m$, such that

$$\int_{\mathbb{R}_+^2}\left\|\Psi^{\mathrm{T}}(t,\,\theta) - \sum_{p_1,p_2=0}^{m} H_{p_1p_2}(t,\,\theta)\left(A_{0;p_1p_2}^{\mathrm{T}} + A_{1;p_1p_2}^{\mathrm{T}}t + A_{2;p_1p_2}^{\mathrm{T}}\theta\right)\right\|^2 F(\mathrm{d}\theta)G(\mathrm{d}t) < \frac{\varepsilon^2}{2}. \tag{7.26}$$

Moreover, $\int_{t>a}\int_{\theta>a}\left\|\Psi^{\mathrm{T}}(t,\,\theta)\right\|^2 G(\mathrm{d}t)F(\mathrm{d}\theta) < \varepsilon^2/4$. Denote

$$G_m^{\mathrm{T}}(t,\,\theta) = \sum_{p_1,p_2=0}^{m} H_{p_1p_2}(t,\,\theta)\left(A_{0;p_1p_2}^{\mathrm{T}} + A_{1;p_1p_2}^{\mathrm{T}}t + A_{2;p_1p_2}^{\mathrm{T}}\theta\right),$$

Therefore we may conclude that

$$\int_{\mathbb{R}_+}\int_{\mathbb{R}_+}\left\|\Psi^{\mathrm{T}}(t,\theta) - G_m^{\mathrm{T}}(t,\theta)\right\|^2 F(\mathrm{d}\theta)G(\mathrm{d}t) < \frac{\varepsilon^2}{4}. \tag{7.27}$$

Using Theorem 7.2 easily we can show that the stochastic integral $\sigma_m(t) \triangleq$

$\int_0^{+\infty}\langle G_m^{\mathrm{T}}(t,\theta), \mathrm{d}\Gamma(\theta)\rangle$ exists. By (7.4) it follows that

$$\begin{aligned}&\int_{\mathbb{R}_+} E\Big|\int_{\mathbb{R}_+}\langle(\Psi^{\mathrm{T}}(t,\,\theta)-G_m^{\mathrm{T}}(t,\,\theta)),\ \mathrm{d}\Gamma(\theta)\rangle\Big|^2 G(\mathrm{d}t)\\ &=\int_{\mathbb{R}_+^2}\|\Psi^{\mathrm{T}}(t,\,\theta)-G_m^{\mathrm{T}}(t,\,\theta)\|^2F(\mathrm{d}\theta)G(\mathrm{d}t)<\frac{\varepsilon^2}{4}.\end{aligned}\tag{7.28}$$

Thus, using the canonical representation of x and (7.28) we can show

$$\begin{aligned}&\int_{\mathbb{R}_+} E(|x(t)-\sigma_m(t)|^2)G(\mathrm{d}t)\\ &=\int_{\mathbb{R}_+} E\Big(\Big|\int_{\mathbb{R}_+}\langle(\Psi^{\mathrm{T}}(t,\,\theta)-G_m^{\mathrm{T}}(t,\,\theta)),\ \mathrm{d}\Gamma(\theta)\rangle\Big|^2\Big)G(\mathrm{d}t)<\frac{\varepsilon^2}{4}.\end{aligned}\tag{7.29}$$

By Theorem 7.2, for $\varepsilon>0$, there exist $\theta_1,\,\theta_2,...,\theta_q:\ \theta_1<\theta_2<\cdots<\theta_q$, which are independent of t, so that

$$E\Big(\Big|\int_0^{+\infty}\langle G_m^{\mathrm{T}}(t,\,\theta),\mathrm{d}\Gamma(\theta)\rangle-\sum_{j=1}^{q}\langle G_m^{\mathrm{T}}(t,\,\theta_j),\ \Delta\Gamma_j\rangle\Big|^2\Big)<\frac{\varepsilon^2}{4},\tag{7.30}$$

where $\Delta\Gamma_j=\big(\gamma_1(\theta_j)-\gamma_1(\theta_{j-1}),\ \gamma_2(\theta_j)-\gamma_2(\theta_{j-1}),...\big)^{\mathrm{T}}$ $(j=1,...,q;\theta_0=0)$. Let

$$S_{a,m}(t)=\sum_{j=1}^{q}\langle G_m^{\mathrm{T}}(t,\theta_j),\Delta\Gamma_j\rangle\ (t\in\mathbb{R}_+).$$

It is easy to show that $S_{a,m}(\cdot)$ is an one dimensional stochastic T–S fuzzy system, which is also represented as follows:

$$\begin{aligned}S_{a,m}(t)\ &=\ \sum_{j=1}^{q}\Big\langle\sum_{p_1,p_2=0}^{m}H_{p_1p_2}(t,\theta_j)\big(A_{0;p_1p_2}^{\mathrm{T}}+A_{1;p_1p_2}^{\mathrm{T}}t+A_{2;p_1p_2}^{\mathrm{T}}\big),\ \Delta\Gamma_j\Big\rangle\\ &=\ \sum_{p=0}^{m}\widetilde{A}_{1p}(t)\Big\{\sum_{j=1}^{q}\sum_{p_2=0}^{m}\widetilde{A}_{2p_2}(\theta_j)\Big(\langle A_{0;pp_2}^{\mathrm{T}}+\theta_jA_{2;pp_2}^{\mathrm{T}},\ \Delta\Gamma_j\rangle\\ &\qquad\qquad+t\cdot\langle A_{1;pp_1}^{\mathrm{T}},\ \Delta\Gamma_j\rangle\Big)\Big\}.\end{aligned}$$

For $p=0,1,...,n$, define the random variable parameters $b_{0;p},\ b_{1;p}$ as

$$\begin{cases}b_{0,p}=\sum\limits_{j=1}^{q}\sum\limits_{p_2=0}^{m}\widetilde{A}_{2p_2}(\theta_j)\Big(\big(b_{0;pp_2}^1+\theta_jb_{2;pp_2}^1\big)\big(\gamma_1(\theta_j)-\gamma_1(\theta_{j-1})\big)\\ \qquad\qquad+\big(b_{0;pp_2}^2+\theta_jb_{2;pp_2}^2\big)\big(\gamma_2(\theta_j)-\gamma_2(\theta_{j-1})\big)+\cdots\Big)\\ b_{1,p}=\sum\limits_{j=1}^{q}\sum\limits_{p_2=0}^{m}\widetilde{A}_{2p_2}(\theta_j)\Big(b_{1;pp_2}^1\big(\gamma_1(\theta_j)-\gamma_1(\theta_{j-1})\big)\\ \qquad\qquad+b_{1;pp_2}^2\big(\gamma_2(\theta_j)-\gamma_2(\theta_{j-1})\big)+\cdots\Big)\end{cases}\tag{7.31}$$

Therefore, $S_{a,m}(t) = \sum_{p=0}^{m} \widetilde{A}_{1p}(t)(b_{0;p} + b_{1;p}t)$ $(t \in \mathbb{R}_+)$. So by (7.30) we have

$$\begin{aligned} &\int_{\mathbb{R}_+} E\Big(\Big|\sigma_m(t) - \sum_{p=0}^{m} \widetilde{A}_{1p}(t)(b_{0,p} + b_{1,p}t)\Big|^2\Big) G(\mathrm{d}t) \\ &= \int_{\mathbb{R}_+} E(|\sigma_m(t) - S_{a,m}(t)|^2) G(\mathrm{d}t) < \frac{\varepsilon^2}{4}. \end{aligned} \tag{7.32}$$

Moreover, by (7.29) (7.32) and the triangle inequality for metric:

$$\begin{aligned} &\Big(\int_{\mathbb{R}_+} E(|x(t) - S_{a,m}(t)|^2) G(\mathrm{d}t)\Big)^{\frac{1}{2}} \\ &= \Big(\int_{\mathbb{R}_+} E\big(|x(t) - \sigma_m(t) + \sigma_m(t) - S_{a,m}(t)\big|^2 G(\mathrm{d}t)\Big)^{\frac{1}{2}} \\ &\le \Big(\int_{\mathbb{R}_+} E\big(|x(t) - \sigma_m(t)\big|^2 G(\mathrm{d}t)\Big)^{\frac{1}{2}} + \Big(E\big(|\sigma_m(t) - S_{a,m}(t)\big|^2 G(\mathrm{d}t)\Big)^{\frac{1}{2}}, \end{aligned}$$

we imply, $\big(\int_{\mathbb{R}_+} E(|x(t) - S_{a,m}(t)|^2) G(\mathrm{d}t)\big)^{\frac{1}{2}} < \varepsilon/2 + \varepsilon/2 = \varepsilon$. The theorem is therefore proved. □

We transform $\{\Gamma(\theta),\, \theta \in \mathbb{R}_+\}$ into a vector valued Brownian motion $B = (b_1, b_2, \dots)$, where b_i $(i = 1, 2, \dots)$ is a standard Brownian motion. Then by (7.10) (7.13) we can establish an one dimensional stochastic T–S system:

$$S_{a,m}(t) = \sum_{p=0}^{m} \widetilde{A}_{1p}(t)(b_{0;p} + b_{1;p}t) \ (t \in \mathbb{R}_+),$$

and $b_{0;p}$, $b_{1;p}$ can be established by the following analytic learning algorithm:

$$\begin{cases} b_{0,p} = \sum\limits_{j=1}^{q} \sum\limits_{p_2=0}^{m} \Big(\widetilde{A}_{2p_2}(\theta_j)\Big((b^1_{0;pp_2} + \theta_j b^1_{2;pp_2})(b_1(\theta_j) - b_1(\theta_{j-1})) + \\ \qquad\qquad (b^2_{0;pp_2} + \theta_j b^2_{2;pp_2})(b_2(\theta_j) - b_2(\theta_{j-1})) + \cdots\Big); \\ b_{1,p} = \sum\limits_{j=1}^{q} \sum\limits_{p_2=0}^{m} \widetilde{A}_{2p_2}(\theta_j)\Big(b^1_{1;pp_2}(b_1(\theta_j) - b_1(\theta_{j-1})) \\ \qquad\qquad + b^2_{1;pp_2}(b_2(\theta_j) - b_2(\theta_{j-1})) + \cdots\Big). \end{cases} \tag{7.33}$$

Moreover, $E\big(|b_i(\theta_j) - b_i(\theta_{j-1})|^2\big) = |\theta_j - \theta_{j-1}|$ $(i = 1, 2, \dots; j = 1, \dots, q)$.

In (6.8) we choose $c_0 = 2$, and the error bound $\varepsilon = 0.2$, moreover, $\widetilde{A}_{ij}$ $(i = 1, 2;\ j = 0, 1, \dots, m)$ is defined by a translation of the triangular fuzzy number

$\widetilde{A}(\cdot)$ defined as

$$\widetilde{A}(t)=\begin{cases}\dfrac{m}{a}t+1, & -\dfrac{a}{m}\le t\le 0,\\ 1-\dfrac{m}{a}t, & 0<t\le\dfrac{a}{m},\\ 0, & \text{otherwise};\end{cases}$$

$$\widetilde{A}_{1(0)}(t)=\begin{cases}\widetilde{A}\ (t), & t\ge 0,\\ 0, & \text{otherwise};\end{cases}\qquad \widetilde{A}_{1m}(t)=\begin{cases}\widetilde{A}\ (t-a), & t\le a,\\ 0, & t>a.\end{cases}$$

And let $\widetilde{A}_{1j}\equiv\widetilde{A}_{2j}$: $\widetilde{A}_{1j}(t)=\widetilde{A}(t-aj/m)$ $(j=1,...,m-1)$.

Example 7.1 Define the stochastic process $x=\{x(t),\ t\in\mathbb{R}_+\}$ as follows:

$$x(t)=\varphi(t)\cdot\cos(\gamma_0 t)\ (t\in\mathbb{R}_+),$$

where γ_0 is a constant, $\varphi(\cdot)$ is a weak stationary process with zero mean, whose covariance function is given by

$$B_\varphi(t,s)\triangleq B_\varphi(\tau)=\exp\{-2|\tau|\},\ \tau=t-s.$$

In practice, $x=\{x(t),\ t\in\mathbb{R}_+\}$ may represent the well–known stochastic telegraph signal. By [15] we can get the relationship between the covariance function $B_\varphi(\cdot,\cdot)$ of φ and its spectral density function $f(\cdot)$:

$$f(\theta)=\frac{1}{\pi}\int_{-\infty}^{+\infty}\exp(-I\theta\tau)B_\varphi(\tau)\mathrm{d}\tau,\quad B_\varphi(\tau)=\frac{1}{2}\int_{-\infty}^{+\infty}\exp(I\theta\tau)f(\theta)\mathrm{d}\theta, \tag{7.34}$$

where $I^2=-1$. By (7.34) it follows that

$$f(\theta)=\frac{4}{\pi}\frac{1}{\theta^2+4},\Longrightarrow F(\mathrm{d}\theta)=f(\theta)\mathrm{d}\theta=\frac{4}{\pi}\frac{1}{\theta^2+4}\mathrm{d}\theta.$$

Since $B_x(t,s)=E(\varphi(t)\cdot\varphi(s))\cdot\cos(\gamma_0 t)\cdot\cos(\gamma_0 s)$, we get

$$\begin{aligned}B_x(t,s)&=\frac{1}{2}\int_{-\infty}^{+\infty}\exp\big(I\tau\theta\big)\cdot\frac{1}{\pi}\frac{4}{\theta^2+4}\mathrm{d}\theta\cdot\cos(\gamma_0 t)\cdot\cos(\gamma_0 s)\\ &=\int_0^{+\infty}\frac{4\big(\cos(\theta t)\cos(\theta s)+\sin(\theta t)\sin(\theta s)\big)}{\pi\cdot(\theta^2+4)}\cdot\cos(\gamma_0 t)\cdot\cos(\gamma_0 s)\mathrm{d}\theta\\ &\triangleq\int_0^{+\infty}\big\langle\Psi^{\mathrm{T}}(t,\theta),\ \Psi(s,\theta)\big\rangle F(\mathrm{d}\theta),\end{aligned}$$

where $\Psi^{\mathrm{T}}(t,\theta)=\big(\cos(\theta t)\cos(\gamma_0 t),\ \sin(\theta t)\cos(\gamma_0 t)\big)$. Consequently by (7.8) it follows that

$$\Phi^{\mathrm{T}}(t,\theta)=C(\theta)\cdot\Psi^{\mathrm{T}}(t,\theta);\ M(\mathrm{d}\theta)=\mathrm{d}\theta=\frac{1}{C(\theta)^2}\cdot f(\theta)\mathrm{d}\theta.$$

Thus, $C(\theta) = 2/\sqrt{\pi(\theta^2+4)}$. Hence

$$\begin{aligned}\Phi^{\mathrm{T}}(t,\theta) &= \frac{2}{\sqrt{\pi(\theta^2+4)}}\Big(\cos(\theta t)\cos(\gamma_0 t),\ \sin(\theta t)\cos(\gamma_0 t)\Big) \\ &\triangleq \big(\varphi_1(t,\theta),\ \varphi_2(t,\theta)\big).\end{aligned}$$

where $\varphi_1(t,\ \theta)$, $\varphi_2(t,\ \theta)$ can be expressed as

$$\varphi_1(t,\ \theta) = \frac{2\cdot\cos(\theta t)\cos(\gamma_0 t)}{\sqrt{\pi(\theta^2+4)}},\ \ \varphi_2(t,\ \theta) = \frac{2\cdot\sin(\theta t)\cos(\gamma_0 t)}{\sqrt{\pi(\theta^2+4)}}.$$

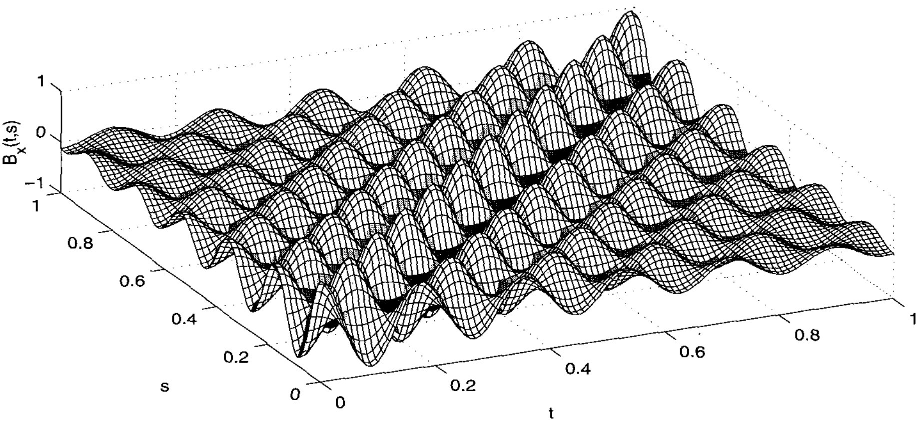


Figure 7.1 The surface of B_x

Choose $\gamma_0 = 35$. Easily we can show, $\|\Phi(\cdot,\ \cdot)\| \in L^2(\mathbb{R}_+^2,\ \mathcal{B}\times\mathcal{B},\ G\times M)$, moreover

$$\int_{t>1,\theta>1}\Big(|\varphi_1(t,\theta)|^2 + |\varphi_2(t,\theta)|^2\Big)M(\mathrm{d}\theta)G(\mathrm{d}t) < 0.005 = \frac{\varepsilon^2}{8}.$$

Hence $a = 1$. By Theorem 7.1, we can get by calculating

$$D_H(\Phi) \triangleq D_H(\varphi_1) \vee D_H(\varphi_2) \le 20.13.$$

Similarly we use Theorem 7.1 to estimate m :

$$m \ge \frac{4\times 2\times 1\times 20.13}{0.2}\cdot\sqrt{(G\times M)([0,\ 1]^2)} \ge 257.$$

So choose $m = 257$, and $A^{\mathrm{T}}_{1;p_1p_2} = A^{\mathrm{T}}_{2;p_1p_2} = (0,\ 0)$, moreover

$$b^1_{0;p_1p_2} = \varphi_1\Big(\frac{p_1}{257},\ \frac{p_2}{257}\Big),\ \ b^2_{0;p_1p_2} = \varphi_2\Big(\frac{p_1}{257},\ \frac{p_2}{257}\Big)\ (p_1,p_2 = 0,1,...,257).$$

It is easy to show that

$$\int_{\mathbb{R}_+}\int_{\mathbb{R}_+}\Big\|\Phi^{\mathrm{T}}(t,\theta)-\sum_{p=0}^{257}\widetilde{A}_{1p}(t)\cdot A_{0;p}^{\mathrm{T}}\Big\|^2 F(\mathrm{d}\theta)G(\mathrm{d}t)<\frac{\varepsilon^2}{4}=0.01.$$

In Theorem 7.10 we choose $\theta_j=j/10\ (j=0,1,...,10)$. Then

$$E\Big(\Big|S_{a,m}(t,\,\theta)-\sum_{j=1}^{10}S_{a,m}(t,\,\theta_j)\Delta b_j\Big|^2\Big)<\varepsilon^2/4=0.01.$$

Therefore, for $p=0,1,...,257$, by (7.33) we get $b_{1;p}=0$, and

$$b_{0;p}=\sum_{j=1}^{10}\sum_{p_2=0}^{257}\widetilde{A}_{2p_2}\Big(\frac{j}{10}\Big)\Big\{\varphi_1\Big(\frac{p}{257},\frac{p_2}{257}\Big)\Big[b_1\Big(\frac{j}{10}\Big)-b_1\Big(\frac{j-1}{10}\Big)\Big] +\varphi_2\Big(\frac{p}{257},\frac{p_2}{257}\Big)\Big[b_2\Big(\frac{j}{10}\Big)-b_2\Big(\frac{j-1}{10}\Big)\Big]\Big\}.$$

Thus, the one dimensional stochastic T–S fuzzy system $S_{a,m}$ can be expressed as

$$S_{a,m}(t)=\sum_{p,p_2=0}^{257}\sum_{j=1}^{10}\widetilde{A}_{1p}(t)\cdot\widetilde{A}_{2p_2}\Big(\frac{j}{10}\Big)\Big\{\varphi_1\Big(\frac{p}{257},\frac{p_2}{257}\Big)\Big[b_1\Big(\frac{j}{10}\Big)-b_1\Big(\frac{j-1}{10}\Big)\Big] +\varphi_2\Big(\frac{p}{257},\frac{p_2}{257}\Big)\Big[b_2\Big(\frac{j}{10}\Big)-b_2\Big(\frac{j-1}{10}\Big)\Big]\Big\}.$$

So we can conclude that

$$E\Big(\Big[b_p\Big(\frac{j_1}{10}\Big)-b_p\Big(\frac{j_1-1}{10}\Big)\Big]\Big[b_q\Big(\frac{j_2}{10}\Big)-b_q\Big(\frac{j_2-1}{10}\Big)\Big]\Big) =\begin{cases}\Big|\frac{j}{10}-\frac{j-1}{10}\Big|=\frac{1}{10}, & j_1=j_2, p=q;\\ 0, & \text{otherwise},\end{cases}$$

and $B_{S_{a,m}}(t,s)=E\big(S_{a,m}(t)\cdot S_{a,m}(s)\big)$. Thus

$$B_{S_{a,m}}(t,s)=\frac{1}{10}\sum_{p_1,p_2,p_3,p_4=0}^{257}\sum_{j=1}^{10}\widetilde{A}_{1p_1}(t)\cdot\widetilde{A}_{1p_3}(s)\cdot\widetilde{A}_{2p_2}\Big(\frac{j}{10}\Big)\cdot\widetilde{A}_{2p_4}\Big(\frac{j}{10}\Big)\cdot \cdot\Big(\varphi_1\Big(\frac{p_1}{257},\frac{p_2}{257}\Big)\cdot\varphi_1\Big(\frac{p_3}{257},\frac{p_4}{257}\Big)+\varphi_2\Big(\frac{p_1}{257},\frac{p_2}{257}\Big)\cdot\varphi_2\Big(\frac{p_3}{257},\frac{p_4}{257}\Big)\Big).$$

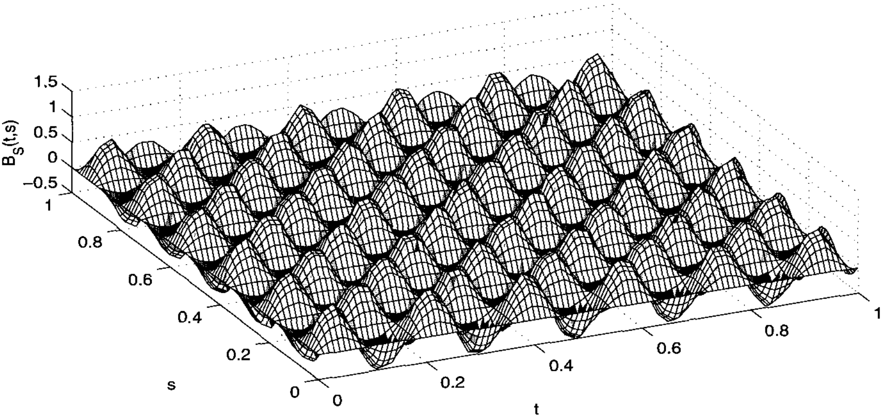


Figure 7.2 The surface of $B_{S_{a,m}}$ when $m = 40$

For simplicity of computation, we choose $m = 40$ to derive the surface of $B_{M_{a,m}}(\cdot, \cdot)$ as shown in Figure 7.2. And Figure 7.1 is the surface of $B_x(\cdot, \cdot)$. From the comparison between Figure 7.1 and Figure 7.2, we may see the fact the accuracy ε in mean square sense is guaranteed.

The section generalizes approximation analysis related to T–S fuzzy systems from deterministic I/O relationships to stochastic ones. That is, T–S fuzzy systems with the multiplication '$\cdot$' norm can with arbitrary degree of accuracy approximate a class of stochastic processes. Thus, the application fields of T–S fuzzy systems may be extended, strikingly. Then, if the fuzzy systems related are Mamdani systems, whether how can we address the corresponding problems? This is one main topic in the following section to study.

§7.4 Universal approximation of stochastic Mamdani fuzzy system

In the section we extend results for Mamdani fuzzy systems in approximating deterministic nonlinear I/O relationships, to the case of stochastic processes in which the sample functions occur randomly [22, 30]. Also the fuzzy operator composition operation '$\vee - \times$' is employed to define the Mamdani fuzzy systems related. A learning algorithm for realizing the approximating procedure is developed.

7.4.1 Approximation of stochastic Mamdani fuzzy system

Assume that $(\mathbb{R}_+, \mathcal{B}, G)$ is a finite measure space. If for any stochastic process $x = \{x(t),\, t \in \mathbb{R}_+\} \in \mathcal{C}(\Omega)$, and $\forall \varepsilon > 0$, there is an one-dimensional stochastic Mamdani fuzzy system $M_{a,m}(\cdot)$ defined as (7.20), so that the following estimation holds: $\left\{\int_{\mathbb{R}_+} E(|x(t) - M_{a,m}(t)|^2)G(\mathrm{d}t)\right\}^{\frac{1}{2}} < \varepsilon$. Then stochastic

Mamdani systems are said to be universal approximators of $\mathcal{C}(\Omega)$.

Theorem 7.6 *Let* $x = \{x(t),\, t \in \mathbb{R}_+\} \in \mathcal{C}(\Omega)$, *and* $(\mathbb{R}_+,\, \mathcal{B},\, G)$ *be a finite measure space. Then for any* $\varepsilon > 0$, *there is one-dimensional stochastic Mamdani system* $M_{a,m}(\cdot)$ *so that*

$$\Big\{\int_{\mathbb{R}_+} E\big(\big|x(t) - M_{a,m}(t)\big|^2\big)G(\mathrm{d}t)\Big\}^{\frac{1}{2}} < \varepsilon,$$

that is, stochastic Mamdani fuzzy systems are universal approximators of $\mathcal{C}(\Omega)$.

Proof. By the assumption and (7.7) we can obtain the following canonical representation of the process $x:\ \forall t \in \mathbb{R}_+$, it follows that

$$x(t) = \int_0^{+\infty} \varphi_1(t,\,\theta)\mathrm{d}\gamma_1(\theta) + \varphi_2(t,\,\theta)\mathrm{d}\gamma_2(\theta) + \cdots \triangleq \int_0^{+\infty} \big\langle \Phi^{\mathrm{T}}(t,\,\theta),\ \mathrm{d}\Gamma(\theta)\big\rangle,$$

where

$$\Phi^{\mathrm{T}}(t,\,\theta) = \big(\varphi_1(t,\,\theta),\ \varphi_2(t,\,\theta),...\big),\ \ \mathrm{d}\Gamma(\theta) = \big(\mathrm{d}\gamma_1(\theta),\ \mathrm{d}\gamma_2(\theta),...\big)^{\mathrm{T}}.$$

Furthermore, $\gamma_1 = \{\gamma_1(\theta),\, \theta \in \mathbb{R}_+\}$, $\gamma_2 = \{\gamma_2(\theta),\, \theta \in \mathbb{R}_+\},...$ are the orthogonal increment processes with the condition (7.8). Using Theorem 7.1 we get, Mamdani fuzzy systems are universal approximators to each function of $L^2(\mathbb{R}_+^2,\, \mathcal{B}\times\mathcal{B},\, G\times F)$. There exist mapping $r_i:\ \mathbb{R}_+^2 \longrightarrow \mathbb{R}$, and $m \in \mathbb{N}$, so that if let $R^{\mathrm{T}}(p_1,\, p_2) = \big(r_1(p_1,\, p_2),\ r_2(p_1,\, p_2),\dots\big)$ for $p_1, p_2 = 0, 1, ..., m$, we can conclude that

$$\int_{\mathbb{R}_+}\int_{\mathbb{R}_+}\Big\|\Phi^{\mathrm{T}}(t,\,\theta) - \sum_{p_1,p_2=0}^{m} H_{p_1p_2}(t,\,\theta)\cdot R^{\mathrm{T}}(p_1,p_2)\Big\|^2 F(\mathrm{d}\theta)G(\mathrm{d}t) < \frac{\varepsilon^2}{4}. \tag{7.35}$$

Moreover $\int_{t>a}\int_{\theta>a}\big\|\Phi^{\mathrm{T}}(t,\,\theta)\big\|^2 G(\mathrm{d}t)F(\mathrm{d}\theta) < \varepsilon^2/8$. Denote

$$S_m^{\mathrm{T}}(t,\,\theta) = \sum_{p_1,p_2=0}^{m} H_{p_1p_2}(t,\,\theta)\cdot R^{\mathrm{T}}(p_1,\,p_2).$$

Rewriting (7.35) we have

$$\int_{\mathbb{R}_+}\int_{\mathbb{R}_+}\Big\|\Phi^{\mathrm{T}}(t,\,\theta) - S_m^{\mathrm{T}}(t,\,\theta)\Big\|^2 F(\mathrm{d}\theta)G(\mathrm{d}t) < \frac{\varepsilon^2}{4}. \tag{7.36}$$

By Theorem 7.5, the stochastic integral $\sigma_m(t) \triangleq \int_0^{+\infty}\big\langle S_m^{\mathrm{T}}(t,\theta),\ \mathrm{d}\Gamma(\theta)\big\rangle$ exists. Moreover by (7.4) (7.36) it follows that

$$\begin{aligned}&\int_{\mathbb{R}_+} E\Big|\int_{\mathbb{R}_+}\big\langle(\Phi^{\mathrm{T}}(t,\,\theta) - S_m^{\mathrm{T}}(t,\theta)),\ \mathrm{d}\Gamma(\theta)\big\rangle\Big|^2 G(\mathrm{d}t)\\ &= \int_{\mathbb{R}_+^2}\|\Phi^{\mathrm{T}}(t,\,\theta) - S_m^{\mathrm{T}}(t,\,\theta)\|^2 F(\mathrm{d}\theta)G(\mathrm{d}t) < \frac{\varepsilon^2}{4}.\end{aligned} \tag{7.37}$$

So using the canonical representation of x and (7.37) we get

$$\begin{aligned}&\int_{\mathbb{R}_+} E\big(\big|x(t)-\sigma_m(t)\big|^2\big)G(\mathrm{d}t)\\&=\int_{\mathbb{R}_+} E\Big(\Big|\int_{\mathbb{R}_+}\big\langle(\Phi^{\mathrm{T}}(t,\theta)-S_m^{\mathrm{T}}(t,\theta)),\ \mathrm{d}\Gamma(\theta)\big\rangle\Big|^2\Big)G(\mathrm{d}t)<\frac{\varepsilon^2}{4}.\end{aligned}\tag{7.38}$$

By Theorem 7.3, for any $\varepsilon>0$, there are $\theta_1,\theta_2,...,\theta_q:\ \theta_1<\theta_2<\cdots<\theta_q$, independently of t, satisfying

$$E\Big(\Big|\int_0^{+\infty}\big\langle S_m^{\mathrm{T}}(t,\theta),\ \mathrm{d}\Gamma(\theta)\big\rangle-\sum_{j=1}^{q}\big\langle S_m^{\mathrm{T}}(t,\theta_j),\Delta\Gamma_j\big\rangle\Big|^2\Big)<\frac{\varepsilon^2}{4},\tag{7.39}$$

where $\Delta\Gamma_j=\big(\gamma_1(\theta_j)-\gamma_1(\theta_{j-1}),\ \gamma_2(\theta_j)-\gamma_2(\theta_{j-1}),\dots\big)^{\mathrm{T}}$ $(j=1,...,q;\ \theta_0=0)$. We write

$$M_{a,m}(t)=\sum_{j=1}^{q}\big\langle S_m^{\mathrm{T}}(t,\theta_j),\ \Delta\Gamma_j\big\rangle\ (t\in\mathbb{R}_+).$$

Obviously $M_{a,m}(\cdot)$ is an one dimensional stochastic Mamdani fuzzy system, which can also expressed as

$$\begin{aligned}M_{a,m}(t)&=\sum_{j=1}^{q}\Big\langle\sum_{p_1,p_2=0}^{m}H_{p_1p_2}(t,\theta_j)R^{\mathrm{T}}(p_1p_2),\ \Delta\Gamma_j\Big\rangle\\&=\sum_{p_1=0}^{m}\widetilde{A}_{1p}(t)\Big(\sum_{j=1}^{q}\sum_{p_2=0}^{m}\widetilde{A}_{2p_2}(\theta_j)\cdot\big\langle R^{\mathrm{T}}(p_1p_2),\ \Delta\Gamma_j\big\rangle\Big).\end{aligned}$$

Hence for $p=0,1,...,m$, we let

$$\begin{aligned}O(p)=\sum_{j=1}^{q}\sum_{p_2=0}^{m}\widetilde{A}_{2p_2}(\theta_j)\Big(&r_1(p,p_2)\big(\gamma_1(\theta_j)-\gamma_1(\theta_{j-1})\big)\\&+r_2(p,p_2)\big(\gamma_2(\theta_j)-\gamma_2(\theta_{j-1})\big)+\cdots\Big).\end{aligned}\tag{7.40}$$

Therefore, $M_{a,m}(t)=\sum\limits_{p=0}^{m}\widetilde{A}_{1p}(t)\cdot O(p)$ $(t\in\mathbb{R}_+)$. Rewriting (7.39) we get

$$\begin{aligned}&\int_{\mathbb{R}_+}E\Big(\Big|\sigma_m(t)-\sum_{p=0}^{m}\widetilde{A}_{1p}(t)\cdot O(p)\Big|^2\Big)G(\mathrm{d}t)\\&=\int_{\mathbb{R}_+}E\big(\big|\sigma_m(t)-M_{a,m}(t)\big|^2\big)G(\mathrm{d}t)<\frac{\varepsilon^2}{4}.\end{aligned}\tag{7.41}$$

By the metric triangle inequality, (7.37) (7.38) and (7.41) we can conclude that the following fact holds:

$$\Big(\int_{\mathbb{R}_+}E\big(\big|x(t)-M_{a,m}(t)\big|^2\big)G(\mathrm{d}t)\Big)^{\frac{1}{2}}<\Big\{\frac{\varepsilon^2}{4}\Big\}^{\frac{1}{2}}+\Big\{\frac{\varepsilon^2}{4}\Big\}^{\frac{1}{2}}=\varepsilon.$$

Consequently we obtain

$$\Big\{E\big(|x(t)-M_{a,m}(t)|^2\big)G(\mathrm{d}t)\Big\}^{\frac{1}{2}}<\varepsilon,$$

by which the theorem is proved. □

In the proof of Theorem 7.6 we obtain an efficient algorithm (7.40), by which an one-dimensional stochastic Mamdani system can directly be constructed.

7.4.2 Example

The proposed approximating method of stochastic process by a stochastic Mamdani fuzzy system can be used in a variety of approximate realizations of stochastic processes. Here, we consider a non-stationary process, by which the stochastic telegraph signals may be described. To this end we at first let $c_0 = 2$ in (6.8), the error bound $\varepsilon = 0.2$, and antecedent fuzzy sets be identical with ones of Example 7.1. Define the stochastic process as follows:

$$\forall t\in\mathbb{R}_+,\ x(t)=z(t)\cdot\big(\sin(\omega_1 t)+\sin(\omega_2 t)\big),$$

where $\omega_1>0,\ \omega_2>0$ are constants, and $z=\{z(t),\ t\in\mathbb{R}_+\}$ is a zero mean weakly stationary process with the following covariance function:

$$B_z(t,s)=E(z(t)\cdot z(s))\triangleq B_z(\tau)=|\tau|\cdot\exp\big\{-|\tau|\big\},\quad \tau=t-s.$$

In practice $x=\{x(t),\ t\in\mathbb{R}_+\}$ may represent a stochastic telegraph signal. It is easy to prove $x=\{x(t),\ t\in\mathbb{R}_+\}$ is a non-stationary process. By the spectral representation (7.7) we get

$$B_z(\tau)=\frac{1}{2}\int_{-\infty}^{+\infty}\exp\{I\theta\tau\}f(\theta)\mathrm{d}\theta,\ \ f(\theta)=\frac{1}{\pi}\int_{-\infty}^{+\infty}\exp\{-I\theta\tau\}B_z(\tau)\mathrm{d}\tau, \tag{7.42}$$

where $I^2=-1$. By (7.42) it follows that

$$\begin{aligned} f(\theta) &= \frac{1}{\pi}\int_{-\infty}^{+\infty}\exp\{-I\theta\tau\}\cdot|\tau|\exp\{-|\tau|\}\mathrm{d}\tau\\ &= \frac{2}{\pi}\int_{0}^{+\infty}\tau\cdot\exp\{-\tau\}\cos(\theta\tau)\mathrm{d}\tau\\ &= \frac{2}{\pi}\int_{0}^{+\infty}\Big\{\exp\{-\tau\}\cos(\tau\theta)-\theta\tau\cdot\exp\{-\tau\}\sin(\tau\theta)\Big\}\mathrm{d}\tau\\ &= \frac{2}{\pi}\int_{0}^{+\infty}\exp\{-\tau\}\big(\cos(\tau\theta)-\theta\sin(\tau\theta)-\theta^2\tau\cdot\cos(\theta\tau)\big)\mathrm{d}\tau \end{aligned}$$

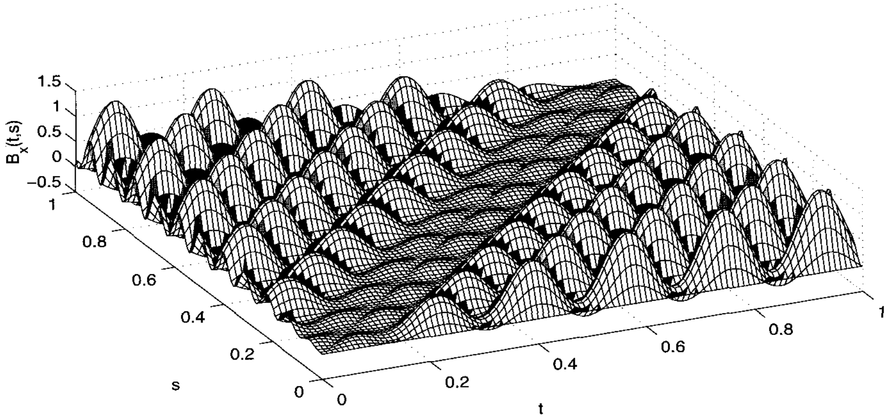


Figure 7.3 The surface of B_x

Therefore we can conclude that the following fact holds:

$$\int_0^{+\infty} \exp\{-\tau\}\cos(\tau\theta)\mathrm{d}\tau = \frac{1}{1+\theta^2}, \quad \int_0^{+\infty} \exp\{-\tau\}\sin(\tau\theta)\mathrm{d}\tau = \frac{\theta}{1+\theta^2}.$$

Thus, $f(\theta) = 2/\big(\pi(\theta^2+1)^2\big)$. Since $B_x(t, s) = E(x(t)\cdot x(s))$, by (7.42) it follows that

$$\begin{aligned} B_x(t,s) &= E(z(t)\cdot z(s))\cdot \big(\sin(\omega_1 t)+\sin(\omega_2 t)\big)\big(\sin(\omega_1 s)+\sin(\omega_2 s)\big) \\ &= \frac{2}{\pi}\cdot \big(\sin(\omega_1 t)+\sin(\omega_2 t)\big)\big(\sin(\omega_1 s)+\sin(\omega_2 s)\big)\cdot \int_{-\infty}^{+\infty} \exp\{I\theta\tau\}\cdot \frac{\mathrm{d}\theta}{(\theta^2+1)^2} \\ &= \frac{4}{\pi}\cdot \big(\sin(\omega_1 t)+\sin(\omega_2 t)\big)\big(\sin(\omega_1 s)+\sin(\omega_2 s)\big)\cdot \int_0^{+\infty} \frac{\cos\big(\theta(t-s)\big)}{(\theta^2+1)^2}\mathrm{d}\theta. \end{aligned}$$

Put $\Phi^{\mathrm{T}}(t, \theta) = \big(\varphi_1(t, \theta),\ \varphi_2(t, \theta)\big)$, and denote

$$\varphi_1(t, \theta) = \sqrt{2}\cdot \big(\sin(\omega_1 t)+\sin(\omega_2 t)\big)\cos(\theta t),$$

$$\varphi_2(t, \theta) = \sqrt{2}\cdot \big(\sin(\omega_1 t)+\sin(\omega_2 t)\big)\sin(\theta t).$$

Considering the fact: $F(\mathrm{d}\theta) = f(\theta)\mathrm{d}\theta$, we can imply

$$B_x(t, s) = \int_0^{+\infty} \big\langle \Phi^{\mathrm{T}}(t, \theta),\ \Phi(s, \theta)\big\rangle F(\mathrm{d}\theta).$$

Let $\Psi^{\mathrm{T}}(t, \theta) = \big(\psi_1(t, \theta),\ \psi_2(t, \theta)\big)$, where $\psi_1(\cdot, \cdot)$, $\psi_2(\cdot, \cdot)$ can be expressed respectively as follows:

$$\psi_1(t, \theta) = \frac{2}{\sqrt{\pi}} \frac{\big(\sin(\omega_1 t)+\sin(\omega_2 t)\big)\cos(\theta t)}{\theta^2+1},$$

$$\psi_2(t, \theta) = \frac{2}{\sqrt{\pi}} \frac{\big(\sin(\omega_1 t)+\sin(\omega_2 t)\big)\sin(\theta t)}{\theta^2+1}.$$

Since $\|\Psi(\cdot,\,\cdot)\|^2 = \psi_1^2(\cdot,\,\cdot) + \psi_2^2(\cdot,\,\cdot)$, easily we have, $\|\Psi(\cdot,\cdot)\| \in L^2(\mathbb{R}_+^2,\ \mathcal{B}\times\mathcal{B},\ G\times M)$. Choosing $\omega_1 = \omega_2 = 35$, we can show

$$\int_{t>1,\theta>1} \big(\psi_1^2(t,\,\theta) + \psi_2^2(t,\,\theta)\big)M(\mathrm{d}\theta)G(\mathrm{d}t) < 0.005 = \frac{\varepsilon^2}{8}.$$

So choose $a = 1$. If partition $[0,\,1]$ identically into $m_0(m_0 \geq 20)$ parts, and consequently $[0,\,1]^2$ is divided into m_0^2 sufficiently small squares. Thus two piecewise linear function $S_1(\cdot,\,\cdot)$, $S_2(\cdot,\,\cdot)$ can be constructed, satisfying

$$\int_{[0,1]^2} \big|\psi_1^2(t,\,\theta) + \psi_2^2(t,\,\theta) - S_1^2(t,\,\theta) - S_2^2(t,\,\theta)\big|G(\mathrm{d}t)M(\mathrm{d}\theta) < \frac{\varepsilon^2}{8}.$$

In order to estimate m, we use Corollary 6.2 to estimate $D_H(\psi_i)$: $D_H(\Psi) \triangleq D_H(\psi_1) \vee D_H(\psi_2) < 40$. Since $\mu\big([0,\,a]^2\big) = (G\times M)\big([0,\,1]^2\big) < 1/10$, it follows by Theorem 7.1 that

$$m > \frac{4D_H(\Psi)c_0\cdot\sqrt{(G\times M)\big([0,1]^2\big)}}{\varepsilon}, \Longrightarrow m \geq \frac{4\times 40\times 2}{0.2\times\sqrt{10}}, \Longrightarrow m = 506.$$

In Theorem 7.4 we may let $q = 10$. Then by (7.40) and Theorem 7.2 we can get an analytic learning algorithm for the stochastic Mamdani fuzzy system $M_{a,m}(t) = \sum\limits_{p=0}^{506} \widetilde{A}_{1p}(t)\cdot O(p)$:

$$\begin{cases} O(p) = \sum\limits_{j=1}^{10}\sum\limits_{p_2=0}^{506} \widetilde{A}_{2p_2}(\theta_j)\Big(r_1(p,\,p_2)\big(b_1(\theta_j) - b_1(\theta_{j-1})\big) \\ \qquad\qquad\qquad + r_2(p,\,p_2)\big(b_2(\theta_j) - b_2(\theta_{j-1})\big)\Big) \\ r_1(p,p_2) = \psi_1\Big(\dfrac{p}{506},\,\dfrac{p_2}{506}\Big),\ \ r_2(p,p_2) = \psi_2\Big(\dfrac{p}{506},\,\dfrac{p_2}{506}\Big). \end{cases} \tag{7.43}$$

For $i = 1,\,2$ using (7.39) and considering $\theta_j = j/10$, we have

$$E(|b_i(\theta_j) - b_i(\theta_{j-1})|^2) = |\theta_j - \theta_{j-1}| = \frac{1}{10}. \tag{7.44}$$

So by learning algorithm (7.40) and (7.34) (7.44) it follows that

$$B_{M_{a,m}}(t,s) = \sum_{p_1,p_2=0}^{506} \widetilde{A}_{1p_1}(t)\cdot \widetilde{A}_{1p_2}(s)E\big(O(p_1)\cdot O(p_2)\big)$$

$$= \sum_{p_1,..,p_4=0}^{506}\sum_{j=1}^{10} \widetilde{A}_{1p_1}(t)\cdot \widetilde{A}_{1p_2}(s)\cdot \widetilde{A}_{2p_3}(\theta_j)\cdot \widetilde{A}_{2p_4}(\theta_j)\cdot$$

$$\cdot\Big\{\psi_1\Big(\frac{p_1}{506},\,\frac{p_3}{506}\Big)\cdot\psi_1\Big(\frac{p_2}{506},\,\frac{p_4}{506}\Big)\cdot E\big(|b_1(\theta_j) - b_1(\theta_{j-1})|^2\big)$$

$$+\psi_2\Big(\frac{p_1}{506},\,\frac{p_3}{506}\Big)\cdot\psi_2\Big(\frac{p_2}{506},\,\frac{p_4}{506}\Big)E(|b_2(\theta_j) - b_2(\theta_{j-1})|^2)\Big\}$$

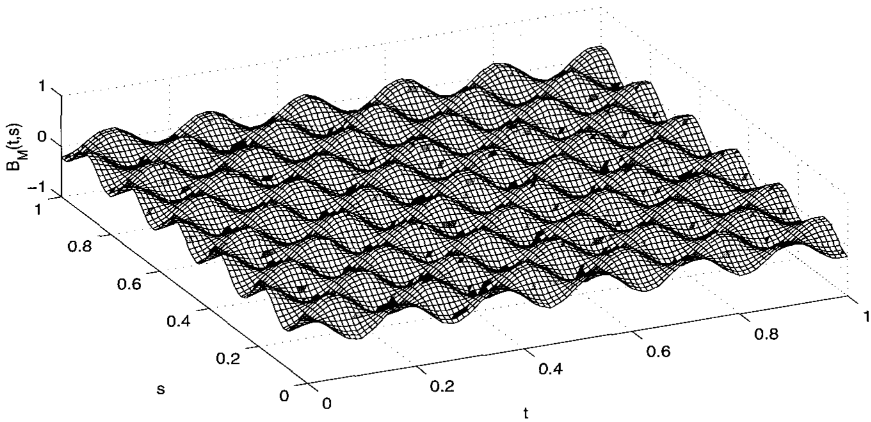


Figure 7.4 The surface of $B_{M_{a,m}}$ when $m = 40$

Therefore we can obtain

$$B_{M_{a,m}}(t,s) = \frac{1}{10}\sum_{p_1,\cdots,p_4=0}^{506}\sum_{j=1}^{10}\tilde{A}_{1p_1}(t)\cdot\tilde{A}_{1p_2}(s)\cdot\tilde{A}_{2p_3}\Big(\frac{j}{10}\Big)\cdot\tilde{A}_{2p_4}\Big(\frac{j}{10}\Big)\cdot$$
$$\cdot\Big\{\psi_1\Big(\frac{p_1}{506},\frac{p_3}{506}\Big)\cdot\psi_1\Big(\frac{p_2}{506},\frac{p_4}{506}\Big)+\psi_2\Big(\frac{p_1}{506},\frac{p_3}{506}\Big)\cdot\psi_2\Big(\frac{p_2}{506},\frac{p_4}{506}\Big)\Big\}.$$

For simplicity of computation, we choose $m = 40$ to derive the surface of $B_{M_{a,m}}(\cdot,\cdot)$ as shown in Figure 7.4. And Figure 7.3 is the surface of $B_x(\cdot,\cdot)$. From the comparison between Figure 7.3 and Figure 7.4, we may easily see the accuracy ε in mean square sense is guaranteed.

References

[1] Abe S. & Lan M. S., Fuzzy rules extraction directly from numerical data for function approximation, *IEEE Trans. on systems, Man and Cybernet.*, 1995, **25**(1): 119–129.

[2] Ashrafzadeh F., Nowicki E. P., Mohamadian M. & Salmon J. C., Optimal approximation of nonlinear functions by fuzzy systems, *Proc. of IEEE Canadian Conf. on Electrical and Computer Engineering, Engineering Innovation: Voyage of Discovery,* 1997, **2**: 781–784.

[3] Ahmet A. & Mehmet K., Determination of fuzzy logic membership functions using genetic algorithms, *Fuzzy Sets and Systems,* 2001, **118**(2): 297–306.

[4] Belli M. R.,Conti M. Crippa P. & Turchetti C., Artificial neural network as approximators of stochastic processes, *Neural Networks*, 1999, **12**: 647–658.

[5] Buckley J. J. & Hayashi Y., Fuzzy input-output controllers are universal approximators, *Fuzzy Sets and Systems,* 1993, **58**:273–278.

[6] Burton R. M. & Dehling H. G., Universal approximation in p-mean by neural networks, *Neural Networks,* 1998, **11**(4): 661–667.

[7] Card H. C., Dynamics of stochastic artificial neurons, *Neurocomputing,* 2001, **41**(1-4): 173–182.

[8] Carozza M. & Rampone S., Function approximation from noise data by an incremental RBF network, *Pattern Recognition,* 1999, **32**: 2081–2083.

[9] Chakraborty S., Pal K. & Nikhil Pal R., A neuro-fuzzy framework for inferencing, *Neural Networks,* 2002, **15**: 247-261.

[10] Conti M. & Turchetti C., Approximation of dynamical systems by continuous-time recurrent approximate identity neural networks, *Neural, Parallel & Sci. Comput.,* 1994, **2**(2): 299–322.

[11] Conti M., Orcioni M. & Turchetti C., A class of neural networks based on approximate identity for analog IC's hardware implementations , *IEICE Trans. Fundamentals,* 1994, **E77-A**(6): 1069–1079.

[12] Ding Y., Ying H. & Shao S., Necessary conditions on minimal system configuration for general MISO Mamdani fuzzy systems as universal approximators, *IEEE Trans. on Systems, Man and Cybernet.– Part B,* 2000, **30**(6): 857–864.

[13] Dickerson J. A. & Kosko B., Fuzzy function approximation with ellipsoidal rules, *IEEE Trans. on Systems, Man and Cybernet. Part B,* 1996, **26**(4): 542–560.

[14] Dirk T. & Andreas S., A modified general regression neural network (MGRNN) with new, efficient training algorithms as a robust 'black box'-tool for data analysis, *Neural Networks,* 2001, **14**: 1023–1034.

[15] Doob J. L., *Stochastic Processes,* New York: Wiley, 1990.

[16] Dunyak J. & Wunsch D., A training technique for fuzzy number neural networks, *Proc. IEEE Internat. Conf. on Neural Network,* 1997: 533–536.

[17] Enrique F. -M., Universal fuzzy system to Takagi-Sugeno fuzzy system compiler, *Proc. of IEEE Internat. Conf. on Fuzzy systems (FUZZ-IEEE'02),* 2002, **1**: 305–309.

[18] Fiordaliso A., Autostructruation of fuzzy systems by rules sensitivity analysis, *Fuzzy Sets and Systems,* 2001, **118**(2): 281–296.

[19] Gardner W. A., *Introduction to random processes—with applications to signals and systems,* New York: Macmillan Publishing Company, 1985.

[20] Gelenbe E., Mao Z. H. & Li Yeda, Function approximation with spiked random networks, *IEEE Trans. on Neural Networks,* 1999, **10**: 3–9.

[21] S. Ghosh and Q. Razouqi, et al, A survey of recent advances in fuzzy logic in telecommunication networks and new challenges, *IEEE Trans. on Fuzzy Systems,* 1998, **6**(4): 443–447.

[22] Gihman I. I. & Skorohod A. V., *The theory of stochastic process,* Berlin, Germany: Springer–Verlag, 1974.

[23] Hunt K. J. & Haas R., Extending the functional equivalence of radial basis function networks and fuzzy inference systems, *IEEE Trans. on Neural Networks,* 1996, **7**(3): 776–781.

[24] Ishibuchi H., Nii M. & Oh C. -H., Approximate realization of fuzzy mapping by regression models, neural networks and rule-based systems, *Proc. IEEE Internat. Conf. on Fuzzy Systems,* 1999: 939–944.

[25] Izquierdo J. M. C., Dimitriadis Y. A. & Sánchez E. G., et al, Learning from noisy information in FasArt and FasBack neuro-fuzzy systems, *Neural Networks,* 2001, **14**: 407–425.

[26] Jang J. S. R., Sun C. T. & Mizutani E., *Neuro-fuzzy and soft computing,* Prentice Hall, 1997.

[27] Kaufmann A. & Gupta M. M., *Introduction to Fuzzy Arithmetic: Theory and Application,* New York: Van Nostrand Reinhold, 1991.

[28] Kohonen T., *Self-organization and associative memory,* Germany: Springer, 1988.

[29] Kurano M., Yasuda M. & Nakagami J. I., An approach to stopping problems of a dynamic fuzzy system, *Fuzzy Sets and Systems,* 2002, **131**(2): 225–233.

[30] Kushner H. J., *Approximation and weak convergence methods for random processes,* Cambridge MA: MIT Press, 1984.

[31] Lee M. R., Rhee H., Cho K. & Cho B. J., Self-tuning of the fuzzy inference rule by integrated method, *Journal of Intelligent & robotic systems,* 2002, **33**(3): 313–327.

[32] Leski J. & Czogala E., Fuzzy systems, *Fuzzy and Neuro-fuzzy Intelligent Systems,* 2000: 129–139.

[33] Lin W. S., Tsai C. H. & Liu J. S., Robust neuro-fuzzy control of multivariable systems by tuning consequent membership functions, *Fuzzy Sets and Systems,* 2001, **124**(2): 181–195.

[34] Lisin D. & Gennert M. A., Optimal function approximation using fuzzy rules, *Proc. Of IEEE Internat. Conf. of North American Fuzzy Information Processing Society,* 1999, **1**: 184–188.

[35] Liu Puyin, Mamdani fuzzy system: universal approximator to a class of random processes, *IEEE Trans. on Fuzzy Systems,* 2002, **10**(6): 756–766.

[36] Liu Puyin, Fuzzy valued Markov processes and their properties, *Fuzzy Sets and Systems,* 1997, **91**: 87–96.

[37] Liu Puyin & Li Hongxing, Analyses for $L_p(\mu)$-norm approximation capability of generalized Mamdani fuzzy systems, *Information Science,* 2001, **138**: 195–210.

[38] Liu Puyin & Li Hongxing, Hierarchical TS fuzzy systems are universal approximators, *Information Science,* 2004 (accepted for publication).

[39] Liu Puyin & Li Hongxing, T–S fuzzy systems can approximate a class of random processes, *Knowledge-based Systems,* 2004 (to appear).

[40] Liu Puyin & Li Hongxing, Approximation of generalized fuzzy system to integrable function, *Science in China, Series E,* 2000, **43**: 618–628.

[41] Liu Puyin & Li Hongxing, Fuzzy Techniques in image restoration— a survey, *Internat. Journal of Comp. Cognition,* 2004, **2**(2): 131–149.

[42] Mitaim S. & Kosko B., Adaptive stochastic resonance, *Proc. IEEE,* 1998, **86**(11): 2152–2183.

[43] Mitaim S. & Kosko B., What is the best shape for a fuzzy set in function approximation, *Proc. of the Fifth IEEE Internat. Conf. on Fuzzy systems,* 1996, **2**: 1237–143.

[44] Moon B. S., A practical algorithm for representing polynomials of two variables by fuzzy systems with accuracy O(h^4), *Fuzzy Sets and Systems,* 2001, **119**(2): 321– 327.

[45] Nakagami Y. Y., Kurano J. M. & Yasuda M., Optimal stopping problems in a stochastic and fuzzy system, *J. Math. Anal. Appl.,* 2000, **246**(1): 135–149.

[46] Nejim S., Optimal approximation of functions using piecewise polynomial fuzzy systems, *Proc. of IEEE Internat. Conf. of North American Fuzzy Information processing Society,* 1999, **1**: 193–197.

[47] Park J. -H., Seo S. -J. & Park G. -T., Robust adaptive fuzzy controller for nonlinear system using estimation of bounds for approximation errors, *Fuzzy Sets and Systems,* 2003, **133**(1): 19–36.

[48] Park J. & Sandberg I. W., Universal approximation using radial–basis–function networks, *neural Comput.,* 1991, **8**(3): 246–257.

[49] Poggio T. & Girosi F., Networks for approximation and learning, *Proc. IEEE,* 1990, **78**: 1481–1497.

[50] Pomares H., Rojas I. & Ortega J. et al, A systematic Approach to a self-generating fuzzy rule-table for function approximation, *IEEE Trans. on Systems, Man and Cybernet., Part B,* 2000, **30**(3): 431–447.

[51] Pushkov S. G., On the general theory of fuzzy systems: global sates and the fuzzy global reaction of a fuzzy system, Internat. J. Computer and Systems Sciences. 2001, 40(5): 778–781.

[52] Rojas I., Gonzalez J. & Pomares H., Multidimensional and multideme genetic algorithms for the construction of fuzzy systems, *Internat. J. Approx. Reasoning,* 2001, **26**(3): 179–201.

[53] Sala A. & Albertos P., Inference error minimisation: fuzzy modelling of ambiguous functions, *Fuzzy Sets and Systems,* 2001, **121**: 95–111.

[54] Salmeri M., Mencattini A. & Rovatti R., Function approximation using non-normalized SISO fuzzy systems, *Internat. Journal of Approx. Reasoning,* 2001, **26**(3): 211–231.

[55] Turchetti C. & Conti M., Approximate identity neural networks for analog synthesis of nonlinear dynamical systems, *IEEE Trans. on Circuits Systems,* 1994, **41**: 841–858.

[56] Turchetti C., Conti M., Crippa P. & Orcioni S., On the approximation of stochastic processes by approximate identity neural networks, *IEEE Trans. on Neural Networks,* 1998, **9**: 1069–1084.

[57] Wong C. -C. & Chen C. -C., A GA-based method for constructing fuzzy systems directly from numerical data, *IEEE Trans. on Systems Man and Cybernet., Part B,* 2000, **30**(6): 904–911.

[58] Wan Feng, Wang L. X., Zhu H. Y. & Sun Y. X., How to determine the minimum number of rules to achieve given accuracy, *Proc. of 10th IEEE Internat. Conf. on Fuzzy Systems (FUZZ-IEEE 2001),* 2001, **2**: 566–569.

[59] Wang L. X., Designing fuzzy models for nonlinear discrete-time systems with guaranteed performance, *Proc. of American Control Conference,* 1998, **3**: 1864–1865.

[60] Wang L. X., Chen Wei, Approximation accuracy of some neuro-fuzzy approaches, *IEEE Trans. on Fuzzy System,* 2000, **8**(4): 470–478.

[61] Ying H., General Takagi–Sugeno fuzzy systems are universal approximators, *Proc. of the third IEEE Conf. on Fuzzy Systems, IEEE World Congress on Computational Intelligence,* 1998, **1**: 819–823.

[62] Ying H., Ding Y. Li S. & Shao S., Typical Takagi–Sugeno and Mamdani fuzzy systems as universal approximators: necessary conditions and comparison, *Proc. of the third IEEE Conf. on Fuzzy Systems, IEEE World Congress on Computational Intelligence,* 1998, **1**: 824–828.

[63] Ying H., Ding Y. Li S. & Shao S., Comparison of necessary conditions for typical takagi–Sugeno and Mamdani fuzzy systems as universal approximators, *IEEE Trans. on Systems, Man and Cybernet.– Part A,* 1999, **29**(5): 508–514.

[64] Zeng X. J. & Singh M. G., Approximation theory of fuzzy systems–SISO case, *IEEE Trans. on Fuzzy Systems,* 1994, **2**: 162–176.

[65] Zeng X. J. & Singh M. G., Approximation properties of fuzzy systems generated by the min inference, *IEEE Trans. on Systems, Man and Cybernet.– Part B,* 1996, **26**: 187–193.

CHAPTER VIII
Application of FNN to Image Restoration

The ambiguity and uncertainty always accompany the acquisition and transmission of a real digital image. So in practice only a few of classes of degraded images can be treated successfully by mathematical model methods [4, 5], most degraded image cases can not be modelled by the conventional approaches. As a soft technique for dealing with the imprecision inhering in human brain, fuzzy sets can be an efficient tool to treat graded images, especially noise images [3, 10, 52–54]. Through fuzzy sets, we may use human knowledge expressed heuristically in natural language to describe digital images. Such a fact may result in the well-known rule-based approach for noise image processing. We can employ heuristic knowledge on noise images to build some suitable inference rules for removing noise [25–29, 54, 62]. To improve the adaptivity and filtering capability, an efficient approach is to utilize neural network filters [55, 69] and fuzzy filters synthetically. As an organic fusion of fuzzy theory and neural networks, the FNN approach for signal and image processing has been of growing interest [2, 32, 66, 68]. The success of FNN's in image processing is that the filters based on FNN's can efficiently suppress noise without destroying important image details such as edges [14, 25, 37–39]. Using adaptive adjust of the FNN we can get an optimal filtering result, and some restored images with good performances can be achieved.

In the chapter, fuzzy inference networks are studied systematically with a general approach, which is the basis to develop image restoration methods. Then we express a two dimensional digital image as a I/O relationship of a fuzzy inference network. And consequently we can develop a corresponding optimal filter, by which a restoration image with good performance can be resulted in from a noise one. To this end, we begin our research to introduce a defuzzifier with general sense, and then define generalized fuzzy inference neural networks (FINN's) and prove that the generalized FINN's can be universal approximators. We establish the equivalence between a FINN and a generalized fuzzy system. We utilize fuzzy sets to describe the gray levels of digital images. Then two dimensional digital images can be dealt with by using fuzzy inference networks, and an efficient FNN filter can be built. With the mean absolute error (MAE), some learning algorithms for the fuzzy inference networks can be developed to design optimal FNN filters. They can lead to good anti-disturbance in image processing, that is, if images are corrupted by impulse noise with high noise occurrence probability ($p > 0.5$), we can employ the FNN filter to give restoring image with good performance. Many simulation examples are presented to show the methods in the chapter are advantageous and efficient in processing noise images.

§8.1 Generalized fuzzy inference network

Fuzzy inference system can simulate and realize natural language and logic inference mechanic. The subjects related, such as how fuzzy rule base can be constructed by given linguistic and data information, whether the systems related can adaptively match fuzzy rules, and so on attract many scholar's attention [45, 46, 50, 57]. As a organic fusion of inference system and neural network, fuzzy inference network can realize automobile generation and automobile matching of fuzzy rules. Further, such a system can adaptively adjust to adapt the changes of conditions and self-improvement. Since 1990, many achievements have been achieved and they have found useful in many applied areas, for example, process control [33, 47], system modelling and system identification [20, 22, 27, 35, 42], expert system [23], forecasting [49] and so on. In the following, we shall study a class of generalized fuzzy inference network within a general framework, and discuss its all kinds of properties, including universal approximation.

8.1.1 Generalized defuzzification operator

As one of main components of fuzzy inference system, defuzzification constitutes one important study object in fuzzy system and fuzzy control [12, 41, 63, 64]. Defuzzification is a procedure by which a fuzzy set is transformed into one crisp value of being able to describe the fuzzy set. In fuzzy control, it turns a fuzzy decision into a concrete control value and system control may be realized. In defuzzification methods, there are two main classes most used. One is maximum of mean (MOM) method [23, 50, 67], and another is center of gravity method (COG) [23, 67]. In addition, many novel defuzzifications to the special subjects are put forward in recent years. These defuzzification approaches have their own advantages and disadvantages [63]. It is impossible to develop a general framework for defuzzification, including all cases. This is because in practice all models possess their own characteristics.

In this subsection, we shall build some specific principles to define defuzzification operators [50, 58]. So it is possible to develop a more general definition for defuzzification. If $a \in [0,1]$, $\tilde{A} \in \mathcal{F}(\mathbb{R})$, then define fuzzy set $a\,T\,\tilde{A} \in \mathcal{F}(\mathbb{R})$, such that $\forall x \in \mathbb{R}$, $(a\,T\,\tilde{A})(x) = a\,T\,\tilde{A}(x)$. In the following we assume that the fuzzy operator T satisfies the condition: $\forall a,\ b \in [0,\ 1]$, $a,\ b > 0, \Longrightarrow a\,T\,b > 0$.

Definition 8.1 Suppose $D_e : \mathcal{F}(\mathbb{R}) \longrightarrow \mathbb{R}$ is a mapping. And let $T : [0,1] \times [0,1] \longrightarrow [0,1]$ be a t$-$norm. $D_e(\cdot)$ is called generalized defuzzification operator, if the following conditions hold:

(i) $\forall \widetilde{A}\in \mathcal{F}(\mathbb{R}),\ D_e(\widetilde{A}) \in \text{Supp}(\widetilde{A})$, thus, $\forall x \in \mathbb{R}, D_e\big(\chi_{\{x\}}\big)= x$;

(ii) $D_e(\cdot)$ is a continuous mapping, that is, if $\widetilde{A}\in \mathcal{F}(\mathbb{R}),\ \{\widetilde{A}_n\ |n \in \mathbb{N}\} \subset \mathcal{F}(\mathbb{R})$, then $D(\widetilde{A}_n, \widetilde{A}) \to 0 (n \to +\infty), \Longrightarrow D_e(\widetilde{A}_n) \to D_e(\widetilde{A})\ (n \to +\infty)$;

(iii) Arbitrarily given $\widetilde{A}_1, ..., \widetilde{A}_n\in \mathcal{F}(\mathbb{R})$, then

$$\bigwedge_{i=1}^{n}\{D_e(\widetilde{A}_i)\} \le D_e\Big(\bigcup_{i=1}^{n} \widetilde{A}_i\Big)\le \bigvee_{i=1}^{n}\{D_e(\widetilde{A}_i)\}.$$

Lemma 8.1 *Let $a \in [0,1]$, and $\widetilde{A}\in \mathcal{F}(\mathbb{R}),\ \{a_n|n \in \mathbb{N}\} \subset [0,1]$. Then the following conclusions hold:*

(i) $D_e : \mathcal{F}(\mathbb{R}) \longrightarrow \mathbb{R}$ is a surjection, that is, $\forall x \in \mathbb{R}$, there is $\widetilde{B}\in \mathcal{F}(\mathbb{R})$, such that $D_e(\widetilde{B}) = x$, consequently, for each $a \in (0,1]$, there is $\widetilde{A}\in \mathcal{F}(\mathbb{R})$ independently of a, satisfying $D_e(a\,T\widetilde{A}) = x$;

(ii) If T is a continuous t−norm, and $\widetilde{A}$ satisfying condition: $\widetilde{A}(\cdot)$ is continuous and Supp$(\widetilde{A})$ is bounded. Further for $\alpha \in [0,1]$, $\widetilde{A}_\alpha$ is a bounded convex set. Then $\lim\limits_{n\to+\infty} a_n = a, \Longrightarrow \lim\limits_{n\to+\infty} D_e(a_n\,T\widetilde{A}) = D_e(a\,T\widetilde{A})$, and $D_e(a\,T\widetilde{A})$ is continuous with respect to a.

Proof. (i) Choose $\widetilde{B}= \chi_{\{x\}}$, i.e. $\widetilde{B}$ is the characteristic function of the point set $\{x\}$. Then $\widetilde{B}\in \mathcal{F}(\mathbb{R})$. By Definition 8.1, $D_e(\widetilde{B}) = x$. And if $a \in (0,\ 1]$, we choose $\widetilde{A}=\widetilde{B}= \chi_{\{x\}}$. Then $\text{Supp}(a\,T\,\widetilde{A}) = \text{Supp}(a\,T\,\chi_{\{x\}}) = \{x\}$. Hence $D_e(a\,T\widetilde{A}) = x$. (i) holds.

(ii) Given arbitrarily $\beta \in [0,\ 1]$, then $\widetilde{A}_\beta$ is a bounded interval, and let the left, right endpoints related be $a^L(\beta),\ a^U(\beta)$, respectively. For any $b \in [0,\ 1]$, $\alpha \in [0,\ 1]$, if $\alpha > b$, obviously we have, $\big(b\,T\widetilde{A}\,\big)_\alpha= \emptyset$; and if $\alpha \in [0,\ b]$, in the following we show

$$bT\beta = \alpha, \Longrightarrow \big(a^L(\beta),\ a^U(\beta)\big)\subset \big(b\,T\widetilde{A}\,\big)_\alpha\subset \big[a^L(\beta),\ a^U(\beta)\big]. \tag{8.1}$$

In fact, for any $x \in \big(a^L(\beta),\ a^U(\beta)\big)\subset\widetilde{A}_\beta$, it follows that $\widetilde{A}(x) \ge \beta$. Then we have, $b\,T\widetilde{A}(x) = \big(b\,T\widetilde{A}\,\big)(x) \ge b\,T\,\beta = \alpha$, that is

$$x \in (b\,T\widetilde{A})_\alpha, \Longrightarrow \big(a^L(\beta),\ a^U(\beta)\big)\subset (b\,T\widetilde{A})_\beta;$$

Conversely, if $x \in (b\,T\widetilde{A})_{\underset{\bullet}{\alpha}}$, then $b\,T\widetilde{A}(x) > \alpha$. Since $\alpha \in [0,\ b] = \big[b\,T\,0,\ b\,T\,1\big]$, by the continuity of T there exists $\beta \in [0,1]$, so that $b\,T\,\beta = \alpha$. Then $\widetilde{A}(x) \ge \beta$, for otherwise we have, $\widetilde{A}\,(x) < \beta, \Longrightarrow b\,T\,\widetilde{A}\,(x) \le b\,T\,\beta = \alpha$, which is a contradiction. So $(b\,T\widetilde{A})_{\underset{\bullet}{\alpha}} \subset\widetilde{A}_\beta$. And (8.1) holds.

$\forall \alpha \in [0, 1]$, $\forall n \in \mathbb{N}$, it follows that $\{a^L(\beta_n)|n \in \mathbb{N}\}$, $\{a^U(\beta_n)|n \in \mathbb{N}\}$ are the left, right endpoints of $(a_n T \widetilde{A})_\alpha$ respectively. Considering $\lim_{n\to+\infty} a_n = a$, we choose β_n satisfying the condition: $\beta_n T a_n = \alpha$, moreover $\lim_{n\to+\infty} \beta_n = \beta$. Also by $\widetilde{A}(\cdot)$ being continuous, $a^L(\cdot)$, $a^U(\cdot)$ are continuous functions. Thus, $\lim_{n\to+\infty} a^L(\beta_n) = a^L(\beta)$, $\lim_{n\to+\infty} a^U(\beta_n) = a^U(\beta)$. Therefore, $(a_n T \widetilde{A})_\alpha \xrightarrow{d_H} (a T \widetilde{A})_\alpha$ $(n \to +\infty)$. Hence $\lim_{n\to+\infty} D(a_n T \widetilde{A}, a T \widetilde{A}) = 0$. (ii) is proved. □

In order to account for the fact that most of defuzzification methods in application are special cases of the generalized defuzzifier in Definition 8.1, we at first restrict the fuzzy sets related into $\widetilde{\mathcal{O}}(a, m_1+m_2)$, for this defining form for fuzzy sets is widely applied in application. We obtain the fuzzy set family $\{\widetilde{A}_j \; |j = -m_1, -m_1+1, ..., 0, 1, ..., m_2\} \subset \widetilde{\mathcal{O}}\,(a, m_1+m_2)$, that is, we partition $[-a, a]$ identically into $m_1 + m_2$ sub-intervals: $-a = a_{-m_1} < a_{-m_1+1} < \cdots < a_0 = 0 < a_1 < \cdots < a_{m_2}$. Let $\widetilde{A}_j$ be a triangular fuzzy number with $\{a_j\}$ being kernel, and a_{j-1}, a_{j+1} being left, right endpoints of the support, respectively, ss Figure 8.1 shown, we can obtain $c_0 = 2$.

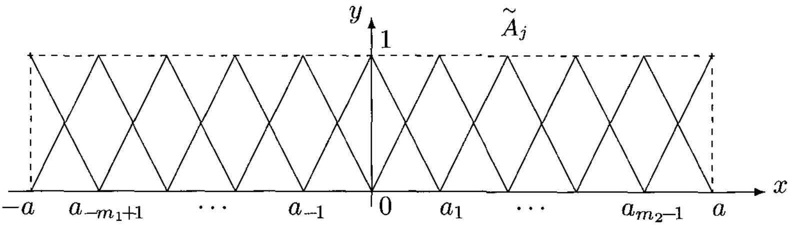


Figure 8.1 The membership curves of fuzzy set family

Example 8.1 For given $\widetilde{A} \in \{\widetilde{A}_j \; |j = -m_1, -m_1+1, ..., 0, 1, ..., m_2\}$, we define the defuzzification methods in the following different cases:

$$\mathrm{COA}(\widetilde{A}) = \frac{\int_{\mathbb{R}} x \cdot \widetilde{A}(x)\mathrm{d}x}{\int_{\mathbb{R}} \widetilde{A}(x)\mathrm{d}x}; \quad \mathrm{MOM}(\widetilde{A}) = \frac{\int_{X^*} x \cdot \widetilde{A}(x)\mathrm{d}x}{\int_{X^*} \widetilde{A}(x)\mathrm{d}x};$$

$$\mathrm{SA}(\widetilde{A}) = \int_0^1 \big(\delta \cdot a_1(\alpha) + (1-\delta) \cdot a_2(\alpha)\big)\mathrm{d}\alpha \big(\widetilde{A}_\alpha = [a_1(\alpha), a_2(\alpha)]\big),$$

where $X^* = \{x^* \in \mathbb{R} | \widetilde{A}(x^*) \text{ is a maximum value of } \widetilde{A}(\cdot)\}$. Then it follows that $\mathrm{COA}(\cdot)$, $\mathrm{MOM}(\cdot)$ and $\mathrm{SA}(\cdot)$ can ensure the conditions (i)–(iii) of Definition 8.1 hold.

Since $\forall \widetilde{A} \in \{\widetilde{A}_j \; |j = -m_1, -m_1+1, ..., 0, 1, ..., m_2\}$, $\widetilde{A}$ are triangular fuzzy number, easily we can show, $\mathrm{MOM}(\widetilde{A})(\cdot)$ [23, 67] and $\mathrm{SA}(\widetilde{A})$ [63] can guarantee the conditions (i)–(iii) of Definition 8.1 hold, for MOM($\tilde{A}$) is a discrete form MOM($\tilde{A}$) $= x^*$ $\big(\tilde{A}(x^*) = 1\big)$; And $\tilde{A} \cup \tilde{B}$ is a convex fuzzy set if and only if Ker($\tilde{A}$) = Ker($\tilde{B}$). So it suffices to show COA($\cdot$) can guarantee the conditions.

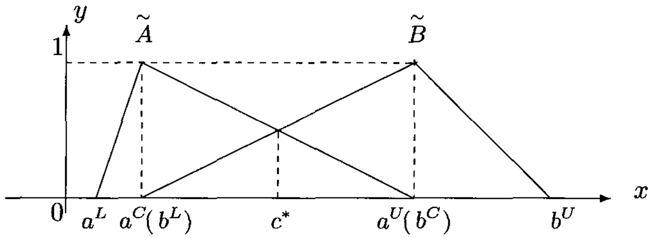


Figure 8.2 The membership curves of two fuzzy intersecting sets

In fact, COA($\cdot$) is a continuous case of centroid defuzzification [23]. Let Supp($\tilde{A}$) $= \big[a^L,\ a^U\big]$. Then $\int_{\mathbb{R}} x\ \tilde{A}(x)\mathrm{d}x = \int_{a^L}^{a^U} x\ \tilde{A}(x)\mathrm{d}x$, so

$$a^L\int_{a^L}^{a^U}\tilde{A}(x)\mathrm{d}x \le \int_{a^L}^{a^U} x\ \tilde{A}(x)\mathrm{d}x \le a^U\int_{a^L}^{a^U}\tilde{A}(x)\mathrm{d}x, \Longrightarrow \mathrm{COA}(\tilde{A}) \in \mathrm{Supp}(\tilde{A}).$$

Also denote $J(\tilde{A}) = \int_{\mathbb{R}} \tilde{A}(x)\mathrm{d}x$, easily we can show, $J(\tilde{A})$ is continuous with respect to $\tilde{A}$. Similarly the integral $\int_{\mathbb{R}} x\ \tilde{A}(x)\mathrm{d}x$ is also continuous with respect to $\tilde{A}$. Therefore, when $J(\tilde{A}) \ne 0$, we have, $\int_{\mathbb{R}} x\ \tilde{A}(x)\mathrm{d}x / \int_{\mathbb{R}} \tilde{A}(x)\mathrm{d}x$ is continuous with respect to $\tilde{A}$, that is, when $D(\tilde{A}_n,\ \tilde{A}) \to 0 (n \to +\infty)$, it follows that $\mathrm{COA}(\tilde{A}_n) \to \mathrm{COA}(\tilde{A})\,(n \to +\infty)$. Next let us show, COA($\cdot$) can ensure the condition (iii). By the induction method it suffices to prove the conclusion when $n = 2$. To this end we at first choose arbitrarily $\tilde{A},\ \tilde{B} \in \big\{\tilde{A}_j\ |j = -m_1,\ -m_1 + 1, ..., 0, 1, ..., m_2\big\}$, and suppose Supp($\tilde{A}$) $= \big[a^L,\ a^U\big]$, Ker($\tilde{A}$) $= \{a^C\}$, also Supp($\tilde{B}$) $= \big[b^L,\ b^U\big]$, Ker($\tilde{B}$) $= \{b^C\}$. Denote

$$s(\tilde{A}) = \int_{a^L}^{a^U}\tilde{A}(x)\mathrm{d}x,\ s(\tilde{B}) = \int_{b^L}^{b^U}\tilde{B}(x)\mathrm{d}x.$$

If $\tilde{A} \cap \tilde{B} = \emptyset$, it follows that

$$\begin{aligned}\mathrm{COA}(\tilde{A} \cup \tilde{B}\,) &= \frac{\int_{a^L}^{a^U} x\cdot \tilde{A}(x)\mathrm{d}x + \int_{b^L}^{b^U} x\cdot \tilde{B}(x)\mathrm{d}x}{\int_{a^L}^{a^U} \tilde{A}(x)\mathrm{d}x + \int_{b^L}^{b^U} \tilde{B}(x)\mathrm{d}x}\\ &= \frac{s(\tilde{A})\cdot \mathrm{COA}(\tilde{A}) + s(\tilde{B})\cdot \mathrm{COA}(\tilde{B})}{s(\tilde{A}) + s(\tilde{B})}.\end{aligned}$$

Since $s(\widetilde{A}) > 0,\ s(\widetilde{B}) > 0$, we get

$$\mathrm{COA}(\widetilde{A}) \wedge \mathrm{COA}(\widetilde{B}) \leq \mathrm{COA}(\widetilde{A} \cup \widetilde{B}) \leq \mathrm{COA}(\widetilde{A}) \vee \mathrm{COA}(\widetilde{B}), \tag{8.2}$$

So the condition (iii) in Definition 8.1 holds; If $\widetilde{A} \cap \widetilde{B} \neq \emptyset$, it is no harm to assume $a^C < b^C$, and $\widetilde{A}$, $\widetilde{B}$ can be characterized by membership curves in Figure 8.2, respectively, where $c^* = (a^C + b^C)/2$. Easily it follows that

$$\mathrm{COA}(\widetilde{A}) = \frac{a^L + a^C + a^U}{3},\quad \mathrm{COA}(\widetilde{B}) = \frac{b^L + b^C + b^U}{3}.$$

For $\widetilde{B}_1, \widetilde{B}_2 \in \mathcal{F}_0(\mathbb{R})$, we can prove

$$\begin{cases} \forall x \geq \mathrm{COA}(\widetilde{B}_2),\ \widetilde{B}_1(x) = \widetilde{B}_2(x);\ \forall x < \mathrm{COA}(\widetilde{B}_2),\ \widetilde{B}_1(x) \geq \widetilde{B}_2(x), \\ \qquad \Longrightarrow \mathrm{COA}(\widetilde{B}_1) \leq \mathrm{COA}(\widetilde{B}_2); \\ \forall x \leq \mathrm{COA}(\widetilde{B}_1),\ \widetilde{B}_1(x) = \widetilde{B}_2(x);\ \forall x > \mathrm{COA}(\widetilde{B}_1),\ \widetilde{B}_1(x) \leq \widetilde{B}_2(x), \\ \qquad \Longrightarrow \mathrm{COA}(\widetilde{B}_1) \leq \mathrm{COA}(\widetilde{B}_2). \end{cases} \tag{8.3}$$

Two conclusions in (8.3) can be proved similarly, so it suffices to show the first one. Denote $\mathrm{Supp}(\widetilde{B}_k) = [b_k^L, b_k^U]$ $(k = 1, 2)$, then $b_1^L \leq b_2^L$. We can assume $\int_{b_1^L}^{\mathrm{COA}(\widetilde{B}_2)} (\widetilde{B}_1(x) - \widetilde{B}_2(x)) \mathrm{d}x \triangleq \triangle_1 > 0$, $\int_{b_1^L}^{b_2^U} \widetilde{B}_2(x) \mathrm{d}x \triangleq \triangle_2 > 0$. Considering

$$\int_{b_1^L}^{\mathrm{COA}(\widetilde{B}_2)} x \widetilde{B}_2(x) \mathrm{d}x + \int_{\mathrm{COA}(\widetilde{B}_2)}^{b_2^U} x \widetilde{B}_2(x) \mathrm{d}x = \mathrm{COA}(\widetilde{B}_2) \cdot \int_{b_1^L}^{b_2^U} \widetilde{B}_2(x) \mathrm{d}x,$$

we can conclude the following facts:

$$\begin{aligned} \mathrm{COA}(\widetilde{B}_1) &= \frac{\int_{b_1^L}^{\mathrm{COA}(\widetilde{B}_2)} x \widetilde{B}_1(x) \mathrm{d}x + \int_{\mathrm{COA}(\widetilde{B}_2)}^{b_2^U} x \widetilde{B}_2(x) \mathrm{d}x}{\int_{b_1^L}^{\mathrm{COA}(\widetilde{B}_2)} \widetilde{B}_1(x) \mathrm{d}x + \int_{\mathrm{COA}(\widetilde{B}_2)}^{b_2^U} \widetilde{B}_2(x) \mathrm{d}x} \\ &= \frac{\triangle_1 \cdot \lambda_1 + \triangle_2 \cdot \mathrm{COA}(\widetilde{B}_2)}{\triangle_1 + \triangle_2}, \end{aligned}$$

where $\lambda_1 = \int_{b_1^L}^{\mathrm{COA}(\widetilde{B}_2)} x (\widetilde{B}_1(x) - \widetilde{B}_2(x)) \mathrm{d}x / \triangle_1 \leq \mathrm{COA}(\widetilde{B}_1)$. So $\mathrm{COA}(\widetilde{B}_1) \leq \mathrm{COA}(\widetilde{B}_2)$. That is, the first conclusion of (8.3) holds. Using (8.2) (8.3) we can conclude that

$$\mathrm{COA}(\widetilde{B}_1) \wedge \mathrm{COA}(\widetilde{B}_2) \leq \mathrm{COA}(\widetilde{B}_1 \cup \widetilde{B}_2) \leq \mathrm{COA}(\widetilde{B}_1) \vee \mathrm{COA}(\widetilde{B}_2).$$

In summary (8.2) holds, so for COA(·) the condition (iii) holds. Hence the conditions (i)–(iii) of Definition 8.1 hold for COA(·).

8.1.2 Fuzzy inference network

For given $a > 0$, and $m \in \mathbb{N}$, define adjustable antecedent fuzzy set family $\{\widetilde{A}_{ij} \mid i = 1, ..., d;\ j = 0, \pm1, ..., \pm m\} \subset \widetilde{\mathcal{O}}(a, m)$. Mamdani inference rule $R_{p_1 ... p_d}$ is presented as in the subsection 6.2.1:

$$R_{p_1 ... p_d}\text{: IF } x_1 \text{ is } \widetilde{A}_{1p_1} \text{ and ... and } x_d \text{ is } \widetilde{A}_{dp_d} \text{ THEN } u \text{ is } \widetilde{U}_{p_1 ... p_d}\ .$$

where $p_1, ..., p_d \in \{-m, -m+1, ..., m-1, m\}$, $\widetilde{U}_{p_1 ... p_d}$ is an adjustable output fuzzy set. For given fuzzy input $\widetilde{A} \in \mathcal{F}([-a, a]^d)$, if $\widetilde{A}$ is singleton fuzzification at $(x_1, ..., x_d) \in [-a, a]^d$, it follows by (6.10) that the output fuzzy set $\widetilde{A} \circ \widetilde{R}_{p_1 ... p_d}$ is determined as follows:

$$(\widetilde{A} \circ \widetilde{R}_{p_1 ... p_d})(u) = H_{p_1 ... p_d}(x_1, ..., x_d)\, T\, \widetilde{U}_{p_1 ... p_d}(u).$$

Using '$\vee - T$' composition rule, by $\{R_{p_1 ... p_d} | p_1, ..., p_d = 0, \pm1, ..., \pm m\}$ we can obtain a synthesizing fuzzy set as follow: $\widetilde{U} \triangleq \bigcup_{p_1, ..., p_d = -m}^{m} (\widetilde{A} \circ \widetilde{R}_{p_1 ... p_d})$:

$$\begin{aligned} \widetilde{U}(u) &= \bigcup_{p_1, ..., p_d = -m}^{m} (\widetilde{A} \circ \widetilde{R}_{p_1 ... p_d})(u) \\ &= \bigvee_{p_1, ..., p_d = -m}^{m} \{H_{p_1 ... p_d}(x_1, ..., x_d)\, T\, \widetilde{U}_{p_1 ... p_d}(u)\}. \end{aligned} \tag{8.4}$$

Assume that D_e is a generalized defuzzifier, and let $F_{in}(x_1, ..., x_d) \triangleq D_e(\widetilde{U})$, it follow that

$$F_{in}(x_1, ..., x_d) = D_e\Big(\bigcup_{p_1, ..., p_d = -m}^{m} (H_{p_1 ... p_d}(x_1, ..., x_d)\, T\, \widetilde{U}_{p_1 ... p_d})\Big). \tag{8.5}$$

As shown in Figure 8.3 we call the system related a fuzzy inference neural network (FINN), whose I/O relationship is as $(x_1, ..., x_d) \longrightarrow F_{in}(x_1, ..., x_d)$, where $K = (2m+1)^d$, k corresponds to a multi-fold index $p_1 ... p_d$ $(p_1, ..., p_d = 0, \pm1, ..., \pm m)$. k_0 is the neuron $p_1 ... p_d$ corresponding to the case of $p_1 = \cdots = p_d = -m$. The FINN architecture consists of five neuron layers, of which there are three hidden layers, one input layer and one output layer. The neuron in hidden layers is called fuzzy inference unit which deals with fuzzy inference rule $R_{p_1 ... p_d}$ $(p_1, ..., p_d = 0, \pm1, ..., \pm m)$. There are d neurons in input layer and one neuron in output layer, which is called synthesizing–defuzzification unit. There are two functions related to output neuron, first synthesizing all fuzzy inference rules by the operator '$\vee$' to establish synthetic fuzzy set $\widetilde{U}$; Second, generalized defuzzifier $D_e(\cdot)$ is applied to $\widetilde{U}$ to derive the crisp output $F_{in}(x_1, ..., x_d)$.

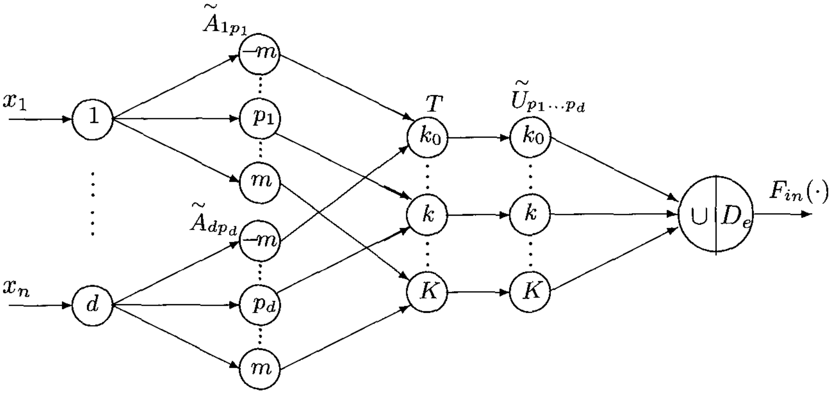


Figure 8.3 Architecture of fuzzy inference network

In the following we account for the connection weights between two neurons in adjacent layers and the corresponding I/O relationships. $2md + d$ neurons are arranged in a column to form hidden layer I, and they are divided into d groups, each of which there are $2m + 1$ neurons. In group 1, the neuron p_1 ($p_1 = -m, -m+1, ..., m-1, m$), whose input is x_1, and output is $\widetilde{A}_{1p_1}(x_1)$ is connected with the first one in the input layer; ...; In group d, the neuron p_d ($p_d = -m, -m+1, ..., m-1, m$), whose input is x_d, and output is $\widetilde{A}_{dp_d}(x_d)$ is connected with the d-th neuron in input layer. Hidden layer II consists of $(2m+1)^d$ neurons, in which the neuron $p_1...p_d$ ($p_1, ..., p_d = 0, \pm 1, ..., \pm m$) is connected with the neuron p_1 of group 1, ..., the neuron p_d of group d, respectively in hidden layer I. In hidden layer II, there are d inputs related to neuron $p_1...p_d$: $\widetilde{A}_{1p_1}(x_1), ..., \widetilde{A}_{dp_d}(x_d)$, whose output is $\widetilde{A}_{1p_1}(x_1)\,T \cdots T\,\widetilde{A}_{dj_d}(x_d) = H_{p_1...p_d}(x_1, ..., x_d)$. Also in hidden layer III, there are $(2m+1)^d$ neurons, which are connected with the corresponding neurons in hidden II, respectively, as shown in Figure 8.3. In hidden III, the input of neuron $p_1...p_d$ is $H_{p_1...p_d}(x_1, ..., x_d)$, and corresponding output is $H_{p_1...p_d}(x_1, ..., x_d)\,T\,\widetilde{U}_{p_1...p_d}$. The output neuron accepts $(2m+1)^d$ outputs of hidden layer III as its input. And by operator '$\vee$' determining synthetic operator '$\cup$' we obtain synthetic fuzzy set:

$$\bigcup_{p_1,...,p_d=-m}^{m} H(x_1, ..., x_d)\,T\,\widetilde{U}_{p_1...p_d} \triangleq \widetilde{U},$$

then by generalized defuzzifier $D_e(\cdot)$ establishing crisp output $F_{in}(x_1, ..., x_d) = D_e(\widetilde{U})$.

The adjustable factors of the FINN F_{in} defined by (8.5) are the antecedent fuzzy sets $\widetilde{A}_{ij}$'s and the consequent fuzzy set $\widetilde{U}_{p_1...p_d}$. In application, we always fix the shapes of the antecedent and consequent fuzzy sets, such as triangular or trapezoidal fuzzy numbers, then adjust the parameters related to determine the fuzzy sets for building a FINN [42, 67, 73, 74]. Sometimes, for convenience of simple computation and theoretic analysis, we can construct some fuzzy systems by not abiding by the composite rules of fuzzy inference rules. For instance, by using the composite operation '$\vee - \times$' to determine the fuzzy set related to the fuzzy rules, we can define a fuzzy system, whose output is not determined by the synthesizing fuzzy set $\widetilde{U}$ according to '$\vee$' and a respective defuzzfication operator, but computed as a weighted sum of a family defuzzfying values corresponding to each inference rule, respectively [20, 22, 36, 43, 60, 74]: The fuzzy set determined the fuzzy rule $R_{p_1\ldots p_d}$ is calculated by (6.10), the crisp value corresponding to maximum defuzzfication method is $u_{p_1\ldots p_d} : \widetilde{U}_{p_1\ldots p_d}(u_{p_1\ldots p_d}) = 1$. And the crisp value of the synthesizing fuzzy set is established the following weighted sum:

$$\frac{\sum\limits_{p_1,\ldots p_d=-m}^{m} H_{p_1\ldots p_d}(x_1,\ldots,x_d)u_{p_1\ldots p_d}}{\sum\limits_{p_1,\ldots p_d=-m}^{m} H_{p_1\ldots p_d}(x_1,\ldots,x_d)}, \quad \text{or} \quad \frac{\sum\limits_{p_1,\ldots p_d=-m}^{m} H_{p_1\ldots p_d}(x_1,\ldots,x_d)^{\alpha}u_{p_1\ldots p_d}}{\sum\limits_{p_1,\ldots p_d=-m}^{m} H_{p_1\ldots p_d}(x_1,\ldots,x_d)^{\alpha}}, \tag{8.6}$$

where $0 \le \alpha \le +\infty$. The second part of (8.6) is a generalized Mamdani system (6.11). Similarly we can also construct the fuzzy inference networks corresponding to T–S fuzzy inference rules.

$TR_{p_1\ldots p_d}$: IF x_1 is $\widetilde{A}_{1p_1}$ and ... and x_d is $\widetilde{A}_{dp_d}$ THEN u is $f_{p_1\ldots p_d}(x_1,\ldots,x_d)$,

where $f_{p_1\ldots p_d}(x_1,\ldots,x_d)$ is an adjustable function of input variables $x_1,\ldots,x_d$, which is chosen as a linear function in the following [22]: $f_{p_1\ldots p_d}(x_1,\ldots,x_d) = b_{0;p_1\ldots p_d} + b_{1;p_1\ldots p_d}x_1 + \cdots + b_{d;p_1\ldots p_d}x_d$. With such a restriction, we can write $TR_{p_1\ldots p_d}$ as a Mamdani inference rule form, let $f_{p_1\ldots p_d}(x_1,\ldots,x_d)$ be a singleton fuzzy set $\chi_{\{f_{p_1\cdots p_d}(x_1,\ldots,x_d)\}}$. Then

$TR_{p_1\ldots p_d}$: IF x_1 is $\widetilde{A}_{1p_1}$ and ... and x_d is $\widetilde{A}_{dp_d}$ THEN u is $\chi_{\{f_{p_1\cdots p_d}(x_1,\ldots,x_d)\}}$,

Similarly with (8.5), using the singleton fuzzification and the synthesizing fuzzy operator '$\vee$' we can get the I/O relationship of the generalized FINN corresponding to the fuzzy rule $TR_{p_1\ldots p_d}$ as follows:

$$F_{in}(x_1,\ldots,x_d) = D_e\Big(\bigcup_{p_1,\ldots,p_d=-m}^{m} H_{p_1\ldots p_d}(x_1,\ldots,x_d)\, T\, \chi_{f_{p_1\cdots p_d}(x_1,\ldots,x_d)}\Big). \tag{8.7}$$

By the following Theorem 8.1, if we choose the defuzzification operators $D_e(\cdot)$'s both in (8.5) and (8.7) as the weighted sum (8.6), respectively, the generalized FINN and the corresponding fuzzy system are equivalent.

Theorem 8.1 *Let $D_e(\cdot)$ be a defuzzification operator for defining the FINN $F_{in}(\cdot)$, and $\widetilde{U}$ be a synthesizing fuzzy set determined by (8.4). $D_e(\widetilde{U})$ is defined as the second part of (8.6), $\alpha \in [0, +\infty]$. Then we have*

(i) In Mamdani inference rule $R_{p_1...p_d}$, if we choose the consequent fuzzy set $\widetilde{U}_{p_1...p_d}=\widetilde{U}_{r(p_1,...,p_d)}\in \mathcal{F}([-b, b])$, where $b > 0$ is an adjustable parameter, $r : \mathbb{R}^d \longrightarrow \mathbb{R}$ is an adjustable function, then the generalized FINN $F_{in}(\cdot)$ defined by (8.5) and the generalized Mamdani fuzzy system $M_m(\cdot)$ by (6.11) are functionally equivalent;

(ii) Corresponding to T–S inference rule $TR_{p_1...p_d}$, the I/O relationship $F_{in} : \mathbb{R}^d \longrightarrow \mathbb{R}$ by (8.6) and the generalized T–S fuzzy system by (6.17) are functionally equivalent.

If choosing $D_e(\cdot)$ as the other forms [50], similarly we can show, $F_{in}(\cdot)$ and the corresponding fuzzy system are functionally equivalent [23, 30, 31].

8.1.3 Universal approximation of generalized FINN

Let us now study the representing capability of $F_{in}(\cdot)$, and show some successful applications of the inference network to system identification. The first step to do that is to show that $F_{in}(\cdot)$ constitutes a universal approximator to a class of real functions, and demonstrate the realizing process.

Theorem 8.2 *Let T related to the generalized fuzzy inference network be a continuous t–norm. Then $F_{in}(\cdot)$ is a universal approximator. That is, for arbitrary $\varepsilon > 0$, and each compact set $U \subset \mathbb{R}^d$, if $f : \mathbb{R}^d \longrightarrow \mathbb{R}$ is an arbitrary continuous function, there are $m \in \mathbb{N}$, and fuzzy sets $\widetilde{U}_{p_1...p_d}\in \mathcal{F}(\mathbb{R})$ $(p_1, ..., p_d = 0, \pm 1, ..., \pm m)$, so that*

$$\forall (x_1, ..., x_d) \in U, \ \big|f(x_1, ..., x_d) - F_{in}(x_1, ..., x_d)\big| < \varepsilon.$$

Proof. Since $U \subset \mathbb{R}^d$ is a compact set, there is $a > 0$, so that $U \subset [-a, a]^d$. By the continuity of f on $[-a, a]^d$ we imply, f is uniformly continuous on $[-a, a]^d$. For $\varepsilon > 0$, there exists $\delta > 0$, such that

$$\forall\, (x_1, ..., x_d), (y_1, ..., y_d) \in [-a, a]^d, \ |x_i - y_i| < \delta \ (i = 1, ..., d),$$
$$\Longrightarrow \big|f(x_1, ..., x_d) - f(y_1, ..., y_d)\big| < \frac{\varepsilon}{2}.$$

Choose $m \in \mathbb{N}$, and partition $[-a, a]$ into $2m$ parts: $-a = a_{-m} < a_{1-m} < \cdots < a_{m-1} < a_m = a$. Let m satisfy the condition: $\xi(a, m) < \delta/(2c_0)$, where c_0 is defined by Definition 6.1. We can define the antecedent fuzzy set family as $\{\widetilde{A}_{ij} \ | i = 1, ..., d; j = 0, \pm 1, ..., \pm m\} \subset \widetilde{\mathcal{O}}(a, m)$, so that each $\widetilde{A}_{ij}(\cdot)$ is continuous on $\mathbb{R}$. Given arbitrarily $(x_1, ..., x_d) \in U \subset [-a, a]^d$, $\forall (p_1, ..., p_d) \in N(x_1, ..., x_d)$, by Lemma 6.2, it follows that $\big|x_i - a_{p_i}\big| \leq 2c_0 \cdot \xi(a, m) < \delta$. Thus

$$(p_1, ..., p_d) \in N(x_1, ..., x_d), \Longrightarrow \ \big|f(x_1, ..., x_d) - f\big(a_{p_1}, ..., a_{p_d}\big)\big| < \frac{\varepsilon}{2}. \quad (8.8)$$

By the definition of the generalized defuzzification operator $D_e(\cdot)$, there is $b > 0$, so that for any $p_1, ..., p_d \in \{-m, -m+1, ..., m-1, m\}$, we can obtain the fuzzy set $\widetilde{U}_{p_1...p_d} \in \mathcal{F}([-b, b])$. By Lemma 8.1 we may choose $\widetilde{U}_{p_1...p_d}$ independent of $H_{p_1...p_d}(x_1, ..., x_d)$. Moreover

$$D_e\big(H_{p_1...p_d}(x_1, ..., x_d)\, T\, \widetilde{U}_{p_1...p_d}\big) = f\big(a_{p_1}, ..., a_{p_d}\big). \tag{8.9}$$

Therefore, for any $(x_1, ..., x_d) \in U$ and $(p_1, ..., p_d) \in N(x_1, ..., x_d)$, using (8.8) (8.9) we can conclude that

$$\begin{aligned} &\big|f(x_1, ..., x_d) - D_e\big(H_{p_1...p_d}(x_1, ..., x_d)\, T\, \widetilde{U}_{p_1...p_d}\big)\big| \\ &= \big|f(x_1, ..., x_d) - f\big(a_{p_1}, ..., a_{p_d}\big)\big| < \frac{\varepsilon}{2}. \end{aligned} \tag{8.10}$$

Thus, $\forall (x_1, ..., x_d) \in U$, by (8.10) and Lemma 4.5 easily we have

$$\begin{cases} \Big|f(x_1, ..., x_d) - \bigwedge\limits_{(p_1,...,p_d)\in N(x_1,...,x_d)} \big\{D_e\big(H_{p_1...p_d}(x_1, ..., x_d)\, T\, \widetilde{U}_{p_1...p_d}\big)\big\}\Big| \\ \triangleq \underline{L}_e(x_1, ..., x_d) < \dfrac{\varepsilon}{2}, \\ \Big|f(x_1, ..., x_d) - \bigvee\limits_{(p_1,...,p_d)\in N(x_1,...,x_d)} \big\{D_e\big(H_{p_1...p_d}(x_1, ..., x_d)\, T\, \widetilde{U}_{p_1...p_d}\big)\big\}\Big| \\ \triangleq \overline{L}_e(x_1, ..., x_d) < \dfrac{\varepsilon}{2}. \end{cases} \tag{8.11}$$

Hence for any $(x_1, ..., x_d) \in U$, using Definition 8.1 and (8.11) we get

$$\begin{aligned} &\big|f(x_1, ..., x_d) - F_{in}(x_1, ..., x_d)\big| \\ &= \Big|f(x_1, ..., x_d) - D_e\Big(\bigcup_{p_1,...,p_d=-m}^{m} H_{p_1...p_d}(x_1, ..., x_d)\, T\, \widetilde{U}_{p_1...p_d}\Big)\Big| \\ &= \Big|f(x_1, ..., x_d) - D_e\Big(\bigcup_{(p_1,...,p_d)\in N(x_1,...,x_d)} H_{p_1...p_d}(x_1, ..., x_d)\, T\, \widetilde{U}_{p_1...p_d}\Big)\Big| \\ &\leq \underline{L}_e(x_1, ..., x_d) \vee \overline{L}_e(x_1, ..., x_d) < \varepsilon. \end{aligned}$$

Consequently $F_{in}(\cdot)$ is universal approximator. □

Theorem 8.2 may provide us with the theoretic basis for the application of generalized FINN's to many real fields, such as system modeling, system identification, image processing and pattern recognition and so on. Next let us take a few of simulation examples to demonstrate the application of the FINN $F_{in}(\cdot)$ in system identification. At first we suppose the antecedent fuzzy set family is $\{\widetilde{A}_{ij}\ |i = 1, ..., d;\ j = -m_1, -m_1+1, ..., m_2-1, m_2\} \subset \widetilde{\mathcal{O}}(a, m_1+m_2)$, and $\widetilde{A}_{ij} = \widetilde{A}_{1j}\ \ (i = 1, ..., d)$, $\widetilde{A}_{1j}$ is the fuzzy set $\widetilde{A}_j$ shown in Figure 8.1.

Example 8.2 Let $d=2$, the error bound $\varepsilon=0.2$. Given the compact set $C=[-1,\ 1]^d$, partition $[-1,\ 1]$ identically into $2m$ parts, that is, in Figure 8.1, we choose the following parameters:

$$a=1,\ m_1=m_2=m,\ a_j=\frac{j}{m}\ (j=0,\ \pm1,...,\pm m).$$

The fuzzy set $\widetilde{A}_{ij}\ \ (i=1,2;\ j=0,\pm1,...,\pm m)$ is defined by the translation of the triangular fuzzy number $\widetilde{A}(\cdot)$:

$$\widetilde{A}(t)=\begin{cases} mt+1, & -\dfrac{1}{m}\le t\le 0,\\ 1-mt, & 0<t\le\dfrac{1}{m},\\ 0, & \text{otherwise.}\end{cases}$$

$$\begin{cases} \widetilde{A}_{1(-m)}(t)=\begin{cases} 1-m-mt, & -1\le t\le -1+\dfrac{1}{m},\\ 0, & \text{otherwise;}\end{cases}\\ \widetilde{A}_{1m}\ (t)=\begin{cases} mt+1-m, & 1-\dfrac{1}{m}\le t\le 1,\\ 0, & \text{otherwise;}\end{cases}\\ \widetilde{A}_{1j}\equiv\widetilde{A}_{2j};\ \ \widetilde{A}_{1j}(t)=\widetilde{A}\Big(t-\dfrac{j}{m}\Big)\ (j=-m+1,...,m-1).\end{cases} \tag{8.12}$$

For convenience for application, we determine the crisp value $D_e(\widetilde{U})$ of fuzzy set $\widetilde{U}$ by (8.6), $\widetilde{U}$ is the synthesizing fuzzy set corresponding to Mamdani inference rule $R_{p_1...p_d}$ or T–S rule $TR_{p_1...p_d}$. And the continuous t−norm T is product '×'. Obviously if $\alpha=1$, for any $x,\ x_1,...,x_d\in[-1,\ 1]$ we have

$$\begin{cases} \sum\limits_{j=-m}^{m}\widetilde{A}_{ij}\,(x)=1,\\ \sum\limits_{p_1,...p_d=-m}^{m} H_{p_1...p_d}(x_1,...,x_d)=\prod\limits_{i=1}^{d}\Big(\sum\limits_{j=-m}^{m}\widetilde{A}_{ij}\,(x_i)\Big)=1.\end{cases} \tag{8.13}$$

Define the continuous function $f(\cdot,\ \cdot)$ as follows:

$$f(x_1,\ x_2)=\sin(10x_1+15x_2)\ (x_1,\ x_2\in[-1,\ 1]).$$

Let $\delta=\varepsilon/25$, we get, if $|x_1^0-y_1^0|<\delta,\ |x_2^0-y_2^0|<\delta$, it follows that

$$|f(x_1^0,\ x_2^0)-f(y_1^0,\ y_2^0)|\le 10|x_1^0-y_1^0|+15|x_2^0-y_2^0|<25\delta=\varepsilon.$$

Let $m \in \mathbb{N}: \ c_0/m = 2/m \leq \varepsilon/25$, i.e. $m \geq 50/\varepsilon = 250$. Choose $m = 250$. So by Remark 6.2 and (8.6) we have

$$
\begin{aligned}
F_{in}(x_1, x_2) &= \frac{\sum\limits_{p_1,p_2=-250}^{250} H_{p_1p_2}(x_1, x_2)^{\alpha} f(\frac{p_1}{250}, \frac{p_2}{250})}{\sum\limits_{p_1,p_2=-250}^{250} H_{p_1p_2}(x_1, x_2)^{\alpha}} \\
&= \frac{\sum\limits_{p_1,p_2=-250}^{250} H_{p_1p_2}(x_1, x_2)^{\alpha} \sin(\frac{p_1}{25} + \frac{3p_2}{50})}{\sum\limits_{p_1,p_2=-250}^{250} H_{p_1p_2}(x_1, x_2)^{\alpha}},
\end{aligned}
$$

$$H_{p_1p_2}(x_1, x_2) = \widetilde{A}_{1p_1}(x_1) \times \widetilde{A}_{2p_2}(x_2) \ (p_1, p_2 = 0, \pm 1, ..., \pm m).$$

We can illustrate the approximating surfaces of $Z = F_{in}(x_1, x_2)$ as Figure 8.4 when $\alpha = 1$, $1/2$ and $\alpha = 2$, respectively. Also the original surface of $f(\cdot, \cdot)$ is shown in Figure 8.4, from which we can see the high approximting accuracy at each point.

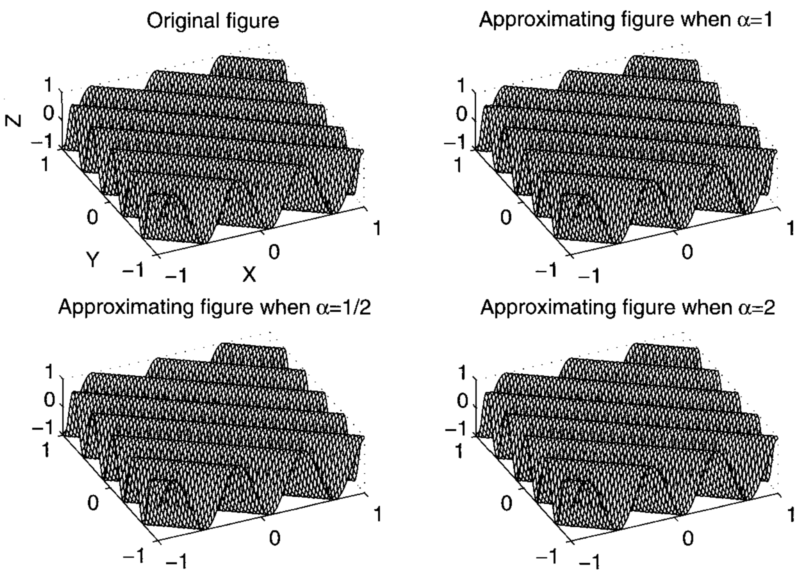


Figure 8.4 Original surface and approximating surfaces when $\alpha = 1$, 0.5, 2

8.1.4 Simulation of system identification

By Example 8.2 and Theorem 8.2, we can utilize generalized FINN's to express a given function approximately. This is a static system identification.

With the same reason the FINN's can be used as a identification model for dynamic systems. Moreover, the identification results related are much more advantageous than ones by neural networks [48, 67]. In the following let us study the identification capability of the generalized FINN $F_{in}(\cdot)$ based on the T–S inference rule $TR_{p_1...p_d}$ and the defuzzifier as (8.6).

$$F_{in}(x_1, ..., x_d) = \frac{\sum\limits_{p_1,...,p_d=-m}^{m} H_{p_1...p_d}(x_1, ..., x_d)^{\alpha} \cdot \sum\limits_{i=0}^{d} b_{i;p_1...p_d} x_i}{\sum\limits_{p_1,...,p_d=-m}^{m} H_{p_1...p_d}(x_1, ..., x_d)^{\alpha}}, \tag{8.14}$$

where $x_0 \equiv 1$. (8.14) is a generalized T–S fuzzy system. By Remark 6.2, if a series of output signals corresponding a identification model are known, we can use the FINN's to simulate the unknown system by adjusting the antecedent fuzzy sets related, where $d = 1$.

The discrete-time system can be described by the following nonlinear difference equation:

$$z(k+1) = f\big(z(k), ..., z(k-p+1);\ x(k), ..., x(k-q+1)\big), \tag{8.15}$$

where $x(j)$ $(j = k-q+1, ..., k)$ represents the input of the SISO system at time j, $z(k)$ is the output, and f is the unknown function to be identified, $p,\ q \in \mathbb{N}$.

1. Parallel Identification Model: The structure of identification model can be described by the following equation [48]:

$$Z(k+1) = F\big(Z(k), ..., Z(k-p+1),\ x(k), ..., x(k-q+1)\big), \tag{8.16}$$

where $F(\cdot)$ represents the function determined by a generalized inference network, Z is the output of the identification system. The learning on line aims mainly on determining the antecedent fuzzy set $\widetilde{A}_{1j}$ step by step, where $j = -m_1,\ -m_1+1, ..., 0,\ 1, ..., m_2)$.

Algorithm 8.1 Adjusting algorithm for antecedent fuzzy sets.

Step 1. Give the initial inputs $x(1),\ x(2)$, and put $n = 2$.

Step 2. Rank $\{x(1), ..., x(n)\}$ with the increasing order as $\{u_1, ..., u_n\}$. Let $b_1 = u_1$, and

$$b_{i+1} = \min\{u_j \big| u_j > b_i + \delta\}\ (i = 1, 2, ..., \gamma), \tag{8.17}$$

where $\delta > 0$ is a constant, $\gamma \le n$. Set $\gamma > 1$. And write $\{b_1, ..., b_\gamma\}$ $(\gamma \le n)$ as the following form, and using Figure 8.1 we define $\widetilde{A}_{1j}$:

$$\{b_1, ..., b_\gamma\} = \begin{cases} \{a_{-m_2}, a_{-m_2+1}, ..., a_{m_1}\}, & b_1 < 0, \\ \{a_0, a_1, ..., a_{m_1}\}, & b_1 = 0, \\ \{a_1, ..., a_{m_1}\}, & b_1 > 0 \end{cases}$$

Step 3. Discriminate $n \ge M$? If yes go to the following step; otherwise let $n = n+1$, and select the input $x(n)$, go to Step 2.

Step 4. Output the antecedent fuzzy set $\{\widetilde{A}_{1j}\}$.

Example 8.3 Let the system identified satisfy the following difference equation [48]:

$$\begin{cases} z(k+1) = 0.3 \cdot z(k) + 0.6 \cdot z(k-1) + g(x(k)), \\ x(k) = \sin\left(\dfrac{2\pi k}{250}\right), \ k = 1, 2, ..., \end{cases} \tag{8.18}$$

where $g(\cdot)$ is a unknown system. And let the function be as follows:

$$g(x) = 0.6 \cdot \sin(\pi x) + 0.3 \cdot \sin(3\pi x) + 0.1 \cdot \sin(5\pi x).$$

Suppose $x(k)$ be the input of the system at k. Choose $\alpha = 1, \ \delta = 0.005$, and the learning iteration number $M = 200$. The system input is $x(k) = \sin(2\pi k/250)$ in the learning procedure. Using Algorithm 8.1 we can obtain the antecedent fuzzy set family $\{\widetilde{A}_{1j} \ | j = -m_1, \ -m_1 + 1, ..., 0, \ 1, ..., m_2\} \subset \widetilde{\mathcal{O}} \ (1, m_1 + m_2)$. Then $\forall x \in [a_{-m_1}, a_{m_2}]$, we have, $\sum\limits_{j=-m_1}^{m_2} \widetilde{A}_{1j}(x) \equiv 1$. To identify (8.18), we employ Theorem 8.2, Remark 6.2, (8.16) and (8.18):

$$\begin{cases} Z(k+1) = 0.3 \cdot Z(k) + 0.6 \cdot Z(k-1) + F_{in}(x(k)), \\ F_{in}(x(k)) = \sum\limits_{j=-m_1}^{m_2} \widetilde{A}_{1j}(x(k)) \cdot \big(Z(k+1) - 0.3 \cdot Z(k) - 0.6 \cdot Z(k-1)\big). \end{cases} \tag{8.19}$$

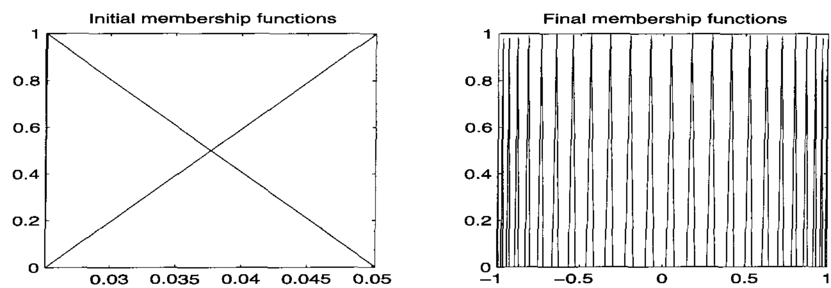


Figure 8.5 Membership function curves of initial and ultimate iteration step for online learning

By learning algorithm 8.1, the number of antecedent fuzzy sets is $m_1+m_2 = 113$. Figure 8.5 demonstrates the initial two antecedent fuzzy sets and the ultimate antecedent fuzzy sets in the learning process, where for clarity we show only a part of ultimate membership function curves, i.e. one curve in every five ones.

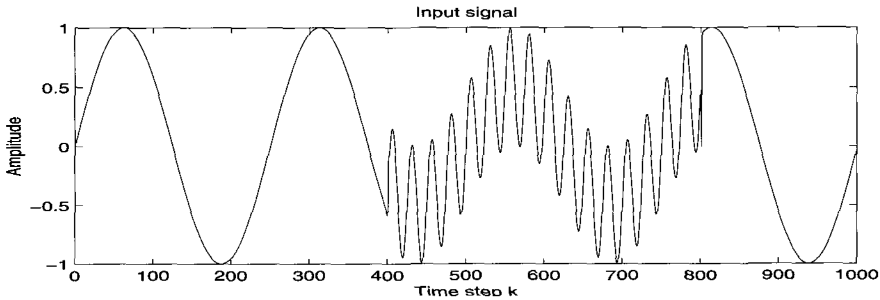


Figure 8.6 Input curve of the system

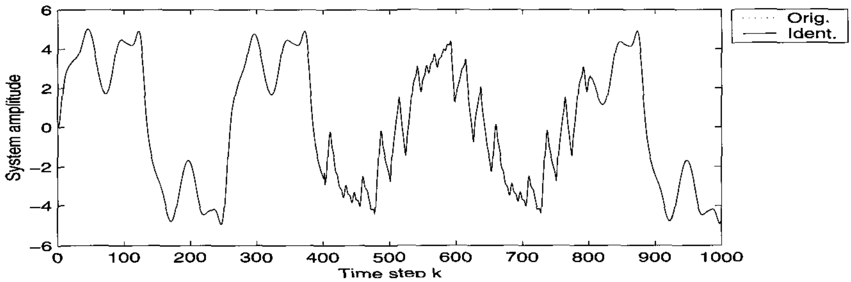


Figure 8.7 Outputs of system (imaginary line) and generalized FINN identifying model (dotted line) for parallel format

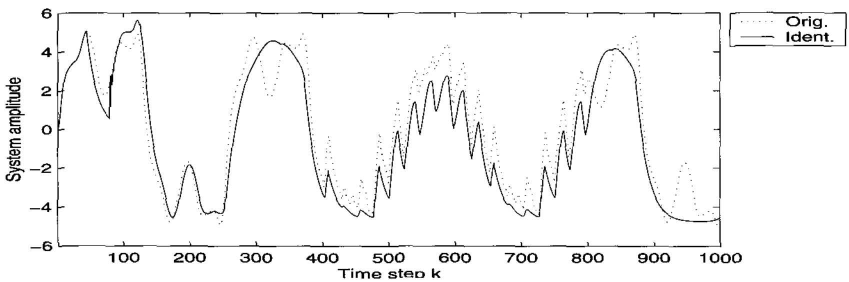


Figure 8.8 System output(imaginary line) and output of Mamdani fuzzy system identifying model with parallel format(dotted line)

To show good identifying performance of the model (8.19) we change the system input after the model is trained (i.e. the time step $k > 200$). The variant inputs are presented the system with the following law:

$$x(k) = \begin{cases} \sin\left(\frac{2k\pi}{250}\right), & 201 \le k \le 400, \\ 0.5 \cdot \sin\left(\frac{2k\pi}{250}\right) + 0.5 \cdot \sin\left(\frac{2k\pi k}{25}\right), & 401 \le k \le 800, \\ \sin\left(\frac{2k\pi}{250}\right), & 801 \le k \le 1000. \end{cases}$$

These input variety is illustrated in Figure 8.6. The outputs of the original system and the trained identifying system, which correspond to these varying inputs are shown in Figure 8.7, from which we can see the identification error is very small even when the system input is changed. Thus the identifying model (8.19) possesses high approximation accuracy.

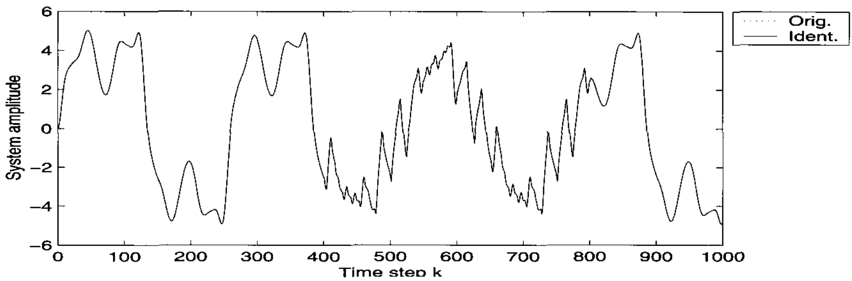


Figure 8.9 System output (imaginary line) and output of generalized FINN identifying model with series–parallel format (dotted line)

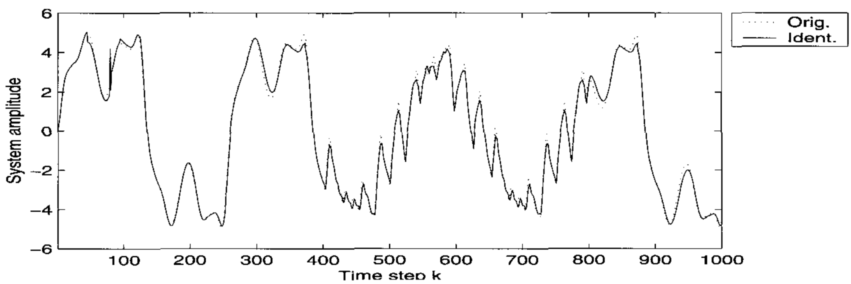


Figure 8.10 Output of system (imaginary line) and Mamdani fuzzy system identifying model(dotted line) with parallel–series format

To test further the identifying capability of the identification model (8.19), we compare the identification performance with that in [67], which is based on Mamdani fuzzy systems with Gaussian membership function fuzzy sets, under the same conditions.

A Gaussian type membership function may be established uniquely by its center x_g, its width σ_g. So by designing the learning algorithms for a family of x_g's and σ_g's, and the coefficients y's. we can establish a Mamdani fuzzy systems with Gaussian membership function fuzzy sets.

Let the number of membership functions be 40. Then there are 120 parameters to be established by online BP algorithm. Figure 8.8 shows the curves corresponding to the original system and the identifying model. Easily we can see, the error resulted by Mamdani fuzzy system based on Gaussian type fuzzy sets is much lager, and when the system input is changed, the identifying system can not simulate the original system very well and it can not follow the output of original system when $k \geq 300$. If using the crisp neural networks, be [67] we can imply that the identifying performance is disadvantage over that of Mamdani fuzzy system based on Gaussian type fuzzy sets. So with the parallel format the identifying performance of the generalized fuzzy inference network identifying model is best.

2. Series–parallel Identification Model: In contrast to the parallel identification model described above, in the series–parallel model the output of the original system (rather than the identification model) is fed back into the identification. The identification model can be expressed as follows [48, 67]:

$$Z(k+1) = 0.3 \cdot z(k) + 0.6 \cdot z(k-1) + F\big(x(k)\big), \tag{8.20}$$

where $F(\cdot)$ is a generalized fuzzy inference network, or a Mamdani system based on Gaussian type fuzzy sets, or a feedforward neural network. It is obvious that the corresponding identifying performance is much better than that of the parallel format because of the use of original system outputs $z(k)$, $z(k-1)$,. Figure 8.9 and Figure 8.10 demonstrate the identifying curves corresponding to generalized fuzzy inference network and Mamdani fuzzy system, respectively. Here we can find, the performance of generalized fuzzy inference network is best, and feedforward neural networks identification can not simulates the system [48, 67].

If the signal sequence to be processed, for example digital image contains some noises, we can treat the generalized FINN by designing learning algorithms for the weight coefficient $u_{p_1 \dots p_d}$ in (8.6) as a noise filter, which is the subject in the following two sections.

§8.2 Representation of two–dimensional image by FINN

In the conventional image theory, we utilize a completely orthogonal function basis to establish some models for digital image representations, and then develop linear theory for image processing [5]. Although we can employ some classical mathematical tools such as Fourier transform and statistics and so on to process image linear models with a systematic approach, linear tools may solve after all only a small of problems related to image processing, and most of them are dealt with only by nonlinear techniques [4, 55]. In the section we present the approximate representation of a 2-D digital image by FINN's. By Theorem 8.2, the generalized FINN's can be universal approximators, so with the given accuracy we can code a 2-D image as the connection weights of a FINN. Moreover we can establish some optimal filters by designing learning algorithms based on minimizing the absolute error.

8.2.1 Local FNN representation of image

The gray level of a digital image at a given point can be expressed as the I/O relationship of a FINN determined by a local neighborhood of the point. To this end we introduce a small operating windows which may slide on the whole image window, and then use the gray levels in the small window to develop some suitable fuzzy inference rules. Thus, a FINN can be defined and a local representation of the 2-D digital image is established. If the image is noise-free the representation is accurate, and if the image is noise, the representation can also serve as a filter.

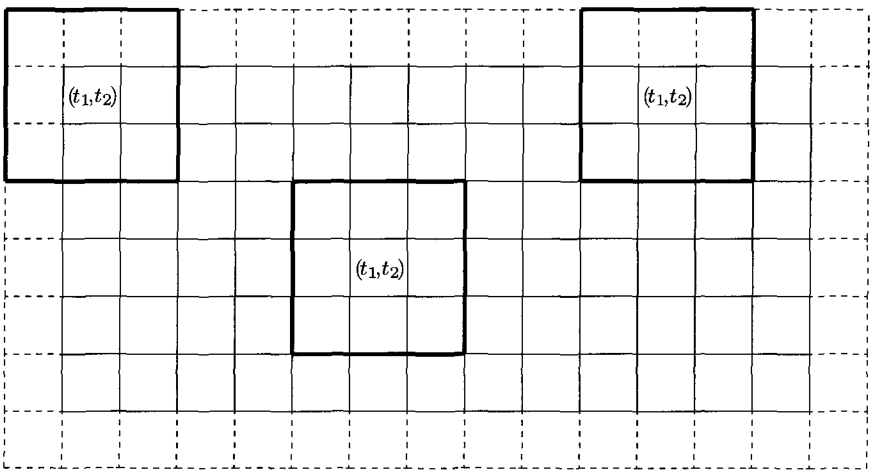


Figure 8.11 Sample window for local image representation

Let $X = (s(t_1,\, t_2))_{N_1 \times N_2}$ be a 2-D digital image with $N_1 \times N_2$ pixels and L gray levels $0, 1, ..., L-1$, that is, $s(t_1,\, t_2) \in \{0, 1, ..., L-1\}$ $(0 \le t_1 \le N_1 - 1,\ 0 \le t_2 \le N_2 - 1)$. For preserving the edges and locally fine structures of the image, we adopt a small window with $n_1 \times n_2$ samples to determine the representation of the gray level value, where $n_1 \ll N_1,\ n_2 \ll N_2,\ n_1, n_2$ are usually odd numbers. Let $\underline{X}(t_1,\, t_2) = \big(s(t_1+k,\, t_2+l)\big)_{n_1 \times n_2}$, where $k = -(n_1-1)/2, ..., (n_1-1)/2,\ l = -(n_2-1)/2, ..., (n_2-1)/2$, and $(t_1,\, t_2)$ is located on the central cell of the sample window as Figure 8.11. Considering $\mu_{n_1 \times n_2}$ is the collection of all matrices with n_1 rows and n_2 columns, we denote $\underline{X}(t_1,\, t_2)$ by $\{s(t_1+k,\, t_2+l) :\ -(n_1-1)/2, ..., (n_1-1)/2;\ -(n_2-1)/2, ..., (n_2-1)/2\}$. To guarantee the small operating window to slide well on the large image window $X = \big(s(t_1,\, t_2)\big)_{N_1 \times N_2}$, we must extend X along the edge, as Figure 8.11. For $t_2 = 0, 1, ..., N_2 - 1,\ t_1 = -1, 0, 1, ..., N_1 - 1,\ s(-1,\, t_2) = s(0,\, t_2),\ s(N_1,\, t_2) = s(N_1 - 1,\, t_2);\ s(t_1,\, -1) = s(t_1,\, 0),\ s(t_1,\, N_2) = s(t_1,\, N_2 - 1)$.

Define rank operator $R :\ \mu_{n_1 \times n_2} \longrightarrow \mathbb{R}^{2^{\mathbb{R}}}$ as follows: for any $\underline{X}(t_1,\, t_2) = (s(t_1+k, t_2+l) :\ -(n_1-1)/2, ..., (n_1-1)/2;\ -(n_2-1)/2, ..., (n_2-1)/2) \in$

$\mu_{n_1\times n_2}$, we have

$$\begin{aligned}R(\underline{X}(t_1,\,t_2)) &= \{s_0(t_1,\,t_2),\ s_1(t_1,\,t_2),...,s_{n_1\times n_2-1}(t_1,\,t_2):\\ &\qquad s_0(t_1,\,t_2)\le\cdots\le s_{n_1\times n_2-1}(t_1,\,t_2)\}\\ &\triangleq \big(R(\underline{X}(t_1,\,t_2))_0,\ R(\underline{X}(t_1,\,t_2))_1,...,R(\underline{X}(t_1,\,t_2),)_{n_1\times n_2-1}\big),\end{aligned}$$

where $R\big(\underline{X}(t_1,\,t_2)\big)_i= s_i(t_1,\,t_2)$ $(i=0,1,...,n_1\times n_2-1)$, and $\mathbb{R}^{2^{\mathbb{R}}}$ means the collection of all subsets of $\mathbb{R}$.

In order to build the fuzzy inference rules for retrieving the noise image, we at first fuzzify the gray level $s(t_1,\,t_2)$ of the pixel $(t_1,\,t_2)$ as the trapezoidal fuzzy number $\tilde{s}\,(t_1,\,t_2)$ $(t_1,\,t_2=0,1,...,L-1)$, as shown Figure 8.12. They may represent the linguistic concepts for the gray levels of the image, such as 'dark', 'very dark', 'darker', 'poorly dark', 'medium', 'very bright', 'bright', 'brighter' and 'poorly bright', and so on.

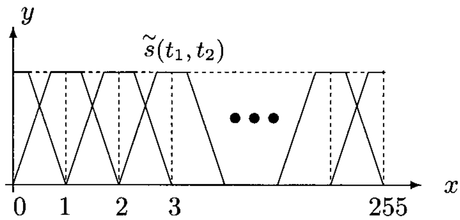


Figure 8.12 Fuzzy numbers portraying gray level

If X is uncorrupted, all pixels inside the sliding small window $\underline{X}(t_1,\,t_2)$ may be assumed to have approximately equal gray levels [3]. So for each $(t_1,\,t_2)$, let $\triangle s(t_1,\,t_2)$ denote the difference between the gray levels of central pixel $(t_1,\,t_2)$ and the median value pixel in $\underline{X}(t_1,\,t_2)$, that is, we have $\triangle s(t_1,\,t_2)=\big|s(t_1,\,t_2)-s_{(n_1\times n_2+1)/2}(t_1,\,t_2)\big|$. Let $\delta s(t_1,\,t_2)$ mean the neighboring difference of $s(t_1,\,t_2)$ that is, if let $s(t_1,\,t_2)=s_k(t_1,\,t_2)$, the $k+1$-th element of $R\big(\underline{X}(t_1,\,t_2)\big)$, then

$$\delta s(t_1,\,t_2)=\frac{1}{2}\Big(\big|s(t_1,\,t_2)-s_{k_f}(t_1,\,t_2)\big|+\big|s(t_1,\,t_2)-s_{k_b}(t_1,\,t_2)\big|\Big),$$

where $k_f=(k-1)\vee 0$, $k_b=(k+1)\wedge(n_1\times n_2-1)$. In impulse noise environment, both $\triangle s(t_1,\,t_2)$ and $\delta s(t_1,\,t_2)$ can be assumed to be 'small' if the pixel $(t_1,\,t_2)$ is noise-free [3], otherwise it is reasonable to believe $(t_1,\,t_2)$ is corrupted. The fuzzy set 'small' and 'large' are denoted as $\tilde{S}$, $\tilde{L}$, respectively, as shown Figure 8.13.

When $(t_1,\,t_2)$ is uncorrupted, the filter should preserve the corresponding gray level, otherwise we have to choose the gray level as the median one of $\underline{X}(t_1,\,t_2)$. This is because, although the median filter may vary the original structure of the image, the probability of the median gray level of $\underline{X}(t_1, t_2)$ to be corrupted is minimum [10, 16]. According to such a principle, we can build the following inference rules:

$$R_{00}: \text{IF } \Delta s(t_1, t_2) \text{ is } \tilde{S} \text{ and } \delta s(t_1, t_2) \text{ is } \tilde{S} \text{ THEN } y(t_1, t_2) \text{ is } \tilde{s}\,(t_1, t_2);$$
$$R_{01}: \text{IF } \Delta s(t_1, t_2) \text{ is } \tilde{S} \text{ and } \delta s(t_1, t_2) \text{ is } \tilde{L} \text{ THEN } y(t_1, t_2) \text{ is } \tilde{m}\,(t_1, t_2);$$
$$R_{10}: \text{IF } \Delta s(t_1, t_2) \text{ is } \tilde{L} \text{ and } \delta s(t_1, t_2) \text{ is } \tilde{S} \text{ THEN } y(t_1, t_2) \text{ is } \tilde{m}\,(t_1, t_2);$$
$$R_{11}: \text{IF } \Delta s(t_1, t_2) \text{ is } \tilde{L} \text{ and } \delta s(t_1, t_2) \text{ is } \tilde{L} \text{ THEN } y(t_1, t_2) \text{ is } \tilde{m}\,(t_1, t_2),$$

where $\tilde{m}(t_1, t_2) = \tilde{s}_{(n_1\times n_2+1)/2}(t_1, t_2)$. From now on let $m(t_1, t_2)$ be a mediate gray level in $\underline{X}(t_1, t_2)$, that is, $m(t_1, t_2) = s_{(n_1\times n_2+1)/2}(t_1, t_2)$. By (8.6), the local FINN representation of the image X on the window $\underline{X}(t_1, t_2)$ is defined as follows:

$$\begin{aligned} y(t_1, t_2) &= \frac{\sum\limits_{j_1,j_2=0}^{1} w_{j_1j_2}\big(\tilde{A}_{1j_1}(x_1)\,T\,\tilde{A}_{2j_2}(x_2)\big)^{\alpha}}{\sum\limits_{j_1,j_2=0}^{1}\big(\tilde{A}_{1j}(x_1)\,T\,\tilde{A}_{2j_2}(x_2)\big)^{\alpha}} \\ &= \frac{\sum\limits_{j_1,j_2=0}^{1} w_{j_1j_2}\Big(\tilde{A}_{1j_1}(\Delta s(t_1, t_2))\,T\,\tilde{A}_{2j_2}(\delta s(t_1, t_2))\Big)^{\alpha}}{\sum\limits_{j_1,j_2=0}^{1}\Big(\tilde{A}_{1j_1}(\Delta s(t_1, t_2))\,T\,\tilde{A}_{2j_2}(\delta s(t_1, t_2))\Big)^{\alpha}}, \end{aligned} \tag{8.21}$$

where $x_1 = \Delta s(t_1, t_2)$, $x_2 = \delta s(t_1, t_2)$, moreover

$$\tilde{A}_{10}=\tilde{A}_{20}=\tilde{S},\ \tilde{A}_{11}=\tilde{A}_{21}=\tilde{L},\ w_{j_1j_2} = \begin{cases} s(t_1, t_2), & j_1 = j_2 = 0; \\ m(t_1, t_2), & \text{otherwise}, \end{cases} \tag{8.22}$$

and $\alpha:\ 0 \le \alpha \le +\infty$ is an adjustable parameter [17].

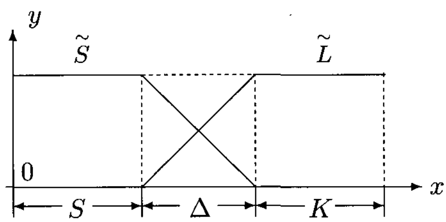


Figure 5.13 Deviation fuzzy numbers

Also we can adjust the output of (8.21) by changing the values of S, Δ and K in Figure 8.13. If the image is uncorrupted, we let $S \geq 80$ and $\Delta \leq 1$, or α be sufficiently large for the image with 256 gray levels. In the noise environment, (8.21) may be utilized as a filter, which is called FNN filter. And S, Δ and K may be determined by some adaptive learning algorithms.

Next, we employ the noise-free image 'Lenna' to show the effectiveness of the local representation (8.21) to reconstruct the original image.

Example 8.4 Let $X = \big(s(t_1, t_2)\big)_{512\times 512}$ be Lenna's image, a 2-D digital image sized 512×512 pixels with $L = 256$ gray levels. So

$$\forall (t_1, t_2) : \ 0 \le t_1 \le 511,\ 0 \le t_2 \le 511,\ s(t_1, t_2) \in \{0, 1, ..., 255\}.$$

We may employ the FINN's determined by (8.21) to express accurately the image X if X is noise-free. To execute the I/O relationship defined as (8.21), a sliding 3×3 sample window is employed to determine the local area, on which the local representation as (8.21) of the image is defined. That is, $n_1 = n_2 = 3$. So $n_1 \times n_2 - 1 = 8$. We choose $S = 100$, $\Delta \le 1$, and the t-norm $T = \times$, $\alpha = 1$. By (8.21) (8.22) and considering the fact:

$$\begin{aligned}&\widetilde{A}_{10}\big(\Delta s(t_1, t_2)\big) + \widetilde{A}_{11}\big(\Delta s(t_1, t_2)\big)\\ &= \widetilde{A}_{20}\big(\delta s(t_1, t_2)\big) + \widetilde{A}_{21}\big(\delta s(t_1, t_2)\big) = 1,\end{aligned}$$

we can obtain the local representation of the Lenna's image as follows:

$$\begin{aligned} y(t_1, t_2) &= \frac{\sum\limits_{j_1, j_2 = 0}^{1} w_{j_1 j_2} \cdot \big(\widetilde{A}_{1j_1}(\Delta s(t_1, t_2) \times \widetilde{A}_{2j_2}(\delta s(t_1, t_2))\big)^{\alpha}}{\sum\limits_{j_1, j_2 = 0}^{1} \big(\widetilde{A}_{1j_1}(\Delta s(t_1, t_2)) \times \widetilde{A}_{2j_2}(\delta s(t_1, t_2))\big)^{\alpha}} \\ &= m(t_1, t_2)\Big\{\widetilde{A}_{10}\big(\Delta s(t_1, t_2)\big) \cdot \widetilde{A}_{21}\big(\delta s(t_1, t_2)\big) \\ &\qquad + \widetilde{A}_{11}\big(\Delta s(t_1, t_2)\big) \cdot \widetilde{A}_{20}\big(\delta s(t_1, t_2)\big) \\ &\qquad + \widetilde{A}_{11}\big(\Delta s(t_1, t_2)\big) \cdot \widetilde{A}_{21}\big(\delta s(t_1, t_2)\big)\Big\} \\ &\quad + s(t_1, t_2) \cdot \widetilde{A}_{10}\big(\Delta s(t_1, t_2)\big) \cdot \widetilde{A}_{20}\big(\delta s(t_1, t_2)\big). \end{aligned}$$

Lenna's original image is shown in Figure 8.14. And Figure 8.15 gives the reconstruction image determined by $Y = \big(y(t_1, t_2)\big)_{512\times 512}$. By the comparison between Figure 8.14 and Figure 8.15 we may find that the representation is accurate. In fact, we can prove the following fact:

$$\forall t_1, t_2 \in \{0, 1, ..., 511\},\ \big|y(t_1, t_2) - s(t_1, t_2)\big| = 0,$$

that is, the error related may vanishes. Thus the image is completely reconstructed.

Figure 8.14 Lenna's original image

Figure 8.15 Lenna's reconstruction image

8.2.2 Optimal FNN filter

In the impulse noise environment, if let in (8.21) (8.22) $S = \Delta = 0$, the FNN filter becomes the median filter; if let $\Delta = K = 0$, the filter will leave the noise image unremoved. So it is important to adjust the values of S, Δ and K, so that the corresponding FNN filter possesses the strong filtering capability.

For simplicity we from now on assume that Δ is a positively small constant, such as $\Delta = 0.9$. Thus, it suffices to design the learning algorithm for S since $K = 255 - S - \Delta$ for the image with 256 gray levels. To this end, the mean absolute error (MAE) criterion is employed to determine S to minimize the filtering error, consequently the optimal FNN filter may be constructed.

Assume that $X = \big(s(t_1, t_2)\big)_{N_1 \times N_2}$ is a 2-D image corrupted by impulse noise with the probability p, that is, if we denote the (t_1, t_2)-th pixel in the uncorrupted image as $s^0(t_1, t_2)$, and let $X^0 = \big(s^0(t_1, t_2)\big)_{N_1 \times N_2}$, then $s(t_1, t_2)$ can take three possible values:

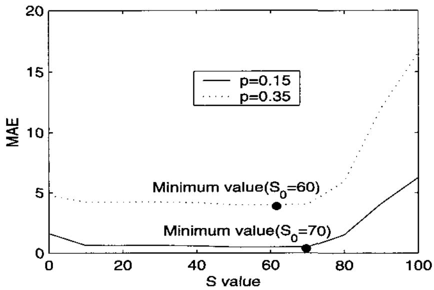
$$s(t_1, t_2) = \begin{cases} s_{\max} & \text{with probability } p/2, \\ s^0(t_1, t_2) & \text{with probability } 1-p, \\ s_{\min} & \text{with probability } p/2, \end{cases} \tag{8.23}$$

where $s_{\max}$ is maximum value in X^0 and appears as white dot; $s_{\min}$ is minimum value in X^0 and appears as black dot.

Suppose that $Y = \big(y(t_1, t_2)\big)_{N_1 \times N_2}$ is the filtering image of X determined by (8.21) (8.22), then Y either is identical to the image X or generates a difference vector $\big\{s(t_1, t_2) - y(t_1, t_2) | t_1 = 0, 1, ..., N_1 - 1,\ t_2 = 0, ..., N_2 - 1\big\}$. Thus, the MAE generated by using Y to estimate X is as follows:

$$E(S) = \|X^0(\cdot, \cdot) - Y(\cdot, \cdot)\| \triangleq \frac{1}{N_1 \times N_2}\Big(\sum_{t_1, t_2} \big|s^0(t_1, t_2) - y(t_1, t_2)\big|\Big), \tag{8.24}$$

where $X^0 = \left(s^0(t_1,\, t_2)\right)_{N_1 \times N_2}$ is the desired image. As the value of S in Figure 8.13 is suitably adjusted, the output of (8.21) is also controlled rationally. And the corresponding MAE is minimized. If there is $S_0 \in \mathbb{R}_+$, so that $E(S_0) = \min\limits_{S}\{E(S)\}$, then representation as (8.21) with the trapezoidal fuzzy numbers $\tilde{S},\ \tilde{L}$ derived by S_0 is called the optimal FNN, which is also called the optimal FNN filter.



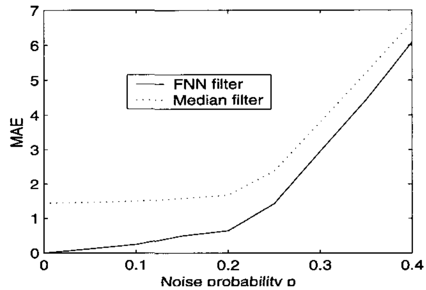


Figure 8.16 Figure 8.17

Figure 8.16: Iteration of S for 'cameraman'; Figure 8.17: MAE of median and FNN filters for 'cameraman'

When $S = 0$, the filter (8.21) becomes a median filter, by which most spike noises are filtered. However, the fine structure of the image is also removed. As the value of S increases, the filter leaves more and more gray levels that are uncorrupted unchanged, consequently the MAE E decreases. But when the value of S is larger than a certain threshold, the filter will leave some corrupted gray levels unremoved, further more and more such noise-corrupted ones in the filtering image arise as S becomes larger. Thus, there must be a S_0 to minimize the MAE $E(S)$. By the following learning, we may find such a S_0.

Algorithm 8.2 Optimal FNN filter:

Step 1. Estimate the amplitude H of the noise image, i.e. $H = d_{\max} - d_0^0$, where d_0^0 is approximately minimum gray level of the image.

Step 2. Put t=0, and let $S(t) = 0,\ E(S(-1)) = H/2$;

Step 3. By (8.21) calculate $y(t_1,\, t_2)$, and then obtain $E(S[t])$ by (8.24);

Step 4. Construct the following iteration scheme and calculate $E(S[t+1])$ by (8.21) (8.24):

$$S[t+1] = S[t] - \beta \cdot \left(E(S[t]) - E(S[t-1])\right);$$

Step 5. Discriminate whether $E(S[t+1]) \leq E(S[t])$ or not? If yes, let $t = t+1$, and go to Step 3; otherwise output $S[t]$ as the optimal S_0 and stop.

In step 4, $\beta > 0$ is a given constant. To avoid that the algorithm fall into the locally minimum points, we may choose two or three different β values to operate the algorithm. Further we continue a few of iterations when the condition in step 5 holds.

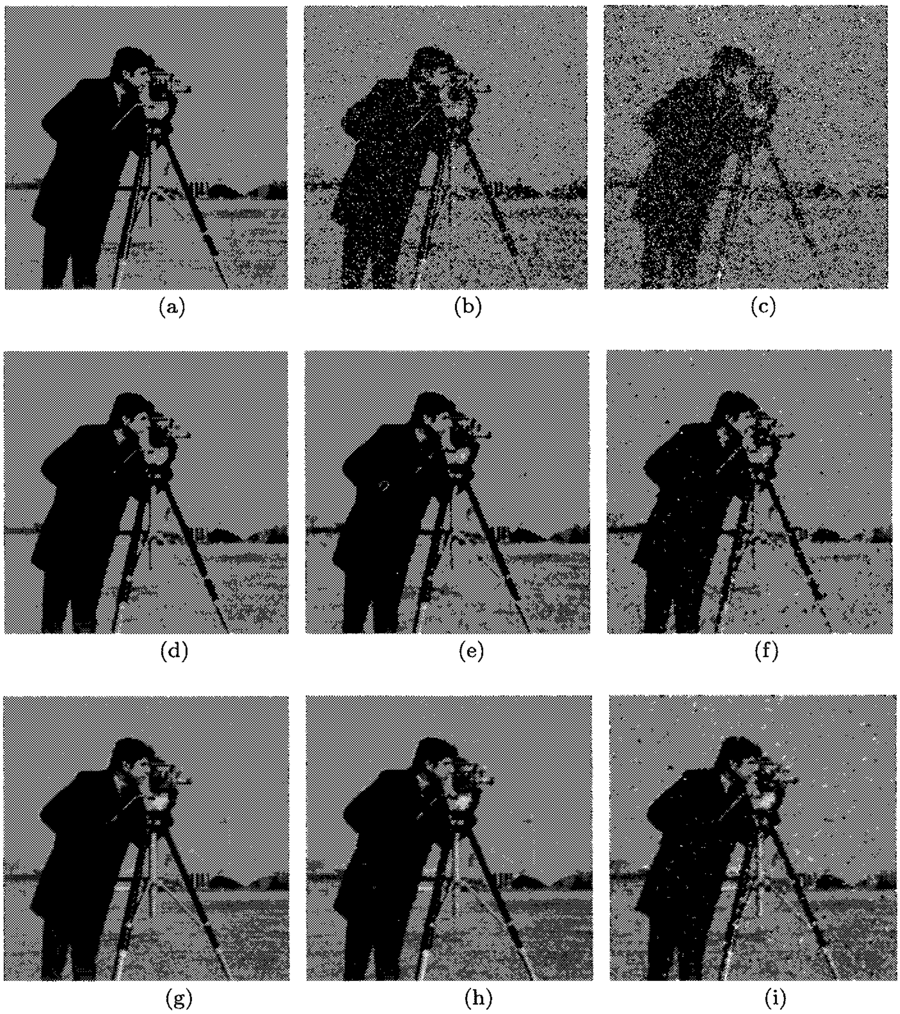


Figure 8.18 (a) Original image; (b) Noise image with $p = 0.15$; (c) Noise image with $p = 0.35$; (d) (e) (f) Filtering images of (a) (b) (c) by median filter; (g) (h) (i) Filtering images of (a) (b) (c) by FNN filter.

8.2.3 Experimental results

In order to evaluate the filtering capability of the FNN filter, we utilize respectively cameraman's and Lenna's images degraded by the impulse noise to show that the FNN filter is superior to the median or the first order RCRS filters in removing impulse noise. The sliding window in the simulations is assumed to be 3×3. Let $X = \big(s(t_1, t_2)\big)_{256\times 256}$ be the image 'cameraman' with 256 gray levels, which is corrupted by two kinds of impulse noises with

$p = 0.15$ and $p = 0.35$, respectively. The corresponding noise images are respectively shown in (b) (c) of Figure 8.18. Using Algorithm 8.2 we can get the iteration curve for finding the optimal S_0, which is shown Figure 8.16, from which we may choose $S_0 = 60$ for $p = 0.35$, and $S_0 = 70$ for $p = 0.15$. In Figure 8.16 we add some iterative steps by the following scheme after the threshold is determined as doing in Step 5 of Algorithm 8.2 for avoiding to fall into the local minimum point:

$$S[t+1] = S[t] - \beta \cdot |E(S[t]) - E(S[t-1])|.$$

Algorithm 8.2 is applied to the left upper part of the image 'cameraman' sized 64×64 for $p = 0.15$ and $p = 0.35$, respectively. Also Figure 8.16 demonstrates the convergence of the algorithm.

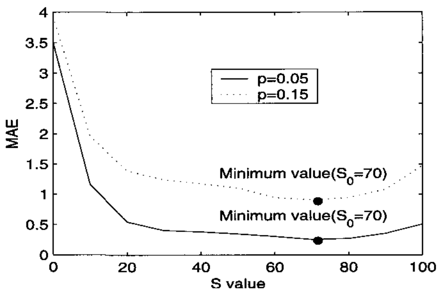


Figure 8.19

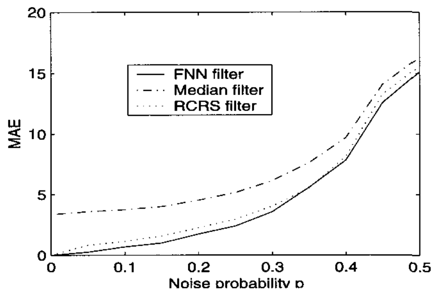


Figure 8.20

Figure 8.19: Iteration of S with image 'Lenna'; Figure 8.20: MAE of median and FNN filters for 'Lenna' with p

Figure 8.17 shows the MAE for median and FNN filters operating respectively on the image 'cameraman' corrupted by impulse noise with different probability p. Each error curve corresponds to the median and FNN filters trained by the sub-image 'cameraman' of size 64×64 locating at the left upper of whole 'cameraman' sized 256×256. Easily we can find that the FNN filter gives the better results. From this plot, the filtering effect of the FNN filter is obviously advantageous over that of median filter.

In Figure 8.18 we present several filtered images for subjective evaluation. The original image 'cameraman' is shown in Figure 8.18 (a), and the noise images with the probabilities $p = 0.15$ and $p = 0.35$ are shown in Figure 8.18 (b) (c), respectively. Figure 8.18 (d) (e) (f) show the restored images by median filter corresponding to Figure 8.18 (a) (b) (c), respectively. And correspondingly Figure 8.18 (g) (h) (i) are the images restored by the FNN filter. From Figure 8.18 the FNN filter appears to have removed impulses as median filter does, also it preserves more of the fine structures of the image than median filter. So the FNN filter has a better performance.

Next, the further comparison of the filtering capability is finished among the FNN filter, median and RCRS filters [19] for 256×256 image 'Lenna'.

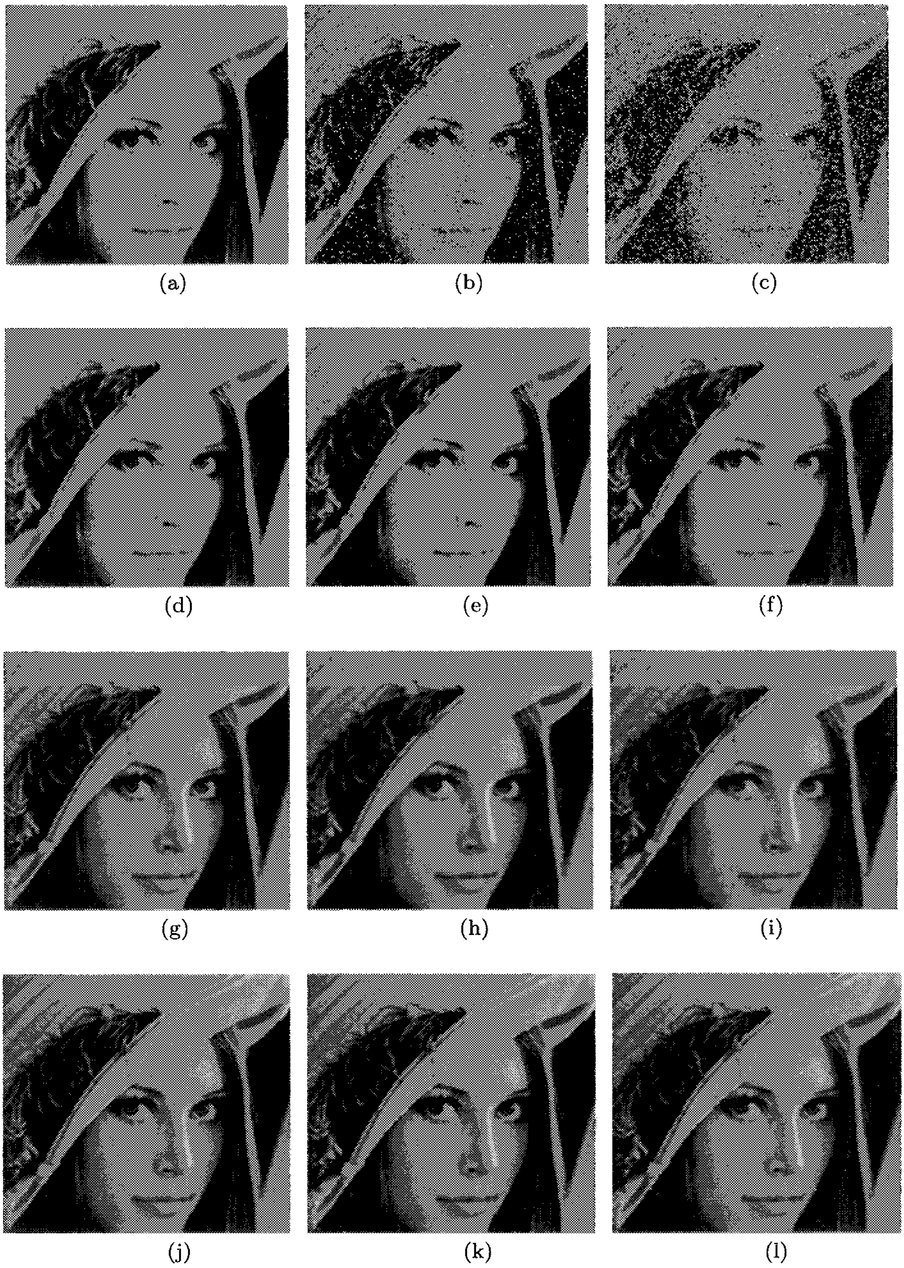


Figure 8.21 (a) Original image; (b) Noise image with $p = 0.05$; (c) Noise image with $p = 0.15$; (d) (e) (f) by median filter; (g) (h) (i) by RCRS filter; (j) (k) (l) by FNN filter, corresponding to (a) (b) (c) respectively.

At first, Figure 8.19 shows the iteration processes of Algorithm 8.2 for finding the optimal S_0, respectively for $p = 0.05$ and $p = 0.15$. By Figure 8.19, when $p = 0.05$, choose $S_0 = 70$; when $p = 0.15$, also we may choose $S_0 = 70$.

Figure 8.20 shows the MAE curves with different probabilities for median, RCRS and FR filters, respectively. Each error curve corresponds to the specified filter trained on the left upper of size 64×64 of 'Lenna' operating on whole 'Lenna' of size 256×256. Easily we can find that the FNN filter gives the best results.

In order to see the subjective evaluation for the performance of the FNN filter, we give noise-free 'Lenna' image as Figure 8.21 (a), and (b) (c) show the noise images with the noise probabilities $p = 0.05$, $p = 0.15$, respectively. Figure 8.21 (d) (e) (f), Figure 8.21 (g) (h) (i) and Figure 8.21 (j) (k) (l) show the images restored using median, RCRS and FNN filters corresponding to (a) (b) (c), respectively. Also we can see the filtering performance of the FNN filter is best.

Pixels and gray levels are two important factors of digital images. In this section we express an image as the I/O relationship of a FINN through this two factors. Thus, FINN's may provide us with a useful framework for image processing, especially for the restoration of noise images. From above discussions we also can see that the filtering performance of the filters based on FINN's is advantageous over that of many crisp nonlinear filters including median and RCRS filters and so on. However, the filters defined by FINN's can not solve all problems related. Many important and meaningful problems in the field are not treated by such filters. For instance, how can an image restoration model within general sense be constructed? When the image related is corrupted by the high probability ($p \geq 50\%$) impulse noise how is the filtering performance improved further? If the image is degraded by hybrid noise, i.e. several noises together corrupt the digital image, how can the corresponding restoration model be developed? and so on. We shall in the following give some further research to above problems.

§8.3 Image restoration based on FINN

By partitioning reasonably input space and gray level set of a digital image, respectively, a novel FNN that is called selection type FNN is developed. Such a system can represent continuous spatial images with arbitrary degree of accuracy. Also a novel inference type FNN is built based on a family of inference rules with real senses. Thus, a novel FNN filter can be derived by the fusion of the selection type FNN and the inference type FNN. Applying FNN filter, we can find a good compromise between removing impulse noise and preserving fine image structure. When noise probability is zero, the image may completely be reconstructed by the novel FNN filter. To the degraded images corrupted by impulse noise and additive Gaussian noise, simultaneously, we can also get good filtering performance by using the novel FNN filter.

8.3.1 Fuzzy partition

In order to use natural language to describe digital images and their gray levels, the interval $[a, b]$ is partitioned by a fuzzy partition. Let $\widetilde{B}_1, ..., \widetilde{B}_p$ be fuzzy numbers. If there is $c \in \mathbb{R}_+$, so that

$$\forall t \in [a, b], \sum_{j=1}^{p} \widetilde{B}_j(t) \leq c,$$

then $\{\widetilde{B}_1, ..., \widetilde{B}_p\}$ is called a quasi-fuzzy partition of $[a, b]$; If $\forall t \in [a, b]$, it follows that $\sum_{j=1}^{p} \widetilde{B}_j(t) = 1$, then $\{\widetilde{B}_1, ..., \widetilde{B}_p\}$ is called a fuzzy partition of $[a, b]$.

In application, a digital image is a set scaled within a finite area. So let $a > 0$, so that the digital images related are restricted in the spatial area $[-a, a]^d$, where $d \in \mathbb{N}$. Choose $m \in \mathbb{N}$, and partition interval $[-a, a]$ into $2m$ sub-intervals with identical length. Thus, we can define the fuzzy set family $\{\widetilde{A}_{ij} \,|i = 1, ..., d;\ j = 0, \pm1, \pm2, ..., \pm m\} \subset \widetilde{\mathcal{O}}_0(a, m)$, so that each $\widetilde{A}_{ij}$ is a fuzzy number, and $\forall t \notin [-a, a]$, $\widetilde{A}_{ij}(t) \equiv 0$. Obviously $\forall i = 1, ..., d$, the fuzzy set family $\{\widetilde{A}_{ij},\ j = 0, \pm1, ..., \pm m\}$ constitutes a quasi-fuzzy partition of $[-a, a]$.

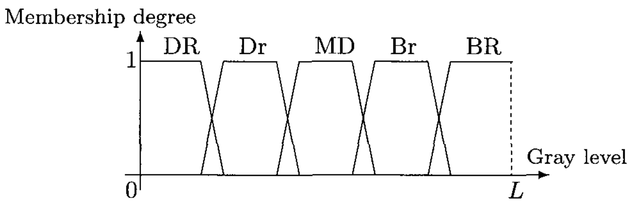


Figure 8.22 Gray level fuzzy sets

Suppose the gray levels of the 2-D image $F = \{F(t_1, t_2),\ -a \leq t_1, t_2 \leq a\}$ belong to $[0, L]$, that is, $\forall (t_1, t_2) \in [-a, a]^2$, $F(t_1, t_2) \in [0, L]$. We introduce a fuzzy partition of $[0, L]$, and use fuzzy sets to describe the image F. Thus, natural language such as 'dark (DK)' 'darker (Dr)' 'medium (MD)' 'brighter (Br)' 'bright (BR)' and so on may be employed to describe gray levels of the image. Their membership functions are shown in Figure 8.22. Let K be an adjustable natural number, $K < L$. Partition $[0, L]$ identically into K parts. Write $h = L/K$. Choosing $k_0 \in \mathbb{N} :\ k_0 \leq K$, we employ natural language determined by k_0 fuzzy sets to describe the digital image F. $\forall k = 1, ..., K$, write $I_k = [(k-1)L/K,\ kL/K] = [(k-1)h,\ kh]$. Define the fuzzy sets describing gray levels as follows: $\forall y \in [0, L]$, let

$$\widetilde{G}_1(y) = \begin{cases} 1, & 0 \leq y \leq h-1, \\ \dfrac{h+1-y}{2}, & h-1 < y \leq h+1; \\ 0, & \text{otherwise.} \end{cases} \tag{8.25}$$

$$\widetilde{G}_{k_0}(y) = \begin{cases} \dfrac{y - L + h + 1}{2}, & L - h - 1 \le y \le L - h + 1, \\ 1, & L - h + 1 < y \le L, \\ 0, & \text{otherwise}. \end{cases} \tag{8.26}$$

And when $k = 2, ..., k_0 - 1$, we suppose $\text{Ker}(\widetilde{G}_{k-1}) \subset I_{k_1}$, $\text{Ker}(\widetilde{G}_k) \subset I_{k_2}$, and $\text{Ker}(\widetilde{G}_{k+1}) \subset I_{k_3}$. Choose

$$\widetilde{G}_k(y) = \begin{cases} \dfrac{y - k_1 h + 1}{(k_2 - k_1 - 1)h + 2}, & k_1 h - 1 \le y \le (k_2 - 1)h + 1, \\ 1, & (k_2 - 1)h + 1 < t \le k_2 h - 1, \\ \dfrac{(k_3 - 1)h + 1 - y}{(k_3 - k_2 - 1)h + 2}, & k_2 h - 1 < y \le (k_3 - 1)h + 1, \\ 0, & \text{otherwise}. \end{cases} \tag{8.27}$$

Figure 8.22 is the case of $K = k_0 = 5$.

For given $t_1, t_2 \in [-a, a]$, and $k \in \{1, ..., k_0\}$, we define the fuzzy mean of the image F at (t_1, t_2) as follows:

$$m_{t_1 t_2}^k(F) = \frac{\sum\limits_{j_1, j_2 = -m}^{m} F\left(\frac{aj_1}{m}, \frac{aj_2}{m}\right) \cdot H_{j_1 j_2}(t_1, t_2) \cdot \widetilde{G}_k\Big(F\left(\frac{aj_1}{m}, \frac{aj_2}{m}\right)\Big)}{\sum\limits_{j_1, j_2 = -m}^{m} H_{j_1 j_2}(t_1, t_2) \cdot \widetilde{G}_k\Big(F\left(\frac{aj_1}{m}, \frac{aj_2}{m}\right)\Big)}. \tag{8.28}$$

$m_{t_1 t_2}^k(F)$ is called the k-th local fuzzy mean of image F at (t_1, t_2). If $F = \{F(t_1, t_2),\ t_1, t_2 \in [-a, a]\}$ is corrupted by impulse noise, for simplicity, we also denote the degraded image by F. Then by (8.23) with probability p the gray level of image F at (t_1, t_2) is changed into L or 0, that is

$$F(t_1, t_2) = \begin{cases} L, & \text{with probability } p/2, \\ F(t_1, t_2), & \text{with probability } 1 - p, \\ 0, & \text{with probability } p/2. \end{cases}$$

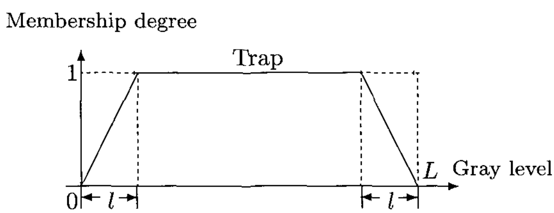


Figure 8.23 Mean fuzzy set

In order to suppress impulse noise, choose adjustable $l \in \mathbb{R}_+ : \ l < L/2$ to define fuzzy set 'Trap' as shown in Figure 8.23.

$$\forall y \in [0,\ L],\ \mathrm{Trap}(y) = \begin{cases} \dfrac{y}{l}, & 0 \le y \le l, \\ 1, & l < y \le L-l, \\ \dfrac{L-y}{l}, & L-l < y \le L, \\ 0, & \text{otherwise.} \end{cases}$$

Trap(·) is called the mean fuzzy set. For $(t_1,\ t_2) \in [-a,\ a]^2$, we call $m_{t_1t_2}(F)$ defined by following (8.29) the fuzzy mean of image F at $(t_1,\ t_2)$:

$$m_{t_1t_2}(F) = \frac{\sum\limits_{j_1,j_2=-m}^{m} F\big(\frac{aj_1}{m},\ \frac{aj_2}{m}\big)\cdot H_{j_1j_2}(t_1,\ t_2)\cdot \mathrm{Trap}\big(F\big(\frac{aj_1}{m},\ \frac{aj_2}{m}\big)\big)}{\sum\limits_{j_1,j_2=-m}^{m} TH_{j_1j_2}(t_1,\ t_2)\cdot \mathrm{Trap}\big(F\big(\frac{aj_1}{m},\ \frac{aj_2}{m}\big)\big)} \qquad (8.29)$$

Define the closed interval $\triangle_j = \big[(-a) \vee \big(a(j-1)/m\big),\ a \wedge \big(a(j+1)/m\big)\big]$ for $j = -m,\ -m+1, ..., m-1,\ m$, and let $\chi_{\triangle_j}$ be the characteristic function of $\triangle_j$. Let $\widetilde{A}_{1j} = \widetilde{A}_{2j} = \chi_{\triangle_j}$, then $c_0 = 3$. Discretize image F as the following digital image: $S = \{s_{j_1j_2},\ j_1,\ j_2 = 0, \pm1, ..., \pm m\}$, where $s_{j_1j_2} = F\big(aj_1/m,\ aj_2/m\big)$. Let $m^k_{t_1t_2}(F) \triangleq a^k(\underline{X})$, $m_{t_1t_2}(F) \triangleq m(\underline{X})$, Rewriting (8.28) (8.29) we obtain

$$\begin{cases} m^k(\underline{X}) = \dfrac{\sum\limits_{j_1=(j_1^0-1)\vee(-m)}^{(j_1^0+1)\wedge m} \sum\limits_{j_2=(j_2^0-1)\vee(-m)}^{(j_2^0+1)\wedge m} s_{j_1j_2}\, \widetilde{G}_k(s_{j_1j_2})}{\sum\limits_{j_1=(j_1^0-1)\vee(-m)}^{(j_1^0+1)\wedge m} \sum\limits_{j_2=(j_2^0-1)\vee(-m)}^{(j_2^0+1)\wedge m} \widetilde{G}_k(s_{j_1j_2})}; \\[2ex] m(\underline{X}) = \dfrac{\sum\limits_{j_1=(j_1^0-1)\vee(-m)}^{(j_1^0+1)\wedge m} \sum\limits_{j_2=(j_2^0-1)\vee(-m)}^{(j_2^0+1)\wedge m} s_{j_1j_2}\mathrm{Trap}(s_{j_1j_2})}{\sum\limits_{j_1=(j_1^0-1)\vee(-m)}^{(j_1^0+1)\wedge m} \sum\limits_{j_2=(j_2^0-1)\vee(-m)}^{(j_2^0+1)\wedge m} \mathrm{Trap}(s_{j_1j_2})}. \end{cases}$$

where $\underline{X} \triangleq \{s_{j_1j_2},\ j_1 = (j_1^0-1) \vee (-m),\ j_1^0,\ (j_1^0+1) \wedge m;\ j_2 = (j_2^0-1) \vee (-m),\ j_2^0,\ (j_2^0+1) \wedge m\}$ is the operating window corresponding to $(t_1,\ t_2) = \big(mj_1^0/m,\ aj_2^0/m\big)$. When $j_1,\ j_2$ change from $-m,\ -m+1, ...,$ to m, window $\underline{X}$ slides on the whole image S. The number of the elements in $\underline{X}$ is called the width of operating window. And $m^k(\underline{X})$ is called the k-th local fuzzy mean of window $\underline{X}$; and $m(\underline{X})$ is called the fuzzy mean of $\underline{X}$.

8.3.2 Selection type FNN and its universal approximation

If the function of each output neuron of feedforward networks is with some criterions to choose one from its inputs as the output, such networks are called selection type neural networks. In this subsection, we develop a selection type FNN, which is a five layer feedforward network, as shown Figure 8.24. The neurons in the first hidden layer have transfer function $\tilde{A}_{ij}(\cdot)$ ($i = 1, 2$; $j = -m, -m+1, ..., m-1, m$), the membership function of gray level fuzzy set. The connection weights of neurons between two hidden layers are chosen as $\tilde{G}_1(\cdot), ..., \tilde{G}_{k_0}(\cdot)$, the fuzzy set membership functions of fuzzy partition of the gray level set. By the fuzzy mean $m_{t_1t_2}(F)$ we can build a criterion for output neuron to choose output. Such a system can be universal approximator, that is, it can with arbitrary accuracy represent each continuous function on any compact set of Euclidean space. Also it can deal with impulse noise, efficiently. The learning algorithm of the network aims at seeking suitable partition of the gray level set.

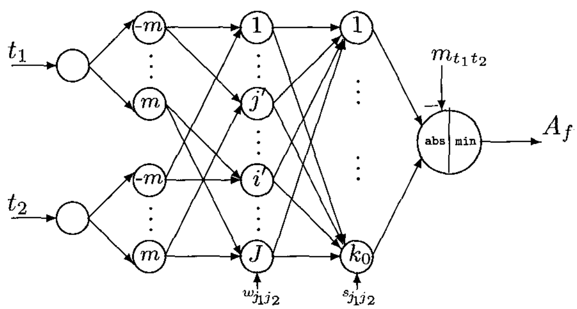


Figure 8.24 Selection type FNN

The FNN that can realize fuzzy inference rules is called inference type FNN. We shall employ inference type FNN as shown Figure 8.25, and selection type FNN to construct another FNN filter —a novel FNN filter.

Since the images related are two dimensional, in the following we aim mainly at the selection type FNN with two input neurons. As for one-input or d-input ($d \geq 3$) selection type FNN, we may give similar discussions. In Figure 8.24, $J = (2m+1)^2$. And in the first hidden layer, there are two types of neurons. The transfer functions of the first type neurons are $\tilde{A}_{1(-m)}(\cdot), ..., \tilde{A}_{1m}(\cdot)$, respectively; And ones of the second type are $\tilde{A}_{2(-m)}(\cdot), ..., \tilde{A}_{2m}(\cdot)$, respectively. In the second hidden layer, the neuron $j' = j_1j_2$ is connected with the j_1-th that is a first type neuron, and the j_2-th that is a second type neuron in the first hidden layer. Its two inputs are $\tilde{A}_{1j_1}(t_1)$, $\tilde{A}_{2j_2}(t_2)$. The corresponding output is $w_{j_1j_2} \cdot (\tilde{A}_{1j_1}(t_1)\, T\, \tilde{A}_{2j_2}(t_2)) = w_{j_1j_2} \cdot H_{j_1j_2}(t_1, t_2)$, where $w_{j_1j_2}$ is an adjustable parameter. In the third hidden layer, the neuron k is connected with the neuron $j' = j_1 j_2$ in the second hidden layer. Their connection weights is $\widetilde{G}_k(s_{j_1j_2})$, where $s_{j_1j_2}$ is adjustable. The output of the neuron k is

$$Y_k(t_1,\, t_2) = \frac{\sum\limits_{j_1, j_2=-m}^{m} s_{j_1j_2} \cdot \widetilde{G}_k(s_{j_1j_2}) \cdot H_{j_1j_2}(t_1,\, t_2)}{\sum\limits_{j_1, j_2=-m}^{m} \widetilde{G}_k(s_{j_1j_2}) \cdot H_{j_1j_2}(t_1,\, t_2)}. \tag{8.30}$$

Also $Y_k(t_1,\, t_2)$ is the k-th input of the output neuron, by which a selective output is obtained. The selection criterion of the output neuron is determined by $m_{t_1t_2}$. And the selection standard is the nearest distance, that is, choosing one from input patterns $Y_1(t_1,\, t_2), ..., Y_{k_0}(t_1,\, t_2)$, so that the chosen input being nearest to $m_{t_1t_2}$ is taken as the output of the FNN:

$$\begin{aligned} &A_f = Y_{k'}(t_1,\, t_2) : \\ &\quad k' = \max_{k_1}\big\{k_1 : |m_{t_1t_2} - Y_{k_1}(t_1,\, t_2)| = \min_k\big\{|m_{t_1t_2} - Y_k(t_1,\, t_2)|\big\}\big\}. \end{aligned} \tag{8.31}$$

(8.30) (8.31) constitute the I/O relationship of the selection type FNN. Similarly, we can derive the I/O relationship of one-dimensional and $d-$ dimensional I/O relationships respectively as follows:

$$\begin{cases} Y_k(t) = \dfrac{\sum\limits_{j=-m}^{m} s_j\, \widetilde{G}_k\,(s_j)\cdot \widetilde{A}_{1j}\,(t)}{\sum\limits_{j=-m}^{m} \widetilde{G}_k\,(s_j)\cdot \widetilde{A}_{1j}\,(t)}, \\ A_f = Y_{k'}(t) : k' = \max\limits_{k_1}\big\{k_1 : |m_t - Y_{k_1}(t)| = \min\limits_{k}\{|m_t - Y_k(t)|\}\big\}. \end{cases}$$

$$\begin{cases} Y_k(t_1, ..., t_d) = \dfrac{\sum\limits_{j_1,...,j_d=-m}^{m} s_{j_1...j_d}\, \widetilde{G}_k\,(s_{j_1...j_d}) \cdot H_{j_1...j_d}(t_1, ..., t_d)}{\sum\limits_{j_1,...,j_d=-m}^{m} \widetilde{G}_k\,(s_{j_1...j_d}) \cdot H_{j_1...j_d}(t_1, ..., t_d)} \\ A_f = Y_{k'}(t_1, ..., t_d) : \\ \quad k' = \max\limits_{k_1}\big\{k_1 :\ |m_{t_1...t_d} - Y_{k_1}(t_1, ..., t_d)| = \min\limits_{k}\{|m_{t_1...t_d} - Y_k(t_1, ..., t_d)|\}\big\}. \end{cases} \tag{8.32}$$

In the following, we show that the selection type FNN defined by (8.32) is a universal approximator. That is, if $F:\ \mathbb{R}^d \longrightarrow \mathbb{R}$ is a continuous function, and $U \subset \mathbb{R}^d$ is an arbitrary compact set, then $\forall \varepsilon > 0$, there exist $m \in \mathbb{N}$, $w_{j_1...j_d} \in \mathbb{R}$, $s_{j_1...j_d} \in \mathbb{R}$, and a mapping that $(t_1, ..., t_d) \longrightarrow m_{t_1...t_d}$, so that $\forall (t_1, ..., t_d) \in U$, $|A_f - F(t_1, ..., t_d)| = |Y_{k'}(t_1, ..., t_d) - F(t_1, ..., t_d)| < \varepsilon$, where

$$k' = \max_{k_1}\big\{k_1 :\ |m_{t_1...t_d} - Y_{k_1}(t_1, ..., t_d)| = \min_{k}\{|m_{t_1...t_d} - Y_k(t_1, ..., t_d)|\}\big\}.$$

Theorem 8.3 *The selection type FNN is a universal approximator, i.e. it can approximate each continuous function defined on arbitrary compact set of* $\mathbb{R}^d$ *with arbitrary degree of accuracy.*

Proof. Let $F: \mathbb{R}^d \longrightarrow \mathbb{R}$ be a continuous function. It is no harm to assume that compact set $U = [-a, a]^d$ $(a > 0)$. Arbitrarily given $\varepsilon > 0$, partition $[-a, a]$ identically into $2m$ parts. Since F is uniformly continuous on U, there is $\delta > 0$, so that

$$\forall \mathbf{x}_1, \mathbf{x}_2 \in U, \ \|\mathbf{x}_1 - \mathbf{x}_2\| < \delta, \Longrightarrow |F(\mathbf{x}_1) - F(\mathbf{x}_2)| < \varepsilon. \tag{8.33}$$

where $\|\cdot\|$ is Euclidean norm. Let $m \in \mathbb{N}$, satisfying $ac_0/m < \delta$. For $j_1, ..., j_d \in \{0, \pm 1, ..., \pm m\}$, choose $w_{j_1...j_d} = s_{j_1...j_d} = F\big(aj_1/m, ..., aj_d/m\big)$. Similarly with (8.29) we define $m_{t_1...t_d}$ as follows:

$$m_{t_1...t_d} = \frac{\sum\limits_{j_1,...,j_d=-m}^{m} F\big(\frac{aj_1}{m}, ..., \frac{aj_d}{m}\big) \cdot H_{j_1...j_2}(t_1, ..., t_d) \cdot \mathrm{Trap}\big(F\big(\frac{aj_1}{m}, ..., \frac{aj_d}{m}\big)\big)}{\sum\limits_{j_1,...,j_2=-m}^{m} H_{j_1...j_d}(t_1, ..., t_d) \cdot \mathrm{Trap}\big(F\big(\frac{aj_1}{m}, ..., \frac{aj_d}{m}\big)\big)}.$$

Let the natural number K related to (8.25) (8.26) be adjusted so that for any $(t_1, ..., t_d) \in [-a, a]^d$, there is $k \in \{1, ..., k_0\}$, satisfying

$$\forall (j_1, ..., j_d) \in N(t_1, ..., t_d), \ \widetilde{G}_k(s_{j_1...j_d}) \neq 0, \ \ A_f = Y_k(t_1, ..., t_d).$$

By (8.33) and Lemma 6.2, it follows that

$$\begin{aligned}
&|F(t_1, ..., t_d) - A_f| = |F(t_1, ..., t_d) - Y_k(t_1, ..., t_d)| \\
&= \left| \frac{\sum\limits_{j_1,...,j_d=-m}^{m} s_{j_1...j_d} \widetilde{G}_k(s_{j_1...j_d}) H_{j_1...j_d}(t_1, ..., t_d)}{\sum\limits_{j_1,...,j_d=-m}^{m} \widetilde{G}_k(s_{j_1...j_d}) H_{j_1...j_d}(t_1, ..., t_d)} - F(t_1, ..., t_d) \right| \\
&\leq \frac{\sum\limits_{(j_1,...,j_d) \in N(t_1,...,t_d)} \widetilde{G}_k(s_{j_1...j_d}) H_{j_1...j_d}(t_1, ..., t_d) \left| F(t_1, ..., t_d) - F\big(\frac{aj_1}{m}, ..., \frac{aj_d}{m}\big) \right|}{\sum\limits_{(j_1,...,j_d) \in N(t_1,...,t_d)} \widetilde{G}_k(s_{j_1...j_d}) H_{j_1...j_d}(t_1, ..., t_d)} \\
&< \frac{\sum\limits_{(j_1,...,j_d) \in N(t_1,...,t_d)} \widetilde{G}_k(s_{j_1...j_d}) H_{j_1...j_d}(t_1, ..., t_d) \cdot \varepsilon}{\sum\limits_{(j_1,...,j_d) \in N(t_1,...,t_d)} \widetilde{G}_k(s_{j_1...j_d}) H_{j_1...j_d}(t_1, ..., t_d)} \\
&= \varepsilon.
\end{aligned}$$

The theorem is therefore proved. □

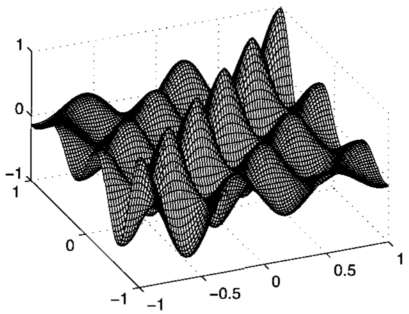


Figure 8.25 Original surface

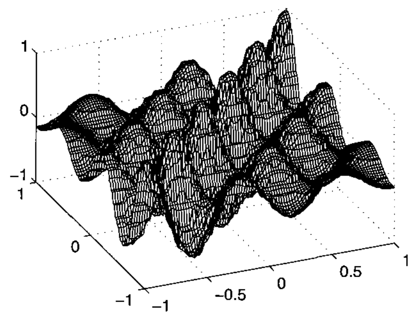


Figure 8.26 Approximate surface

Give a two-variate function as $F(t_1,\, t_2) = \exp\{-|t_1 - t_2|\} \cdot \sin(8t_1) \cdot \sin(8t_2)$ for $t_1,\, t_2 \in [-1,\, 1]$. With the error bound $\varepsilon = 0.1$, we shall give the approximate representation of F by the selection type FNN. To this end, assume that $c_0 = 2,\ m = 1$, and t$-$norm $T =$ '$\times$'. Easily we have

$$\forall \mathbf{x}_1,\, \mathbf{x}_2 \in [-1,\, 1]^2,\ \|\mathbf{x}_1 - \mathbf{x}_2\| < 0.01, \Longrightarrow |F(\mathbf{x}_1) - F(\mathbf{x}_2)| < \varepsilon.$$

Let $c_0/m \le 0.01$. We can choose $m = 200$. And assume that $\widetilde{A}_{1j} = \widetilde{A}_{2j}$ $(j = -200, ..., 200)$, and

$$\forall t \in [-1, 1],\ \widetilde{A}(t) = \begin{cases} 200t + 1, & -\dfrac{1}{200} \le t \le 0, \\ 1 - 200t, & 0 < t \le \dfrac{1}{200}, \\ 0, & \text{otherwise}; \end{cases} \qquad \widetilde{A}_{1j}(t) = \widetilde{A}\Big(t - \frac{j}{200}\Big).$$

Then we can conclude that

$$A_f = Y_{k'}(t_1,\, t_2) = \frac{\sum\limits_{j_1, j_2 = -200}^{200} s_{j_1 j_2}\, \widetilde{G}_{k'}(s_{j_1 j_2}) \cdot \widetilde{A}_{1j_1}(t_1) \cdot \widetilde{A}_{1j_2}(t_2)}{\sum\limits_{j_1, j_2 = -200}^{200} \widetilde{G}_{k'}(s_{j_1 j_2}) \cdot \widetilde{A}_{1j_1}(t_1) \cdot \widetilde{A}_{1j_2}(t_2)},$$

$$k' = \max_{k_1}\big\{k_1 : \big|m_{t_1 t_2} - Y_{k_1}(t_1, t_2)\big| = \min_{k}\big\{\big|m_{t_1 t_2} - Y_k(t_1,\, t_2)\big|\big\}\big\}.$$

$$m_{t_1 t_2} = \frac{\sum\limits_{j_1, j_2 = -200}^{200} s_{j_1 j_2} \mathrm{Trap}(s_{j_1 j_2}) \cdot \widetilde{A}_{1j_1}(t_1) \cdot \widetilde{A}_{1j_2}(t_2)}{\sum\limits_{j_1, j_2 = -200}^{200} \mathrm{Trap}(s_{j_1 j_2}) \cdot \widetilde{A}_{1j_1}(t_1) \cdot \widetilde{A}_{1j_2}(t_2)},$$

$$s_{j_1 j_2} = F\Big(\frac{j_1}{m},\, \frac{j_2}{m}\Big).$$

As shown in Figure 8.25 is the surface of function $z = F(t_1, t_2)$. And we obtain the approximate surface $z = A_f$ by the selection type FNN, as shown in Figure 8.26. By the comparison between Figure 8.25 and Figure 8.26, it shows the high approximating accuracy of the selection type FNN.

8.3.3 A novel FNN filter

Suppose $X = (s_{ij})_{N_1 \times N_2}$ is a given digital image. Also X denotes its noise version. For given $p \in \mathbb{N}$, let the operating window be written as $\underline{X} = (x_{-p}, ..., x_0, ..., x_p)$. In this subsection we employ the operating window $\underline{X}$ to design another noise filter—a novel FNN filter. By Theorem 8.3 it can simulate some digital signals with arbitrary degree of accuracy. However, the filters remove fine image structure also. It is necessary to introduce the inference type FNN, as shown in Figure 8.27.

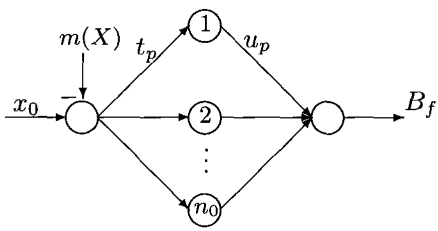


Figure 8.27 Inference type FNN

If X is a noise-free image, the gray degree levels $x_{-p}, ..., x_0, ..., x_p$ in the window $\underline{X}$ are approximately equal [3], that is, $x_{-p} \approx \cdots \approx x_0 \approx \cdots \approx x_p \approx m(\underline{X})$. So if u is an output variable, and $v = |x_0 - m(\underline{X})|$ is an input variable, then we can obtain the following Mamdani inference rules:

$$\text{IF } v \text{ is 'small' THEN } u \text{ is } \underset{\sim}{x_0};$$
$$\text{IF } v \text{ is 'large' THEN } u \text{ is } \underset{\sim}{A_f},$$

where $\underset{\sim}{x_0}$, $\underset{\sim}{A_f}$ mean the fuzzifications of x_0, A_f, respectively, satisfying $x_0 \in \text{Ker}(\underset{\sim}{x_0})$, $A_f \in \text{Ker}(\underset{\sim}{A_f})$. On the gray level set $[0, L]$, we define $\underset{\sim}{S}$= 'small', $\underset{\sim}{L}$='large', $\underset{\sim}{S}$ and $\underset{\sim}{L}$ are called selection type fuzzy sets, as shown Figure 8.28, which is similar with ones of deviation fuzzy sets in Figure 8.13.

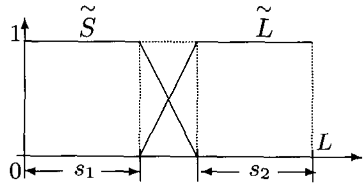


Figure 8.28 Selection type fuzzy sets

Above inference relation may be realized by a three-layer inference type FNN, as shown Figure 8.27, where $m_0 = 2$, t_p means the absolute value function. Thus the input of the neurons in hidden layer is $|x_0 - m(X)|$, and the transfer functions are determined by $\widetilde{S}(\cdot)$ and $\widetilde{L}(\cdot)$, respectively. By [13], The I/O relationship of the inference type FNN is as follows:

$$\begin{aligned} B_f &= \frac{x_0 \cdot \widetilde{S}(v) + A_f \cdot \widetilde{L}(v)}{\widetilde{S}(v) + \widetilde{L}(v)} \\ &= \frac{x_0 \cdot \widetilde{S}(|x_0 - m(X)|) + A_f \cdot \widetilde{L}(|x_0 - m(X)|)}{\widetilde{S}(|x_0 - m(X)|) + \widetilde{L}(|x_0 - m(X)|)}. \end{aligned} \tag{8.34}$$

And using (8.30) we can conclude that

$$\begin{cases} m^k(X) = \dfrac{\sum\limits_{j=-p}^{p} x_j \cdot \widetilde{G}_k(x_j)}{\sum\limits_{j=-p}^{p} \widetilde{G}_k(x_j)}, \quad m(X) = \dfrac{\sum\limits_{j=-p}^{p} x_j \cdot \mathrm{Trap}(x_j)}{\sum\limits_{j=-p}^{p} \mathrm{Trap}(x_j)}; \\ A_f = m^{k'}(X), \text{ where} \\ \quad k' = \max\limits_{k_1}\{k_1 : |m(X) - m^{k_1}(X)| = \min\limits_{k}\{|m(X) - m^k(X)|\}\}. \end{cases} \tag{8.35}$$

(8.34) (8.35) constitute the I/O relationship of the novel FNN filter, whose architecture is shown in Figure 8.29.

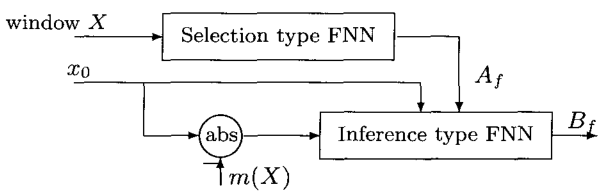


Figure 8.29 A novel FNN filter

8.3.4 Learning algorithm

This subsection aims at seeking an optimal FNN filter, which is constructed by designing learning algorithms for the parameters of the selection type FNN and the inference type FNN, respectively. The target function is MAE of the output of the filter, i.e. we obtain the optimal FNN filter by minimizing MAE.

The learning algorithm for the selection type FNN aims at determining the partition of the gray level set $[0, L]$, that is, establishing the value of k_0, and the trapezoidal fuzzy numbers in fuzzy partition. The algorithm can be described as follows:

(i) To a $m-$bit digital image, the variety not exceeding 2^{m-4} in gray level of image will not lead to obvious visual changes [25]. So we may partition [0, L] identically into 2^{m-4} parts.

(ii) Seek the concentration area of gray levels of the image X : Calculate the number Γ_k of $\{s_{ij},\ i=1,...,N_1,\ j=1,...,N_2\}$ belonging to the interval $I_k \triangleq \big[(k-1)L/(2^{m-4}),\ kL/(2^{m-4})\big)$ $(k=1,...,2^{m-4})$. And discriminate $\Gamma_k \geq \eta$? If yes, I_k is called concentration area of gray levels of S, where η is a given constant corresponding to the image X.

(iii) Determine the fuzzy partition of [0, L]: Let k_0 be the number of concentration areas of gray levels. These concentration areas of gray levels are $I_{i_1},...,I_{i_{k_0}}$. By (8.25)–(8.27) we define the trapezoidal fuzzy numbers.

In the definition of mean fuzzy set for the selection type FNN, the parameter l is assumed to be about 3, and the selection standard value is $m(X)$, we can get good results for removing impulse noise.

Based on above learning algorithm for the selection type FNN, we discuss the learning procedure of the inference type FNN, by which s_1, s_2 are determined to minimize MAE. If the noises in image is mainly impulse noise, we can assume that $s_1+s_2 \approx L$, for example, let $s_1+s_2=L-1$. So we can find out optimal value of s_1 to minimize MAE. The algorithm is described as follows:

(i) Put the initial value of s_1 as $s_1[0]$, and let the iteration step be $t=0$;

(ii) Calculate absolute error $err[t] = \Big(\sum\limits_{i=1}^{N_1}\sum\limits_{j=1}^{N_2}\big|B_f^q(i,j)-U(i,j)\big|\Big)\Big/(N_1\times N_2)$, where $U(i,j)$ is the desired output of the novel FNN filter at pixel (i,j), and $B_f^q(i,j)$ is the real output of the novel FNN filter at pixel (i,j) when the iteration step is t.

(iii) Iterate s_1 with the following scheme:

$$s_1[t+1] = s_1[t] + \alpha\cdot\frac{\triangle err[t]}{t+1},$$

where α is the learning constant, $\triangle err[t] = err[t] - err[t-1]$, and $err[-1] = err[0]/2$.

(iv) Discriminate $\big|s_1[t+1]-s_1[t]\big| < 0.1$? If yes, output the value $s_1[t+1]$; otherwise let $t=t+1$, and go to step (ii).

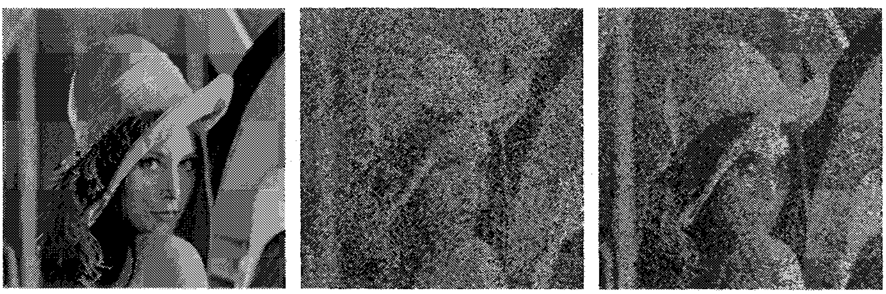
8.3.5 Simulation examples

Assume that image $X=\{s_{ij}\}$ with 512×512 pixels is a 8−bit Lenna image. If X is noise-free, it is shown as Figure 8.30 (a). In this subsection we employ Lenna image to examine the capability of the novel FNN filter to remove noises. To the degraded images corrupted by impulse noise, the novel FNN filter can give much better performance than AWFM filter in [25]. The capability of removing noise of the novel FNN filter is superior to one of conventional filters, such as median filter, RCRS filter, and so on. This is because by [25], AWFM filter can give better performance than the RS type filters can. So we choose median and AWFM filters to demonstrate the advantageous performance of the novel FNN filter to remove noises.

At first for simplicity, we sample uniformly from X to form sub-image $X' = (s'_{ij})_{64\times 64}$. Choose an operating window $\underline{X}$ with 3×3. For given impulse noise probability as $p = 0$, 0.1, 0.3, 0.5, 0.6, 0.8, 1.0, we train the selection type FNN and inference type FNN by the sub-image X' to obtain the optimal filter, respectively. Table 8.1 gives the respective minimum MAE's of median, AWFM and FNN filters corresponding to the given noise probabilities, respectively. From Table 8.1, the novel FNN filter gives the best filtering results, moreover, fine image structure is well preserved.

Table 8.1 MAE's of three filters under different noise occurrence probabilities

	$p = 0$	$p = 0.1$	$p = 0.3$	$p = 0.5$	$p = 0.6$	$p = 0.8$	$p = 1.0$
median filter	8.904	10.038	13.528	26.399	40.8604	77.091	130.0
AWFM filter	2.519	9.333	13.093	15.854	19.804	30.688	124.453
novel FNN filter	0.0	3.274	6.903	9.255	14.687	29.831	124.453



(a) (b) (c)

Figure 8.30 Lenna image (a) noise-free, (b) corrupted by impulse noise ($p = 0.6$), (c) corrupted by impulse ($p = 0.4$) and Gaussian noises

With noise probability $p = 0.6$, we employ X' to train the FNN's related. That is, under the criterion of minimum MAE we calculate optimal parameter s_1 in the inference fuzzy sets $\tilde{S}$, $\tilde{L}$ shown in Figure 8.28, and the optimal FNN filter can be constructed. To this end, let the learning constant $\alpha = 0.1$. With 100 iteration steps, s_1 converges approximately to 15, and the approximate MAE of the novel FNN filter is 14.6867.

(a) (b) (c)

Figure 8.31 Restoration of impulse noise image (a) median filter, (b) AWFM filter, (c) novel FNN filter

(b) of Figure 8.30 is Lenna noise image X degraded by impulse noise ($p = 0.6$). And Figure 8.31 is the restorations of impulse noise image in Figure 8.30 (b), where (a) is the image restored by median filter; (b) (c) are the images restored by AWFM filter and the novel FNN filter, respectively. From the comparison among (a) (b) and (c) in Figure 8.31, we can conclude that the novel FNN filter gives the best results though the performance of median filter is improved by AWFM filter. the novel FNN filter appears to be best both in removing impulse noise and preserving image structure.

(a) (b) (c)

Figure 8.32 Restoration of hybrid noise image (a) median filter, (b) AWFM filter, (c) novel FNN filter

Hybrid noise images mean ones that are corrupted by two or more kinds of noises. Applying the novel FNN filter to these noise images we also obtain acceptable restoring images. Assume that there are in image X impulse noise ($p = 0.4$) and Gaussian additive noise whose mean value $\mu = 0$, variance $\sigma^2 = 0.02$. X is shown in Figure 8.30 (c). We denote the minimum MAE of median filter, AWFM filter and the novel FNN filter by MAE_m, MAE_a and MAE_f, respectively. It follows that

$$\mathrm{MAE}_m = 29.466, \quad \mathrm{MAE}_a = 30.053, \quad \mathrm{MAE}_f = 26.725.$$

The restoring images by such three filters is shown in Figure 8.32 (a) (b) (c), respectively. From Figure 8.32, it shows easily that the novel FNN filter can result in highest quality restoration, compared with median and AWFM filters.

The suitable partitions of the input and output spaces are the basis to construct selection type FNN's. Such networks possess strong capability for locally representing some I/O systems. They also leads to good anti-disturbance in image processing, that is, if the image is corrupted by impulse noise with high noise occurrence probability ($p > 0.5$), we can employ a selection type FNN to give restoring image with good performance. One presupposition to design an inference type FNN is to preserve the fine image structure. So as the fusion of a section type FNN and an inference FNN, the novel FNN filter can not only remove noises in the image, but also preserve fine image structure. In future research, selection type FNN's and inference type FNN's can widely be applied in many real fields, such as system modelling, system simulation and system identification, and so on.

References

[1] Abreu E., Livinstone M. & Mitra S. K. et al, A new efficient approach for the removal of impulse noise from highly corrupted images, *IEEE Trans. on Image Processing,* 1996, **5**: 1012–1025.

[2] Anna E., Eugene C. E. & Carlos A. Reyes-Garcia, Speech noise cancellation based on a neuro-fuzzy system: further improvements, *J. Advanced Computational Intelligence,* 2001, **5**(1): 15–21.

[3] Arakawa K., Median filter based on fuzzy rules and its application to image processing, *Fuzzy Sets and Systems,* 1996, **77**: 3–13.

[4] Astola J. & Kuosmanen P., *Fundamentals of Nonlinear Digital Filtering,* Boca Raton FL: CRC, 1997.

[5] Castleman K. R., *Digital image processing,* Beijing: Tsinghua University Press, Prentice Hall, Inc., 1998.

[6] Chen R. C. & Yu P. T., Fuzzy selection filters for image restoration with neural learning, *IEEE Trans. on Signal Processing,* 1999, **47**: 1446–1450.

[7] Chen R. C. & Yu P. T., An optimal design of fuzzy (m, n) rank order filtering with hard decision neural learning, *J. Inform. Sci. Eng.,* 1995, **11**: 567–593.

[8] Cheng C. B., Cheng C. J. & Lee E. S., Neuro-fuzzy and genetic algorithm in multiple response optimization, *Computers & Math. with Appl.,* 2002, **44**(12): 1503–1514.

[9] Cheng H. D. et al, A novel fuzzy entropy approach to image enhancement and thresholding, *Signal Processing,* 1999, **75**(3): 277–301.

[10] Choi Y. & Krishnapuram R., A robust approach to image enhancement based on fuzzy logic, *IEEE Trans. on Image Processing,* 1997, **6**(6): 808–825.

[11] Civanlar M. R. & Trussell H. J., Digital image restoration using fuzzy sets, *IEEE Trans. on Acoust., Speech, Signal Processing,* 1986, **34**: 919–936.

[12] Deng H., Lau M. & Millar K., Defuzzification in fuzzy multicriteria analysis, *Proc. 19th Internat. Conf. of the North American Fuzzy Information Processing Soc.,* 2000: 222–226.

[13] Doroodchi M. & Reza A. M., A new reasoning based filter, *Proc. IEEE Internat. Conf. on Systems, Man and Cybertnet.,* 2000: 1656–1660.

[14] Eng H. L. & Ma K. K., Noise adaptive soft-switching median filter, *IEEE Trans. on Image Processing,* 2001, **10**(2): 242–251.

[15] Esogbue A. O., Song Q. & Hearnes II W. E., Defuzzification filters and applications to power system stabilization problems, *J. Math. Anal. Appl.,* 2000, **251**: 406–432.

[16] Farbiz F.& Menhaj M. B., et al, A new fuzzy logic filter for image enhancement, *IEEE Trans. on Systems, Man and Cybernet.-Part B,* 2000, **30**: 110–119.

[17] Filev D. P. & Yager R. R., A generalized defuzzification method via bad distributions, *Internat. J. of Intelligent Systems,* 1991, **6**: 687–697.

[18] Grara M., Anjou A. D. & Gonzalez A. I., et al, Competitive stochastic neural networks for vector quantization of images, *Neurocomputing,* 1995, **7**(2): 187–195.

[19] Hardie R. C. & Barner K. E., Rank conditional rank selection filters for signal restoration, *IEEE Trans.on Image Processing,* 1994, **3**: 192–206.

[20] Horikawa S., Furuhashi T. & Uchikawa Y., On fuzzy modeling using fuzzy neural networks with the back-propagation algorithm, *IEEE Trans.on Neural Networks,* 1992, **3**: 801–806.

[21] Ingman D. & Merlis Y., Maximum entropy signal reconstruction with neural networks, *IEEE Trans. on Neural Networks,* 1992, **3**: 195–201.

[22] Jang J. S. R., ANFIS: Adaptive-network-based fuzzy inference system, *IEEE Trans. on Systems, Man and Cybernet.,* 1993, **23**: 665–684.

[23] Jang J. S. R., Sun C. T. & Mizutani E., *Neuro-fuzzy and soft computing,* Prentice Hall, 1997.

[24] Kim J. S. & Cho H. S., A fuzzy logic and neural network approach to boundary detection for noise imagery, *Fuzzy Sets and Systems,* 1994, **65**: 141–159.

[25] Kuo Y. H., Lee C. S. & Chen C. L., High-stability AWFM filter for signal restoration and its hardware design, *Fuzzy Sets and Systems,* 2000, **114**: 185–202.

[26] Lee C. S., Kuo Y. H. & Yu P. T., Weighted fuzzy mean filters for image processing, *Fuzzy Sets and Systems,* 1997, **89**: 157–180.

[27] Lee K. -M., Kwak D. -H. & Wang H. -L., Tuning of fuzzy models by fuzzy neural networks, *Fuzzy Sets and Systems,* 1995, **76**: 47–61.

[28] Lee K. Y., Complex fuzzy adaptive filter with LMS algorithm, *IEEE Trans. on Signal Processing,* 1996, **44**: 424–429.

[29] Lee Y. -G. & Hsueh Y. -C., Fuzzy logic approach for removing noise, *Neurocomputing,* 1995, **9**: 349–355.

[30] Li Hongxing, Output–back fuzzy logic systems and equivalence with feedback neural networks, *Chinese Science Bulletin,* 2000, **45**: 592–596.

[31] Li Hongxing & Chen C. L. P., The equivalence between fuzzy logic systems and feedforward neural networks, *IEEE Trans. on Neural Networks,* 2000, **11**: 356–365.

[32] Li S., Wang Y. & Zhang C., et al, Fuzzy filter based on neural network and its application to image restoration, *Proc. 5th Internat. Conf. on Signal Processing,* 2000: 1133–1138.

[33] Lin C. T., Neural-network-based fuzzy logic control and decision system, *IEEE Trans. on Comp.,* 1991, **40**: 1320–1336.

[34] Lin C. T. & Martin L.K., A neural fuzzy system for image motion estimation, *Fuzzy Sets and Systems,* 2000, **114**(2): 281–304.

[35] Lin C. T., Duh F. B. & Liu D. J., A neural fuzzy network for word information processing, *Fuzzy Sets and systems,* 2002, **127**(1): 37–48.

[36] Lin C. T. & Lu Y. C., A neural fuzzy system with linguistic teaching signals, *IEEE Trans. on Fuzzy Systems,* 1995, **3**(2): 169–189.

[37] Liu Puyin, Representation of digital image by fuzzy neural networks, *Fuzzy Sets and Systems,* 2002, **130**(1): 109–123.

[38] Liu Puyin & Li Hongxing, Image restoration techniques based on fuzzy neural networks, *Science in China, Series F,* 2002, **45**(4): 273–285.

[39] Liu Puyin & Li Hongxing, Fuzzy Techniques in image restoration— a survey, *Internat. Journal of Comp. Cognition,* 2004, **2**(2): 131–149.

[40] Mancuso M., De Luca R. & Poluzzi R. et al, A fuzzy decision directed filter for impulsive noise reduction, *Fuzzy Sets and Systems,* 1996, **77**: 111–116.

[41] Mendil B. & Benmahammed K., Activation and defuzzification methods for fuzzy rule-based systems, *J. of Intelligent & Robotic Systems,* 2001, **32**(4): 437–444.

[42] Meneganti M., Saviello F. S. & Tagliaferri R., Fuzzy neural networks for classification and detection of anomalies, *IEEE Trans. on Neural Networks,* 1998, **9**: 848–861.

[43] Mitaim S. & Kosko B., The shape of fuzzy sets in adaptive function approximation, *IEEE Trans. on Fuzzy Systems,* 2001, *9*(4): 637–656.

[44] Mitchell H. B. & Schaefer P. A., A fuzzy noise-smoothing filter, *Int. J. Intell. Systems,* 2000, *15*: 303–315.

[45] Nauck D. & Kruse R., A neuro-fuzzy method to learn fuzzy classification rules from data, *Fuzzy Sets and Systems,* 1997, **89**: 277–288.

[46] Nauck D. & Kruse R., Neuro-fuzzy systems for function approximation, *Fuzzy Sets and Systems,* 1999, **101**: 261–271.

[47] Nauck D. & Kruse R., A fuzzy neural network learning fuzzy control rules and membership functions by fuzzy error backpropagation, *Proc. IEEE Internat. Conf. on Neural Networks,* 1993: 1022–1027.

[48] Narendra K.S. & Parthasarathy K., Identification and control of dynamical systems using neural networks, *IEEE Trans. on Neural Networks,* 1990, **1**: 4–27.

[49] Nie J., Nonlinear time-series forecasting: A fuzzy-neural approach, *Neurocomputing,* 1997, **16**: 63–76.

[50] Nishina T. & Hagiwara M., Fuzzy inference neural network, *Neurocomputing,* 1997, **14**: 223–239.

[51] Oliveira J. V., A set-theoretical defuzzification method, *Fuzzy Sets and Systems,* 1995, **76**: 63–71.

[52] Pai N. S. & Li T. H., State estimation via Lyapunov-type fuzzy filter, *Elect. Lett.,* 1999, **35**: 1028–1030.

[53] Pal S. K. & King R. A., Image enhancement using smoothing with fuzzy sets, *IEEE Trans. on Systems Man and Cybernat.,* 1981, **11**: 494–501.

[54] Pal S. K. & King R. A., Image enhancement using fuzzy sets, *Electron. Lett.,* 1980, **16**: 376–378.

[55] Pitas I. & Venetsanopoulos A. N., *Nonlinear digital filters— principles and applications,* Boston: Kluwer Academic Publishers, 1990.

[56] Plataniotis K. N., Androutsos D. & Venestsanopoulos A. N., Fuzzy adaptive filters for multichannel image processing, *Signal Processing,* 1996, **55**: 93–106.

[57] Pomares H., Rojas I., Gonzalez J. & Prieto A., Structure identification incomplete rule-based fuzzy systems, *IEEE Trans. on Fuzzy Systems,* 2002, **10**(3): 349–359.

[58] Runkler T. A. & Glesner M., A set of axioms for defuzzification strategies towards a theory of rational defuzzification operators, *Proc. IEEE Intern. Conf. on Fuzzy Systems,* 1993, **2**: 1161–1166.

[59] Russo F., Recent advances in fuzzy techniques for image enhancement, *IEEE Instrumentation & Measurement Magazine,* 1998, **12**: 29–31(35).

[60] Russo F., Hybrid neuro-fuzzy filter for impulse noise removal, *Pattern Recognition,* 1999, **32**: 1843–1855.

[61] Russo F. & Ramponi G., Nonlinear fuzzy operators for image processing, *Signal Processing,* 1994, **38**: 429–440.

[62] Russo F. & Ramponi G., A fuzzy filter for images corrupted by impulse noise, *IEEE Signal Process Lett.,* 1996, **3**: 168–170.

[63] Saade J.J., A unifying approach to defuzzification and comparison of the outputs of fuzzy controllers, *IEEE Trans.on Fuzzy Systems,*1996, **4**:227–237.

[64] Saade J. J. & Diab H. B., Defuzzification techniques for fuzzy controllers, *IEEE Trans. on Systems,Man,and Cybernet.—Part B,* 2000, **30**: 223–229.

[65] Sun Qu, Li Renhou & Zhang Pingan, Stable and optimal adaptive fuzzy control of complex systems using fuzzy dynamic model, *Fuzzy Sets and systems,* 2003, **133**: 1–17.

[66] Tizhoosh H. R., Krell G. & Michaelis B., Knowledge–based enhancement of mega voltage images in radiation therapy using a hybrid neuro-fuzzy system, *Image and Vision Computing,* 2001, **19**(4): 217–233.

[67] Wang L. X., *Adaptive fuzzy systems and control: design stability analysis,* PTR Prentice-Hall, Englewood Cliffs, NJ, 1994.

[68] Wei J. S. & Sun S. H. J., Fuzzy neural nets with non-symmetric π membership functions and applications in signal processing and image analysis, *Signal Processing,* 2000, **80**(6): 965–983.

[69] Yin L., Astola J. & Neuvo Y., A new class of nonlinear filters—neural filters, *IEEE Trans. on Signal Processing,* 1993, **41**(3): 1201–1221.

[70] Yang Xinhua & Toh Peng Seng, Adaptive fuzzy multilevel median filter, *IEEE Trans. on Image Processing,* 1995, **4**(5): 680–682.

[71] Yu P. T. & Chen R. C., Fuzzy stack filters—their definition, fundamental properties and application in image processing, *IEEE Trans. on Image Processing,* 1996, **5**: 838–854.

[72] Zahlmann G. & Kochner B., Hybrid fuzzy image processing for situation assessment diabetic retinopathy, *IEEE Eng. in Medicine and Bio. Mag.* 2000, **19**(1): 76–83.

[73] Zeng X. J. & Sing M. G., A relationship between membership functions and approximation accuracy in fuzzy system, *IEEE Trans. on Systems, Man and Cybernet.-B,* 1996, **26**(1): 176–180.

[74] Zeng X. J. & Sing M. G., Approximation properties of fuzzy systems generated by the min inference, *IEEE Trans. on Systems, Man and Cybernet.-B,* 1996, **26**(1): 187–193.

Indices

NOTE: The indices include a few of notation classes. For example, universes, the basic sets in which the subjects are discussed, notations on sets and terminologies and so on. They are not intended to assist the reader in surveying the subject matter of the book (for this, see table of contents), but merely to help him locate notations and definitions.

§I.1 Symbols

§I.2 Terminologies

A

T

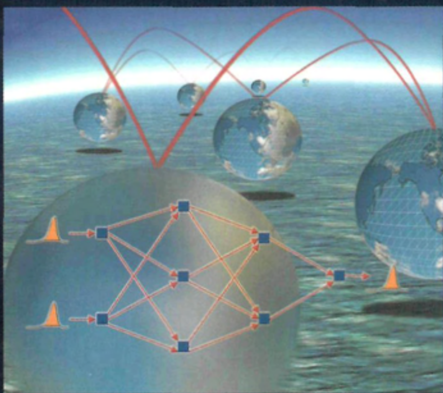
U

V

W

Z

Others



FUZZY NEURAL NETWORK THEORY AND APPLICATION

This book systematically synthesizes research achievements in the field of fuzzy neural networks in recent years. It also provides a comprehensive presentation of the developments in fuzzy neural networks, with regard to theory as well as their application to system modeling and image restoration. Special emphasis is placed on the fundamental concepts and architecture analysis of fuzzy neural networks. The book is unique in treating all kinds of fuzzy neural networks and their learning algorithms and universal approximations, and employing simulation examples which are carefully designed to help the reader grasp the underlying theory. This is a valuable reference for scientists and engineers working in mathematics, computer science, control or other fields related to information processing. It can also be used as a textbook for graduate courses in applied mathematics, computer science, automatic control and electrical engineering.

www.worldscientific.com
5493 hc
ISBN 981-238-786-2
9 789812 387868