Introduction to Fuzzy Logic using MATLAB

S.N. Sivanandam, S. Sumathi and S.N. Deepa

Introduction to Fuzzy Logic using MATLAB

With 304 Figures and 37 Tables

Springer

Dr. S.N. Sivanandam
Professor and Head
Department of Computer
Science and Engineering
PSG College of Technology
Coimbatore 641 004
Tamil Nadu, India
E-mail: snsivanandam@yahoo.co.in

S.N. Deepa
Faculty
Department of Computer
Science and Engineering
PSG College of Technology
Coimbatore 641 004
Tamil Nadu, India
E-mail: deepanand1999@yahoo.co.in

Dr. S. Sumathi
Assistant Professor
Department of Electrical
and Electronics Engineering
PSG College of Technology
Coimbatore 641 004
Tamil Nadu, India
E-mail: ss_eeein@yahoo.com

Library of Congress Control Number: 2006930099

ISBN-10 3-540-35780-7 Springer Berlin Heidelberg New York
ISBN-13 978-3-540-35780-3 Springer Berlin Heidelberg New York

Springer is a part of Springer Science+Business Media.

springer.com

Typesetting by the authors and SPi
Cover design: Erich Kirchner, Heidelberg

Printed on acid-free paper SPIN 11764601 89/3100/SPi 5 4 3 2 1 0

Introduction to Fuzzy Logic using MATLAB

S.N. Sivanandam, S. Sumathi and S.N. Deepa

Introduction to Fuzzy Logic using MATLAB

With 304 Figures and 37 Tables

Springer

Dr. S.N. Sivanandam
Professor and Head
Department of Computer
Science and Engineering
PSG College of Technology
Coimbatore 641 004
Tamil Nadu, India
E-mail: snsivanandam@yahoo.co.in

S.N. Deepa
Faculty
Department of Computer
Science and Engineering
PSG College of Technology
Coimbatore 641 004
Tamil Nadu, India
E-mail: deepanand1999@yahoo.co.in

Dr. S. Sumathi
Assistant Professor
Department of Electrical
and Electronics Engineering
PSG College of Technology
Coimbatore 641 004
Tamil Nadu, India
E-mail: ss_eeein@yahoo.com

Library of Congress Control Number: 2006930099

ISBN-10 3-540-35780-7 Springer Berlin Heidelberg New York
ISBN-13 978-3-540-35780-3 Springer Berlin Heidelberg New York

Springer is a part of Springer Science+Business Media.

springer.com

Typesetting by the authors and SPi
Cover design: Erich Kirchner, Heidelberg

Printed on acid-free paper SPIN 11764601 89/3100/SPi 5 4 3 2 1 0

Preface

The world we live in is becoming ever more reliant on the use of electronics and computers to control the behavior of real-world resources. For example, an increasing amount of commerce is performed without a single banknote or coin ever being exchanged. Similarly, airports can safely land and send off airplanes without ever looking out of a window. Another, more individual, example is the increasing use of electronic personal organizers for organizing meetings and contacts. All these examples share a similar structure; multiple parties (e.g., airplanes or people) come together to co-ordinate their activities in order to achieve a common goal. It is not surprising, then, that a lot of research is being done into how a lot of mechanics of the co-ordination process can be automated using computers.

Fuzzy logic means approximate reasoning, information granulation, computing with words and so on.

Ambiguity is always present in any realistic process. This ambiguity may arise from the interpretation of the data inputs and in the rules used to describe the relationships between the informative attributes. Fuzzy logic provides an inference structure that enables the human reasoning capabilities to be applied to artificial knowledge-based systems. Fuzzy logic provides a means for converting linguistic strategy into control actions and thus offers a high-level computation.

Fuzzy logic provides mathematical strength to the emulation of certain perceptual and linguistic attributes associated with human cognition, whereas the science of neural networks provides a new computing tool with learning and adaptation capabilities. The theory of fuzzy logic provides an inference mechanism under cognitive uncertainty, computational neural networks offer exciting advantages such as learning, adaptation, fault tolerance, parallelism, and generalization.

About the Book

This book is meant for a wide range of readers, especially college and university students wishing to learn basic as well as advanced processes and techniques in fuzzy systems. It can also be meant for programmers who may be involved in programming based on the soft computing applications.

The principles of fuzzy systems are dealt in depth with the information and the useful knowledge available for computing processes. The various algorithms and the solutions to the problems are well balanced pertinent to the fuzzy systems' research projects, labs, and for college- and university-level studies.

Modern aspects of soft computing have been introduced from the first principles and discussed in an easy manner, so that a beginner can grasp the concept of fuzzy systems with minimal effort.

The solutions to the problems are programmed using Matlab 6.0 and the simulated results are given. The fuzzy logic toolbox are also provided in the Appendix for easy reference of the students and professionals.

The book contains solved example problems, review questions, and exercise problems.

This book is designed to give a broad, yet in-depth overview of the field of fuzzy systems. This book can be a handbook and a guide for students of computer science, information technology, EEE, ECE, disciplines of engineering, students in master of computer applications, and for professionals in the information technology sector, etc.

This book will be a very good compendium for almost all readers — from students of undergraduate to postgraduate level and also for researchers, professionals, etc. — who wish to enrich their knowledge on fuzzy systems' principles and applications with a single book in the best manner.

This book focuses mainly on the following academic courses:

- Master of Computer Applications (MCA)
- Master of Computer and Information Technology
- Master of Science (Software)-Integrated
- Engineering students of computer science, electrical and electronics engineering, electronics and communication engineering and information technology both at graduate and postgraduate levels
- Ph.D research scholars who work in this field

Fuzzy systems, at present, is a hot topic among academicians as well as among program developers. As a result, this book can be recommended not only for students, but also for a wide range of professionals and developers who work in this area.

This book can be used as a ready reference guide for fuzzy system research scholars. Most of the algorithms, solved problems, and applications for a wide variety of areas covered in this book can fulfill as an advanced academic book.

In conclusion, we hope that the reader will find this book a truly helpful guide and a valuable source of information about the fuzzy system principles for their numerous practical applications.

Organization of the Book

The book covers 9 chapters altogether. It starts with introduction to the fuzzy system techniques. The application case studies are also discussed.

The chapters are organized as follows:

- Chapter 1 gives an introduction to fuzzy logic and Matlab.
- Chapter 2 discusses the definition, properties, and operations of classical and fuzzy sets. It contains solved sample problems related to the classical and fuzzy sets.
- The Cartesian product of the relation along with the cardinality, operations, properties, and composition of classical and fuzzy relations is discussed in chapter 3.
- Chapter 4 gives details on the membership functions. It also adds features of membership functions, classification of fuzzy sets, process of fuzzification, and various methods by means of which membership values are assigned.
- The process and the methods of defuzzification are described in chapter 5. The lambda cut method for fuzzy set and relation along with the other methods like centroid method, weighted average method, etc. are discussed with solved problems inside.
- Chapter 6 describes the fuzzy rule-based system. It includes the aggregation, decomposition, and the formation of rules. Also the methods of fuzzy inference system, mamdani, and sugeno methods are described here.
- Chapter 7 provides the information regarding various decision-making processes like fuzzy ordering, individual decision making, multiperson decision making, multiobjective decision making, and fuzzy Bayesian decision-making method.
- The application of fuzzy logic in various fields along with case studies and adaptive fuzzy in image segmentation is given in chapter 8.
- Chapter 9 gives information regarding a few projects implemented using the fuzzy logic technique.
- The appendix includes fuzzy Matlab tool box.
- The bibliography is given at the end after the appendix chapter.

Salient Features of Fuzzy Logic

The salient features of this book include

- Detailed description on fuzzy logic techniques
- Variety of solved examples

- Review questions and exercise problems
- Simulated results obtained for the fuzzy logic techniques using Matlab version 6.0
- Application case studies and projects on fuzzy logic in various fields.

S.N. Sivanandam completed his B.E (Electrical and Electronics Engineering) in 1964 from Government College of Technology, Coimbatore, and M.Sc (Engineering) in Power System in 1966 from PSG College of Technology, Coimbatore. He acquired PhD in Control Systems in 1982 from Madras University. His research areas include modeling and simulation, neural networks, fuzzy systems and genetic algorithm, pattern recognition, multidimensional system analysis, linear and nonlinear control system, signal and image processing, control system, power system, numerical methods, parallel computing, data mining, and database security. He received "Best Teacher Award" in 2001, "Dhakshina Murthy Award for Teaching Excellence" from PSG College of Technology, and "The Citation for Best Teaching and Technical Contribution" in 2002 from Government College of Technology, Coimbatore. He is currently working as a Professor and Head of Computer Science and Engineering Department, PSG College of Technology, Coimbatore. He has published nine books and is a member of various professional bodies like IE (India). ISTE, CSI, ACS, etc. He has published about 600 papers in national and international journals.

S. Sumathi completed B.E. (Electronics and Communication Engineering), M.E. (Applied Electronics) at Government College of Technology, Coimbatore, and Ph.D. in data mining. Her research interests include neural networks, fuzzy systems and genetic algorithms, pattern recognition and classification, data warehousing and data mining, operating systems, parallel computing, etc. She received the prestigious gold medal from the Institution of Engineers Journal Computer Engineering Division for the research paper titled "Development of New Soft Computing Models for Data Mining" and also best project award for UG Technical Report titled "Self-Organized Neural Network Schemes as a Data Mining Tool." Currently, she is working as Lecturer in the Department of Electrical and Electronics Engineering, PSG College of Technology, Coimbatore. Sumathi has published several research articles in national and international journals and conferences.

Deepa has completed her B.E. from Government College of Technology, Coimbatore, and M.E. from PSG College of Technology, Coimbatore. She was a gold medallist in her B.E. exams. She has published two books and articles in national and international journals and conferences. She was a recipient of national award in the year 2004 from ISTE and Larsen & Toubro Limited. Her research areas include neural network, fuzzy logic, genetic algorithm, digital control, adaptive and nonlinear control.

Coimbatore, India
2006–2007

S.N. Sivanandam
S. Sumathi
S.N. Deepa

Acknowledgments

The authors are always thankful to the Almighty for perseverance and achievements. They wish to thank Shri. G. Rangasamy, Managing Trustee, PSG Institutions; Shri. C.R. Swaminathan, Chief Executive; and Dr. R. Rudramoorthy, Principal, PSG College of Technology, Coimbatore, for their whole-hearted cooperation and great encouragement given in this successful endeavor. Sumathi owes much to her daughter Priyanka and to the support rendered by her husband, brother and family. Deepa wishes to thank her husband Anand and her daughter Nivethitha, and her parents for their support.

Contents

1 Introduction

1.1 Fuzzy Logic

In the literature sources, we can find different kinds of justification for fuzzy systems theory. Human knowledge nowadays becomes increasingly important – we gain it from experiencing the world within which we live and use our ability to reason to create order in the mass of information (i.e., to formulate human knowledge in a systematic manner). Since we are all limited in our ability to perceive the world and to profound reasoning, we find ourselves everywhere confronted by *uncertainty* which is a result of lack of information (lexical impression, incompleteness), in particular, inaccuracy of measurements. The other limitation factor in our desire for precision is a natural language used for describing/sharing knowledge, communication, etc. We understand core meanings of word and are able to communicate accurately to an acceptable degree, but generally we cannot precisely agree among ourselves on the single word or terms of common sense meaning. In short, natural languages are *vague*.

Our perception of the real world is pervaded by concepts which do not have sharply defined boundaries – for example, *many, tall, much larger than, young,* etc. are true only to some degree and they are false to some degree as well. These concepts (facts) can be called *fuzzy* or *gray* (*vague*) concepts – a human brain works with them, while computers may not do it (they reason with strings of 0s and 1s). Natural languages, which are much higher in level than programming languages, are fuzzy whereas programming languages are not. The door to the development of fuzzy computers was opened in 1985 by the design of the first logic chip by Masaki Togai and Hiroyuki Watanabe at Bell Telephone Laboratories. In the years to come fuzzy computers will employ both *fuzzy hardware* and *fuzzy software*, and they will be much closer in structure to the human brain than the present-day computers are.

The entire real world is complex; it is found that the complexity arises from uncertainty in the form of ambiguity. According to Dr. Lotfi Zadeh, Principle of Compatability, the complexity, and the imprecision are correlated and adds,

> The closer one looks at a real world problem, the fuzzier becomes its solution (Zadeh 1973)

The Fuzzy Logic tool was introduced in 1965, also by Lotfi Zadeh, and is a mathematical tool for dealing with uncertainty. It offers to a soft computing partnership the important concept of computing with words'. It provides a technique to deal with imprecision and information granularity. The fuzzy theory provides a mechanism for representing linguistic constructs such as "many," "low," "medium," "often," "few." In general, the fuzzy logic provides an inference structure that enables appropriate human reasoning capabilities. On the contrary, the traditional binary set theory describes crisp events, events that either do or do not occur. It uses probability theory to explain if an event will occur, measuring the chance with which a given event is expected to occur. The theory of fuzzy logic is based upon the notion of relative graded membership and so are the functions of mentation and cognitive processes. The utility of fuzzy sets lies in their ability to model uncertain or ambiguous data, Fig. 1.1, so often encountered in real life.

It is important to observe that there is an *intimate connection* between *Fuzziness* and *Complexity*. As the complexity of a task (problem), or of a system for performing that task, exceeds a certain threshold, the system must necessarily become fuzzy in nature. Zadeh, originally an engineer and systems scientist, was concerned with the rapid decline in information afforded by traditional mathematical models as the complexity of the target system increased. As he stressed, with the increasing of complexity our ability to make precise and yet significant statements about its behavior diminishes. Real-world problems (situations) are too complex, and the *complexity involves the degree of uncertainty* – as uncertainty increases, so does the complexity of the problem. Traditional system modeling and analysis techniques are too precise for such problems (systems), and in order to make complexity less daunting we introduce appropriate simplifications, assumptions, etc. (i.e., *degree of* uncertainty or *Fuzziness*) to achieve a satisfactory compromise between the information we have and the amount of uncertainty we are willing to accept. In this aspect, fuzzy systems theory is similar to other engineering theories, because almost all of them characterize the real world in an approximate manner.

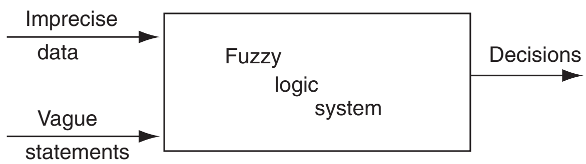

Fig. 1.1. A fuzzy logic system which accepts imprecise data and vague statements such as low, medium, high and provides decisions

Fuzzy sets provide means to model the uncertainty associated with vagueness, imprecision, and lack of information regarding a problem or a plant, etc. Consider the meaning of a "short person." For an individual X, the short person may be one whose height is below $4'25''$. For other individual Y, the short person may be one whose height is below or equal to $3'90''$. This "short" is called as a linguistic descriptor. The term "short" informs the same meaning to the individuals X and Y, but it is found that they both do not provide a unique definition. The term "short" would be conveyed effectively, only when a computer compares the given height value with the preassigned value of "short." This variable "short" is called as *linguistic variable*, which represents the imprecision existing in the system.

The uncertainty is found to arise from ignorance, from chance and randomness, due to lack of knowledge, from vagueness (unclear), like the fuzziness existing in our natural language. Lotfi Zadeh proposed the *set membership* idea to make suitable decisions when uncertainty occurs. Consider the "short" example discussed previously. If we take "short" as a height equal to or less than 4 feet, then $3'90''$ would easily become the member of the set "short" and $4'25''$ will not be a member of the set "short." The membership value is "1" if it belongs to the set or "0" if it is not a member of the set. Thus membership in a set is found to be binary i.e., the element is a member of a set or not.

It can be indicated as,

$$\chi_A(x) = \left\{ \begin{matrix} 1 & , x \in A \\ 0 & , x \notin A \end{matrix} \right\},$$

where $\chi_A(x)$ is the membership of element x in set A and A is the entire set on the universe.

This membership was extended to possess various "degree of membership" on the real continuous interval [0,1]. Zadeh formed *fuzzy sets* as the sets on the universe X which can accommodate "degrees of membership." The concept of a fuzzy set contrasts with a classical concept of a bivalent set (crisp set), whose boundary is required to be precise, i.e., a crisp set is a collection of things for which it is known whether any given thing is inside it or not. Zadeh generalized the idea of a crisp set by extending a valuation set $\{1, 0\}$ (definitely in/definitely out) to the interval of real values (degrees of membership) between 1 and 0 denoted as [0,1]. We can say that the degree of membership of any particular element of a fuzzy set express the degree of compatibility of the element with a concept represented by fuzzy set. It means that a fuzzy set A contains an object x to degree $a(x)$, i.e., $a(x) = Degree(x \in A)$, and the map $a : X \to \{Membership\ Degrees\}$ is called a *set function* or *membership function*. The fuzzy set A can be expressed as $A = \{(x, a(x))\}$, $x \in X$, and it imposes an elastic constrain of the possible values of elements $x \in X$ called the *possibility distribution*. Fuzzy sets tend to capture vagueness exclusively via membership functions that are mappings from a given universe of discourse

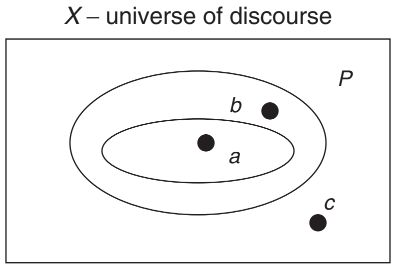

Fig. 1.2. Boundary region of a fuzzy set

X to a unit interval containing membership values. It is important to note that membership can take values between 0 and 1.

Fuzziness describes the ambiguity of an event and randomness describes the uncertainty in the occurrence of an event. It can be generally seen in classical sets that there is no uncertainty, hence they have crisp boundaries, but in the case of a fuzzy set, since uncertainty occurs, the boundaries may be ambiguously specified.

From the Fig. 1.2, it can be noted that a is clearly a member of fuzzy set P, c is clearly not a member of fuzzy set P, but the membership of b is found to be vague. Hence a can take membership value 1, c can take membership value 0 and b can take membership value between 0 and 1 [0 to 1], say 0.4, 0.7, etc. This is set to be a partial member ship of fuzzy set P.

The membership function for a set maps each element of the set to a membership value between 0 and 1 and uniquely describes that set. The values 0 and 1 describe "not belonging to" and "belonging to" a conventional set respectively; values in between represent "fuzziness." Determining the membership function is subjective to varying degrees depending on the situation. It depends on an individual's perception of the data in question and does not depend on randomness. This is important, and distinguishes fuzzy set theory from probability theory (Fig. 1.3).

In practice fuzzy logic means computation of words. Since computation with words is possible, computerized systems can be built by embedding human expertise articulated in daily language. Also called a fuzzy inference engine or fuzzy rule-base, such a system can perform approximate reasoning somewhat similar to but much more primitive than that of the human brain. Computing with words seems to be a slightly futuristic phrase today since only certain aspects of natural language can be represented by the calculus of fuzzy sets, but still fuzzy logic remains one of the most practical ways to mimic human expertise in a realistic manner. The fuzzy approach uses a premise that humans do not represent classes of objects (e.g. *class of bald men*, or the *class of numbers which are much greater than 50*) as fully disjoint but rather as sets in which there may be grades of membership intermediate between full

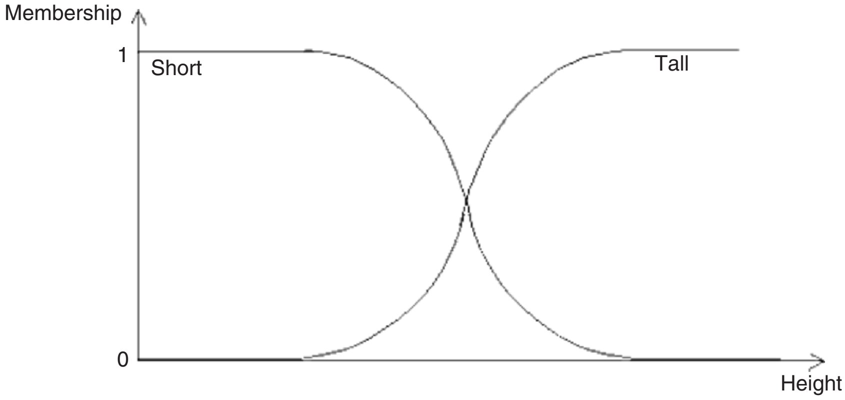

Fig. 1.3. *The fuzzy sets "tall" and "short."* The classification is subjective – it depends on what height is measured relative to. At the extremes, the distinction is clear, but there is a large amount of overlap in the middle

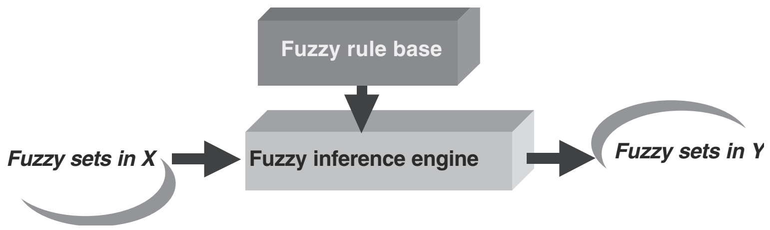

Fig. 1.4. Configuration of a pure fuzzy system

membership and non-membership. Thus, a fuzzy set works as a concept that makes it possible to *treat fuzziness in a quantitative manner*.

Fuzzy sets form the building blocks for fuzzy *IF–THEN* rules which have the general form "IF X is A THEN Y is B," where A and B are fuzzy sets. The term "fuzzy systems" refers mostly to systems that are governed by fuzzy IF–THEN rules. The IF part of an implication is called the *antecedent* whereas the second, THEN part is a *consequent*. A fuzzy system is a set of fuzzy rules that converts inputs to outputs. The basic configuration of a pure fuzzy system is shown in Fig. 1.4. The fuzzy inference engine (algorithm) combines fuzzy *IF–THEN* rules into a mapping from fuzzy sets in the input space X to fuzzy sets in the output space Y based on fuzzy logic principles. From a knowledge representation viewpoint, a fuzzy *IF–THEN* rule is a scheme for capturing knowledge that involves imprecision. The main feature of reasoning using these rules is its *partial matching* capability, which enables an inference to be made from a fuzzy rule even when the rule's condition is only partially satisfied.

Fuzzy systems, on one hand, are rule-based systems that are constructed from a collection of linguistic rules; on the other hand, fuzzy systems are nonlinear mappings of inputs (stimuli) to outputs (responses), i.e., certain types of fuzzy systems can be written as compact nonlinear formulas. The inputs and outputs can be numbers or vectors of numbers. These rule-based systems in theory model represents any system with arbitrary accuracy, i.e., they work as *universal approximators*.

The Achilles' heel of a fuzzy system is its rules; smart rules give smart systems and other rules give smart systems and other rules give less smart or even dumb systems. The *number of rules* increases exponentially with the dimension of the input space (number of system variables). This rule explosion is called the *principle of dimensionality* and is a general problem for mathematical models. For the last five years several approaches based on decomposition (cluster) merging and fusing have been proposed to overcome this problem.

Hence, Fuzzy models are not replacements for probability models. The fuzzy models sometimes found to work better and sometimes they do not. But mostly fuzzy is evidently proved that it provides better solutions for complex problems.

1.2 Mat LAB – An Overview

Dr Cleve Moler, Chief scientist at MathWorks, Inc., originally wrote Matlab, to provide easy access to matrix software developed in the LINPACK and EISPACK projects. The very first version was written in the late 1970s for use in courses in matrix theory, linear algebra, and numerical analysis. Matlab is therefore built upon a foundation of sophisticated matrix software, in which the basic data element is a matrix that does not require predimensioning.

Matlab is a product of The Math works, Inc. and is an advanced interactive software package specially designed for scientific and engineering computation. The Matlab environment integrates graphic illustrations with precise numerical calculations, and is a powerful, easy-to-use, and comprehensive tool for performing all kinds of computations and scientific data visualization. Matlab has proven to be a very flexible and usable tool for solving problems in many areas. Matlab is a high-performance language for technical computing. It integrates computation, visualization, and programming in an easy-to-use environment where problems and solutions are expressed in familiar mathematical notation. Typical use includes:

– Math and computation
– Algorithm development
– Modeling, simulation, and prototyping
– Data analysis, exploration, and visualization
– Scientific and engineering graphics
– Application development, including graphical user interface building

Matlab is an interactive system whose basic elements are an array that does not require dimensioning. This allows solving many computing problems, especially those with matrix and vector formulations, in a fraction of the time it would take to write a program in a scalar noninteractive language such as C or FORTRAN. Mathematics is the common language of science and engineering. Matrices, differential equations, arrays of data, plots, and graphs are the basic building blocks of both applied mathematics and Matlab. It is the underlying mathematical base that makes Matlab accessible and powerful. Matlab allows expressing the entire algorithm in a few dozen lines, to compute the solution with great accuracy in about a second.

Matlab is both an environment and programming language, and the major advantage of the Matlab language is that it allows building our own reusable tools. Our own functions and programs (known as M-files) can be created in Matlab code. The toolbox is a specialized collection of M-files for working on particular classes of problems. The Matlab documentation set has been written, expanded, and put online for ease of use. The set includes online help, as well as hypertext-based and printed manuals. The commands in Matlab are expressed in a notation close to that used in mathematics and engineering. There is a very large set of commands and functions, known as Matlab M-files. As a result solving problems in Matlab is faster than the other traditional programming. It is easy to modify the functions since most of the M-files can be open. For high performance, the Matlab software is written in optimized C and coded in assembly language.

Matlab's two- and three-dimensional graphics are object oriented. Matlab is thus both an environment and a matrix/vector-oriented programming language, which enables the use to build own required tools.
The main features of Matlab are:

- Advance algorithms for high-performance numerical computations, especially in the field of matrix algebra.
- A large collection of predefined mathematical functions and the ability to define one's own functions.
- Two- and three-dimensional graphics for plotting and displaying data.
- A complete help system online.
- Powerful matrix/vector-oriented high-level programming language for individual applications.
- Ability to cooperate with programs written in other languages and for importing and exporting formatted data.
- Toolboxes available for solving advanced problems in several application areas.

Figure 1.5 shows the main features and capabilities of Matlab.

SIMULINK is a Matlab toolbox designed for the dynamic simulation of linear and nonlinear systems as well as continuous and discrete-time systems. It can also display information graphically. Matlab is an interactive package for numerical analysis, matrix computation, control system design, and linear system analysis and design available on most CAEN platforms (Macintosh,

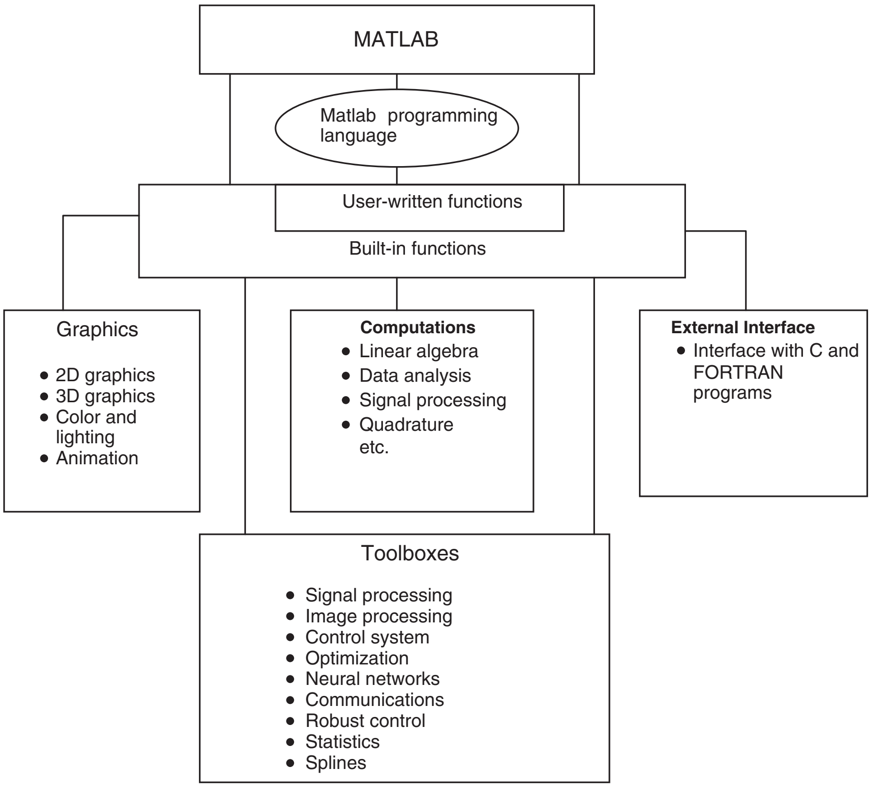

Fig. 1.5. Features and capabilities of Matlab

PCs, Sun, and Hewlett-Packard). In addition to the standard functions provided by Matlab, there exist large set of toolboxes, or collections of functions and procedures, available as part of the Matlab package. The toolboxes are:

– *Control system*. Provides several features for advanced control system design and analysis
– *Communications*. Provides functions to model the components of a communication system's physical layer
– *Signal processing*. Contains functions to design analog and digital filters and apply these filters to data and analyze the results
– *System identification*. Provides features to build mathematical models of dynamical systems based on observed system data
– *Robust control*. Allows users to create robust multivariable feedback control system designs based on the concept of the singular value Bode plot
– *Simulink*. Allows you to model dynamic systems graphically
– *Neural network*. Allows you to simulate neural networks
– *Fuzzy logic*. Allows for manipulation of fuzzy systems and membership functions

– *Image processing.* Provides access to a wide variety of functions for reading, writing, and filtering images of various kinds in different ways
– *Analysis.* Includes a wide variety of system analysis tools for varying matrices
– *Optimization.* Contains basic tools for use in constrained and unconstrained minimization problems
– *Spline.* Can be used to find approximate functional representations of data sets
– *Symbolic.* Allows for symbolic (rather than purely numeric) manipulation of functions
– *User interface utilities.* Includes tools for creating dialog boxes, menu utilities, and other user interaction for script files

Matlab has been used as an efficient tool, all over this text to develop the applications based on neural net, fuzzy systems and genetic algorithm.

Review Questions

1) Define uncertainty and vagueness
2) Compare – precision an impression
3) Explain the concept of fuzziness a said by Lotfi A. Zadeh
4) What is a membership function?
5) Describe in detail about fuzzy system with basic configuration
6) Write short note on "degree of uncertainty"
7) Write an over view of Mat Lab

2

Classical Sets and Fuzzy Sets

2.1 Introduction

The theory on classical sets and the basic ideas of the fuzzy sets are discussed in detail in this chapter. The various operations, laws and properties of fuzzy sets are introduced along with that of the classical sets. The classical set we are going to deal is defined by means of the definite or crisp boundaries. This means that there is no uncertainty involved in the location of the boundaries for these sets. But whereas the fuzzy set, on the other hand is defined by its vague and ambiguous properties, hence the boundaries are specified ambiguously. The crisp sets are sets without ambiguity in their membership. The fuzzy set theory is a very efficient theory in dealing with the concepts of ambiguity. The fuzzy sets are dealt after reviewing the concepts of the classical or crisp sets.

2.2 Classical Set

Consider a classical set where X denotes the universe of discourse or universal sets. The individual elements in the universe X will be denoted as x. The features of the elements in X can be discrete, countable integers, or continuous valued quantities on the real line. Examples of elements of various universes might be as follows.

– The clock speeds of computers CPUs.
– The operating temperature of an air conditioner.
– The operating currents of an electronic motor or a generator set.
– The integers 1–100.

Choosing a universe that is discrete and finite or one that it continuous and infinite is a modeling choice, the choice does not alter the characterization of sets defined on the universe. If the universe possesses continuous elements, then the corresponding set defined on the universe will also be continuous.

The total number of elements in a universe X is called its *cardinal number* and is denoted by η_x. Discrete universe is composed of countable finite collection of elements and has a finite cardinal number and the continuous universe consists of uncountable or infinite collection of elements and thus has a infinite cardinal number.

As we all know, the collection of elements in the universe are called as sets, and the collections of elements within sets are called as subsets. The collection of all the elements in the universe is called the whole set. The null set $\emptyset$, which has no elements is analogous to an impossible event, and the whole set is analogous to certain event. Power set constitutes all possible sets of X and is denoted by $P(X)$.

Example 2.1. Let universe comprised of four elements $X = \{1, 2, 3, 4\}$ find cardinal number, power set, and cardinality of the power set.

Solution. The cardinal number is the number of elements in the defined set. The defined set X consists of four elements $1, 2, 3$, and 4. Therefore, the Cardinal number $= \eta_x = 4$.

The power set consists of all possible sets of X. It is given by,

Power set $P(x) = \{\emptyset, \{1\}, \{2\}, \{3\}, \{4\}, \{1, 2\}, \{1, 3\}, \{1, 4\}, \{2, 3\}, \{2, 4\}, \{3, 4\}, \{1, 2, 3\}, \{2, 3, 4\}, \{1, 3, 4\}, \{1, 2, 4\}, \{1, 2, 3, 4\}\}$

Cardinality of the power set is given by,

$\eta_{P(x)} = 2^{\eta_x} = 2^4 = 16$.

2.2.1 Operations on Classical Sets

There are various operations that can be performed in the classical or crisp sets. The results of the operation performed on the classical sets will be definite. The operations that can be performed on the classical sets are dealt in detail below:

Consider two sets A and B defined on the universe X. The definitions of the operation for classical sets are based on the two sets A and B defined on the universe X.

Union

The Union of two classical sets A and B is denoted by $A \cup B$. It represents all the elements in the universe that reside in either the set A, the set B or both sets A and B. This operation is called the logical OR.

In set theoretic form it is represented as

$$A \cup B = \{x/x \in A \text{ or } x \in B\}\,.$$

In Venn diagram form it can be represented as shown in Fig. 2.1.

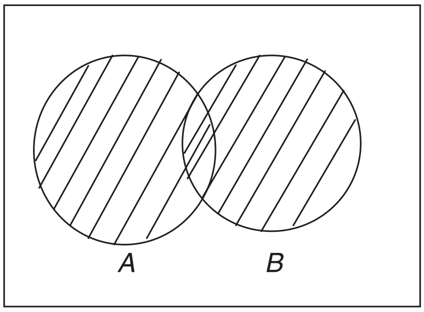

Fig. 2.1. $A \cup B$

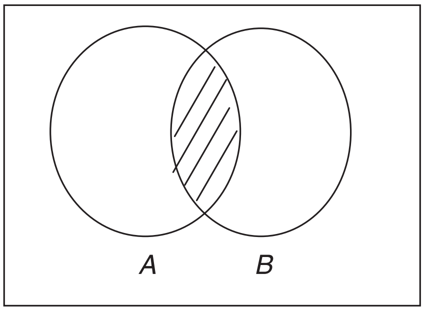

Fig. 2.2. $A \cap B$

Intersection

The intersection of two sets A and B is denoted $A \cap B$. It represents all those elements in the universe X that simultaneously reside in (or belongs to) both sets A and B.

In set theoretic form it is represented as

$$A \cap B = \{x/x \in A \text{ and } x \in B\} .$$

In Venn diagram form it can be represented as shown in Fig. 2.2.

Complement

The complement of set A denoted $\overline{A}$, is defined as the collection of all elements in the universe that do not reside in the set A.

In set theoretic form it is represented as

$$\overline{A} = \{x/x \notin A, x \in X\} .$$

In Venn diagram form it is represented as shown in Fig. 2.3.

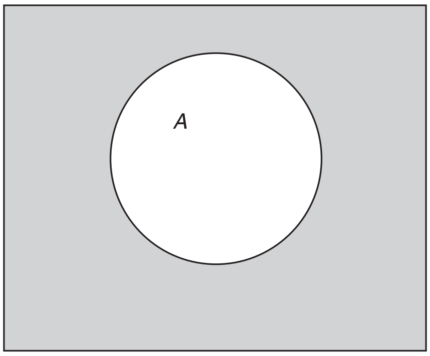

Fig. 2.3. Complement of set A

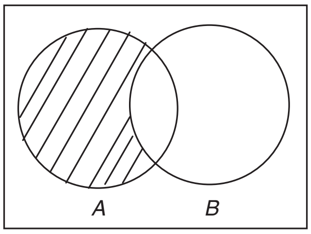

Fig. 2.4. Difference $A|B$

Difference

The difference of a set A with respect to B, denoted $A|B$ is defined as collection of all elements in the universe that reside in A and that do not reside in B simultaneously.

In set theoretic form it is represented as

$$A|B = \{x/x \in A \text{ and } x \notin B\} .$$

In Venn diagram form it is represented as shown in Fig. 2.4.

2.2.2 Properties of Classical Sets

In any mathematical operations the properties plays a major role. Based upon the properties, the solution can be obtained for the problems. The following are the important properties of classical sets:
Commutativity

$$A \cup B = B \cup A,$$

$$A \cap B = B \cap A.$$

Associativity

$$A \cup (B \cup C) = (A \cup B) \cup C,$$
$$A \cap (B \cap C) = (A \cap B) \cap C.$$

Distributivity

$$A \cup (B \cap C) = (A \cup B) \cap (A \cup C),$$
$$A \cap (B \cup C) = A \cap B) \cup (A \cap C).$$

Idempotency

$$A \cup A = A,$$
$$A \cap A = A.$$

Identity

$$A \cup \phi = A$$
$$A \cap X = A$$
$$A \cap \phi = \phi$$
$$A \cup X = X.$$

Transitivity

$$\text{If } A \subseteq B \subseteq C, \text{ then } A \subseteq C.$$

In this case the symbol $\subseteq$ means contained in or equivalent to and $\subset$ means contained in.

Involution

$$\overline{\overline{A}} = A.$$

The other two important special properties include the Excluded middle laws and the Demorgan's law.

Excluded middle law includes the law of excluded middle and the law of contradiction. The excluded middle laws is very important because these are the only set operations that are not valid for both classical and fuzzy sets.

Law of excluded middle. It represents union of a set A and its complement.

$$A \cup \overline{A} = X.$$

Law of contradication. It represents the intersection of a set A and its complement

$$A \cap \overline{A} = \phi.$$

De Morgan's Law

These are very important because of their efficiency in proving the tautologies and contradictions in logic. The demorgan's law are given by

$$\overline{A \cap B} = \overline{A} \cup \overline{B},$$
$$\overline{A \cup B} = \overline{A} \cap \overline{B}.$$

In Venn diagram form it is represented as shown in Fig. 2.5.

The complement of a union or an intersection of two sets is equal to the intersection or union of the respective complements of the two sets. This is the statement made for the demorgan's law.

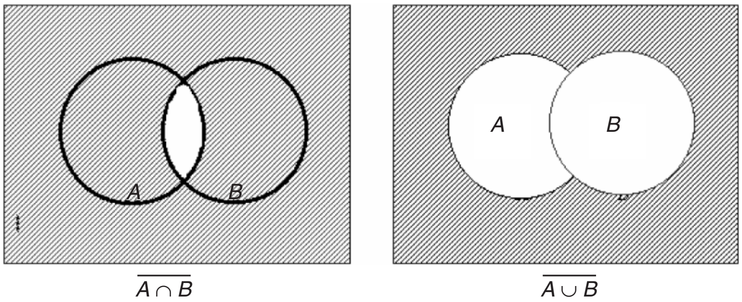

Fig. 2.5. Demorgan's law

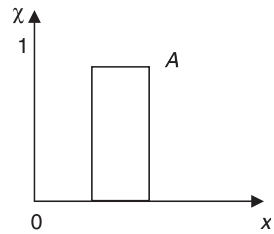

Fig. 2.6. Membership mapping for Crisp Set A

2.2.3 Mapping of Classical Sets to a Function

Mapping of set theoretic forms to function theoretic forms is an important concept. In general it can be used to map elements or subsets on one universe of discourse to elements or sets in another universe. Suppose X and Y are two different universe of discourse. If an element x is contained in X and corresponds to an element y contained in Y, it is generally represented as $f : X \rightarrow Y$, which is said as the *mapping from X to Y*. The characteristic function χ_A is defined by

$$\chi_A(x) = \begin{cases} 1 & x \in A, \\ 0 & x \notin A, \end{cases}$$

where χ_A represents the membership in set a for the elements x in the universe. The membership mapping for the crisp set A is shown in Fig. 2.6.

Let us define two sets A and B on the Universe X.

Union

The union of these two sets in terms of function theoretic form is given as follows:

$$\begin{aligned} A \cup B \rightarrow \chi_{A \cup B}(x) &= \chi_A(x) V \chi_B(x) \\ &= \max(\chi_A(x), \chi_B(x)). \end{aligned}$$

Here V indicates the maximum operator.