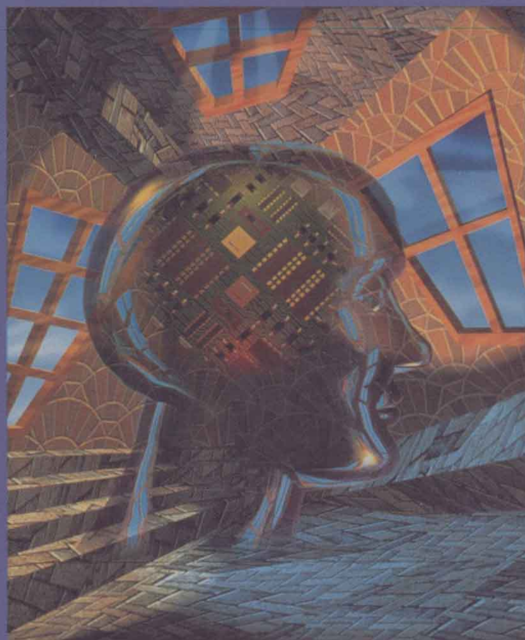
The **Fuzzy Systems** Handbook

A PRACTITIONER'S GUIDE TO BUILDING, USING, AND MAINTAINING FUZZY SYSTEMS



EARL COX

Foreword by
Lotfi Zadeh

AP PROFESSIONAL

IBM DISK with C++ Source Code

The

Fuzzy Systems Handbook

A Practitioner's Guide to Building, Using, and Maintaining Fuzzy Systems

EARL COX

The Metus Systems Group

Chappaqua, New York

AP PROFESSIONAL

Boston San Diego New York
London Sydney Tokyo Toronto

This book is printed on acid-free paper. ♾

AP PROFESSIONAL
1300 Boylston Street, Chestnut Hill, MA 02167
World Wide Web Site at http://www.apnet.com

An Imprint of ACADEMIC PRESS, INC.
A Division of HARCOURT BRACE & COMPANY

United Kingdom Edition published by
ACADEMIC PRESS LIMITED
24–28 Oval Road, London NW1 7DX

Library of Congress Cataloging-in Publication Data
Cox, Earl.
The fuzzy systems handbook: a practitioner's guide to building,
using, and maintaining fuzzy systems / Earl Cox.
p. cm.
Includes bibliographical references and index.
ISBN 0-12-194270-8
1. System design. 2. Adaptive control systems. 3. Fuzzy systems.
I. Title.
QA76.9S88C72 1994 93-42608
003'7--dc20 CIP

Printed in the United States of America
96 97 98 99 IP 9 8 7 6 5 4

This book is dedicated to my wonderful wife Marilyn.
Without her encouragement, insight, and patience
this handbook would not have been possible.

***

Foreword

My 1965 paper on fuzzy sets was motivated in large measure by the conviction that traditional methods of systems analysis are unsuited for dealing with systems in which relations between variables do not lend themselves to representation in terms of differential or difference equations. Such systems are the norm in biology, sociology, economics and, more generally, in fields in which the systems are humanistic rather than mechanistic in nature.

Traditional methods of analysis are oriented toward the use of numerical techniques. By contrast, much of human reasoning involves the use of variables whose values are fuzzy sets. This observation is the basis for the concept of a linguistic variable, that is, a variable whose values are words rather than numbers. The concept of a linguistic variable has played and is continuing to play a key role in the applications of fuzzy set theory and fuzzy logic.

The use of linguistic variables represents a significant paradigm shift in systems analysis. More specifically, in the linguistic approach the focus of attention in the representation of dependencies shifts from difference and differential equations to fuzzy if-then rules in the form *if X is A then Y is B*, where *X* and *Y* are linguistic variables and *A* and *B* are their linguistic values, e.g., *if Pressure is high then Volume is low.* Such rules serve to characterize imprecise dependencies and constitute the point of departure for the construction of what might be called the calculus of fuzzy rules or CFR, for short. In this perspective, *The Fuzzy Systems Handbook* may be viewed as an up-to-date, informative and easy-to-follow introduction to the methodology of fuzzy if-then rules and its applications. A key aspect of this methodology is that its role model is the human mind.

Although I am not an impartial observer, I have no doubt that, in the years ahead, CFR will become an essential tool in the representation and analysis of dependencies which are (a) intrinsically imprecise or (b) can be treated imprecisely to exploit the tolerance for imprecision. At this juncture, many of the applications of fuzzy set theory and fuzzy logic in the realm of knowledge systems fall into the first category. By contrast, most of the applications in the realm of control fall into the second category. This is especially true of the applications of CFR in the design of consumer products such as washing machines, cameras, air conditioners and vacuum cleaners. In such products, the methodology of fuzzy if-then rules serves to increase the MIQ (Machine Intelligence Quotient) and lower the cost of products by making it possible to program their behavior in the language of fuzzy rules — and thus lessen the need for precision in data gathering and data manipulation. In this context, the use of linguistic variables and fuzzy rules may be viewed as a form of data compression.

The numerous applications of fuzzy set theory and fuzzy logic described in *The Fuzzy Systems Handbook* underscore the unreality of traditional methods in dealing with decision-making in an environment of uncertainty and imprecision. A prominent flaw of such methods relates to the assumption that probabilities can be assumed to be expressible as real numbers. In fact, most real-world probabilities are fuzzy rather than crisp. As a case in point, consider the following problem: I want to insure my car against theft. In this connection, I have to know the probability that my car might be stolen. What is it? The answer is: There is no way for estimating the probability in question as a real number. But what fuzzy set theory and fuzzy logic make possible is to estimate the probability in question as a fuzzy number. The fuzzy probability, then, could be used to compute the fuzzy expected loss and thus ultimately lead to a crisp decision.

Many of the real-world problems arising in the analysis and design of decision, control and knowledge systems are far from simple. Fuzzy set theory and fuzzy logic do not replace the existing methods for dealing with such problems. Rather, they provide additional tools which significantly enlarge the domain of problems which can be solved. Earl Cox's *Fuzzy Systems Handbook* is an important contribution in this regard. Oriented toward applications, it is certain to have a wide-ranging impact by showing how fuzzy set theory and fuzzy logic can be employed to exploit the tolerance for imprecision and how they make it possible to address problems that lie beyond the reach of traditional methods.

Lotfi A. Zadeh
November 1, 1993

Acknowledgments

The book represents the composite knowledge of many friends and associates over the years that I have been designing and using fuzzy logic systems. Foremost among those is Lotfi Zadeh, the founder of fuzzy set theory. Lotfi's encouragements, criticisms, and critical view of fuzzy systems as well as his review of early handbook manuscripts were invaluable. A decidedly long lasting and pervasive influence on my view of the potential of fuzzy logic and its application to information systems belongs to my friend and fuzzy logic pioneer, Peter Llewelyn Jones. Bart Kosko, with whom I have had a lively debate concerning the merits of linguistic variables versus a more rigorous mathematical view of fuzzy systems, contributed much to the initial explanation of fuzzy systems as universal approximators (all errors of interpretation, however, are mine.) I owe a special thanks to Steve Marsh, Jim Huffman, and Bob Seaton at Motorola's Center for Emerging Computer Technologies in Austin, Texas. Their vision and work in fuzzy logic as well as their (often revealing) feed-back on my views about fuzzy system modeling helped shape many of my ideas about the interpretation and direction of fuzzy systems. Others who provided invaluable insights, ideas, and criticisms include: Tom Parish at the TPIS Group in Austin, Arnold Siegel at GEC Marconi, and Tomohiro Takagi, deputy director of LIFE, in Yokohama, Japan. Some of the ideas on how to handle output fuzzy set saturation in dense, complex fuzzy systems as well as ways to encode a fuzzy explanatory facility belong to my friend Roger Stein, a senior analyst at Moody's Investors Service in New York.

This book grew out of a series of articles on fuzzy logic I wrote for *AI Expert* magazine in the latter part of 1992 and 1993, from my monthly *AI Apprentice* columns in the same magazine, and from a series of articles on conventional and adaptive fuzzy systems that appeared in the *IEEE Spectrum* magazine. Alan Zeichick, then editor of *AI Expert*, and Mike Reisenman, senior editor for *IEEE Spectrum*, both contributed editorial suggestions that improved the readability of these articles and helped shape the way in which this handbook is written.

And my special thanks go to Kathleen Tibbetts, David Pallai, and Michael Lindgren at AP PROFESSIONAL who provided encouragement and editorial assistance. Their efforts made this handbook possible.

Preface

Understanding comes from doing.

—Mr. Robert Miller
My Twelfth-Grade Physics Teacher
Glen Burnie Senior High, Class of 1964

While several books are available today that address the mathematical and philosophical foundations of fuzzy logic, none, unfortunately, provides the practicing knowledge engineer, system analyst, and project manager with specific, practical information about fuzzy system modeling. Those few books that include applications and case studies concentrate almost exclusively on engineering problems: pendulum balancing, truck backer-uppers, cement kilns, antilock braking systems, image pattern recognition, and digital signal processing. Yet the application of fuzzy logic to engineering problems represents only a fraction of its real potential. As a method of encoding and using human knowledge in a form that is very close to the way experts think about difficult, complex problems, fuzzy systems provide the facilities necessary to break through the computational bottlenecks associated with traditional decision support and expert systems. Additionally, fuzzy systems provide a rich and robust method of building systems that include multiple conflicting, cooperating, and collaborating experts (a capability that generally eludes not only symbolic expert system users but analysts that have turned to such related technologies as neural networks and genetic algorithms).

Yet the application of fuzzy logic in the areas of information technology, decision support, and database analysis and mining has been largely ignored by both the commercial vendors of decision support products and the knowledge engineers that use them. Fuzzy logic has not found its way into the information modeling field due to a number of factors that are rapidly changing—unfamiliarity with the concept, a predilection for the use of confidence factors and Bayesian probabilities among most knowledge engineers (stemming from the early successes of expert systems such as *MYCIN*, *PROSPECTOR*, and *XCON*), and a suspicion that there is something fundamentally wrong with a reasoning system that announces its own imprecision. Fuzzy logic is the essential oxymoron.

Fuzzy logic, however, is a technology that has patiently bided its time. Today, in the world of highly complex, international business systems, webs of communications networks, high-density information overloads, and the recognition that many seemingly simple problems belie a deep nonlinearity, fuzzy logic is proving itself as a powerful tool in knowledge modeling. Fuzzy logic will soon usher in the second wave of intelligent systems.

I have good reason to believe this prediction.

A little more than 13 years ago, while marketing an enterprise modeling system in the United Kingdom, I was introduced to the idea of fuzzy logic by my friend Peter Llewelyn Jones. Peter is the author of *REVEAL*, the first commercial fuzzy expert system and, with Ian Graham, the author of *Expert Systems: Knowledge, Uncertainty and Decision*,[1] one of the very first books on fuzzy information systems. Sitting one evening in a pub just off Fleet Street, and tucked neatly into the outskirts of Covent Gardens, about a block from our office on the Waterloo Bridge, Peter explained in clear and convincing terms just why fuzzy logic, in the more general form of approximate reasoning, was an important emerging technology. Like Paul on the road to Damascus, a brilliant light went off in my mind and I left the pub an eager devotee to the cult of fuzzy logic. Like all naive revolutionaries, we expected the world to welcome our insights and revealed truths with open arms. However, in spite of its evident potential and the success of many projects, *REVEAL* was shelved by its owners and fuzzy logic remained the arcane study of Lotfi Zadeh and his ever but slowly increasing circle of believers (usually graduate students who remained well within the sheltering walls of Evans Hall high on a hill at the University of California at Berkeley).

In the years since I was introduced to and began using fuzzy logic, I have seen first-hand the power and breadth that fuzzy decision and expert systems bring to a wide spectrum of unusually difficult problems. I have architected, designed, and programmed three production fuzzy expert systems. These tools have been successfully applied to large, real-world applications in such areas as transportation, managed health care, financial services, insurance risk assessment, database information mining, company stability analysis, multiresource and multiproject management, fraud detection, acquisition suitability studies, new product marketing, and sales analysis. Generally, the final models were less complex, smaller, and easier to build, implement, maintain, and extend than similar systems built using conventional symbolic expert systems.

Nature of This Handbook

You can use this handbook as a guide in evaluating possible fuzzy system projects, designing model policies, developing the underlying programming code, integrating model components, and evaluating overall model performance. The book assumes no previous familiarity with fuzzy logic, first-order predicate calculus, or Bayesian probabilities, but it does presuppose a moderately deep familiarity with business system analysis, high school algebra, and C++ programming. In general, the concentration of concepts, examples, and application code listings is geared toward real-world implementation issues. The case studies are also drawn from actual fuzzy models in such areas as new product pricing, risk assessment, and inventory management. While a few of the examples have been slightly simplified for illustrative reasons,[2] I have attempted to retain the flavor of robustness, error tolerance, and thoroughness that are required in actual commercial applications.

This handbook examines the architecture, function, and operation of rule-based fuzzy systems in a business environment. In general we will be concerned with understanding the basics of fuzzy logic and fuzzy systems, selecting the appropriate problems and domain representation, the design of fuzzy sets, the use of operators, and interpreting model output. Less attention is paid to the mathematical principles underlying the nature of fuzzy logic making the handbook a minimally mathematical tour of how fuzzy systems are designed, built, delivered, and maintained. A comprehensive bibliography provides an annotated listing of books that can provide a complete explanation of the mathematics underlying fuzzy logic.

This handbook is conceptually divided into two major parts. The first part, Chapters 1 through 6, deals with the fundamental issues of imprecision, fuzzy sets, fuzzy operators of various kinds, the handling of fuzzy propositions (also known as rules), and how a final result is resolved from a fuzzy model. The second part, Chapters 7 through 9, addresses the idea of fuzzy models and fuzzy systems by exploring the nature of fuzzy models and how they are represented; the architecture of operation of several fuzzy system case studies, and the outline of a general methodology for developing a fuzzy system. The handbook concluded with a description of the fuzzy modeling software in Chapter 10, a glossary of terms and a bibliography.

Adapting and Using the C++ Code Library

The C++ code included in this handbook has been compiled and tested under Borland's C++ 3.1 and Microsoft's Visual C++ 1.0 compilers. Although C++ is emerging as the standard language of choice for workstation-based systems, the exclusive use of object-oriented constructs would have unnecessarily restricted the handbook's audience. The code in this handbook is provided in "flat" C++, that is, without object encapsulation. Reader's familiar with the C++ CLASS organization should have little difficulty in converting the code into a form compatible with their own applications. The larger audience of readers using C or recently moving into C++ should find the modeling code easy to understand, modify, and integrate with their own code. In nearly all cases, when given a choice between self-documentation and "efficiency" or "compactness" a decision to write simple, straightforward code was always made.

Since this fuzzy logic code is extracted from a larger library of modeling software used to build production decision and expert systems, the code in the handbook also contains some functions not directly associated with fuzzy logic (such as the error message handler, the line entokeners, string managers, etc.).

The Graphical Representation of Fuzzy Sets

The code libraries included with this handbook contain several programs that draw fuzzy sets. In keeping with the general principle of maintaining platform independence, these programs write character-based graphs to a specified file. An individual fuzzy set, when plotted, will appear as the following:

```
FuzzySet:   near 2*MfgCosts
Description:
 1.00                         ..
 0.90                      ...  ....
 0.80                    ..         .
 0.70                   .            ..
 0.60                 ..               .
 0.50                .                  .
 0.40               .                    .
 0.30              .                      .
 0.20             .                        .
 0.10           ..                          ..
 0.00...........                              ........................
     0---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---0
   16.00   18.50   21.00   23.50   26.00   28.50   31.00   33.50   36.00
   Domained:        16.00 to      36.00
```

When multiple fuzzy sets are printed on the same graph, each is represented by its own character symbol. A legend following the graph indicates which fuzzy set is represented by which symbol. The following shows a PI (bell-shaped) fuzzy set and the new fuzzy set created by applying the hedge somewhat.

```
Hedge 'somewhat' applied to Fuzzy Set "PI.Curve"


      1.00                                *
      0.90                             *** ***
      0.80                            *.     .*
      0.70
      0.60                           *.       .*
      0.50                          *           *
      0.40                           .         .
      0.30                         *             *
      0.20                        * .           . *
      0.10                         .             .
      0.00************************.               .************************
          0---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---0
        16.00   20.00   24.00   28.00   32.00   36.00   40.00   44.00   48.00
     .  FuzzySet:    PI.Curve
        Description:
     *  FuzzySet:    PI.Curve
        Description: somewhat PI.Curve
        Domained:        16.00 to      48.00
```

To extract a table of domain values with their corresponding fuzzy membership values for use in graphical tools such as *Mathematica*, *MatLab*, or for use in Microsoft's *Windows* environment, the `FzyExtractSetdata` program should be used.

Icons and Topic Symbols

Throughout the handbook you will encounter a series of "sidebars" covering a wide variety of special topics. Each topic is set off from the main text and is introduced by a specific icon.

Code Discussion and Cross-reference. The diskette icon indicates that this section deals with code models, programs, and software in general. This icon usually means that a concept discussion is being tied back to the actual supporting code in one or more of the libraries.

System Internals. The plumbing icon indicates that this section discusses system internals. System internals refer to the data structures, control blocks, interprogram linkages, process flow, and other details associated with the way fuzzy modeling systems work.

Key Topics. A key icon indicates a detailed discussion on some important aspect of the system. The nature of key topics covers a broad spectrum: from hints and techniques that may not be obvious to detailed and off-line explorations of concepts and philosophies.

System Construction. The hammer icon represents discussions centered around the construction of models, applications, and high-level software systems.This topic is not specifically concerned with function code or data structures.

Practical Hints and Techniques. The wingnut ("nuts and bolts") icon indicates a discussion about the practical considerations of the current topic. This includes programming techniques, system design considerations, methodology approaches, etc.

Mathematical Topics. The ruler icon indicates a discussion of mathematical, logical, or algorithmic topics. This includes the underlying mathematics of fuzzy set theory, technical computer algorithms, topics in stability and verification, as well as discussions of performance metrics. The icon is only used for detailed discussed outside the main manual text.

Danger! Warning! The lightning bolt icon indicates a topic that could be potentially dangerous to model integrity, validation, or data. The danger icon is also used to highlight harmful side effects that are not obvious. You should read every danger topic carefully.

Reminders and Warnings. The reminder icon indicates a discussion of important, but generally not obvious, points that the user must consider when exploiting some portion of the fuzzy modeling technology.

Symbol	*Use*
~	set NOT (also complement or inversion)
$\cap$	set AND (also intersection operator)
$\cup$	set OR (also union operator)
$\aleph$	higher dimensional fuzzy space
[x,x,x]	indicates a fuzzy membership value
$\in$	member of a set (general membership)
poss(x)	the possibility of event x
prob(x)	the probability of event x
{x}	crisp or Boolean membership function
•	dyadic operator
$\xi(x)$	The expected value of a fuzzy region
μ	fuzzy membership function
$\propto$	proportionality
$\mu[x]$	membership or truth function in fuzzy set
$\Re$	element from domain of fuzzy set
$\otimes$	Cartesian product or space
$\varnothing$	empty or null set
$\supset$	implication
$\wedge$	logical AND
$\vee$	logical OR
Σ	summation

NOTES

1. 1988. Published by Chapman and Hall, London.

2. I wanted to say "for pedagogical purposes" but resisted the temptation to slip into unnecessary jargon. That's another characteristic of the handbook: unnecessary and pointedly academic jargon has usually been replaced with straightforward English. A complete polysyllabic glossary is included for academic readers who are able to actually read the handbook and thus have been bewildered and disoriented.

Contents