FUZZY LOGIC IN COMPUTER-BASED ENGINEERING

by

Hua Yao

B.Eng, Beijing University of Posts and Telecom, July, 1984

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF APPLIED SCIENCE
in the School
of
Engineering Science

SIMON FRASER UNIVERSITY
June 1993

APPROVAL

Name: Hua Yao

Degree: Master of Applied Science

Title of thesis: **Fuzzy Logic in Computer-Based Engineering**

Examining Committee: Dr. John Dill
Chair

Dr. John D. Jones
Senior Supervisor

Prof. H. Q. Yang
Supervisor

Dr. William Gruver
Supervisor

Dr. William Havens
External Examiner

Date Approved: June 28, 1993

ABSTRACT

Since it was first conceived in 1965, fuzzy logic has been an important way of simulating human reasoning. Fuzzy logic provides an effective conceptual framework for dealing with uncertainty and imprecision, which are inherent in preliminary engineering design.

In this thesis, we present an approach based on fuzzy logic and mathematical modelling of potential design components. We develop an algorithm for fuzzy calculation of functions, and compare it with previously published methods. We review some of the literature on fuzzy logic in design, in particular the work of Antonsson and his students. Then we develop a novel approach, which introduces a vocabulary of linguistic variables to describe potential design components. This vocabulary is stored in a knowledge base. Linguistic variables are also used to describe design requirements, and a metric is suggested to allow trade-off between multiple performance parameters and design requirements.

A detailed example is given for the design of Stirling engine heat exchangers.

DEDICATION

To

Lixin

and

my parents

ACKNOWLEDGMENTS

My sincere thanks to my senior supervisor Dr. John D. Jones for his valuable advice and guidance throughout the course of this research and time that he spent with me in the preparation of this thesis. I would also like to thank him for his supervision during my studies.

I also would like to thank my supervisor Prof. H. Q. Yang, visiting scholar in Simon Fraser University, for his advice and useful suggestions. I would also like to thank my fellow students and friends in the School of Engineering Science for their help.

Finally, I thank all who encouraged and supported me in my research, special acknowledge with humble gratitude, the constant encouragement and support from my husband.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

NATURE OF DESIGN

1.1 Design and Design Process

Design leads to the creation of new products, processes, software, and systems. Every field of engineering includes the design or synthesis process. The design goals are satisfied to provide the desired output, following specified requirements. Each decision made at the initial stage of design affects all subsequent stages.

Design involves a continuous interplay between *what we want to achieve* and *how we want to achieve it.* The objective is stated in the functional domain, whereas the solution is generated in the physical domain. Design is the transformation between a functional and a physical description of a device, which involves interlinking these two domains at every hierarchical level of the design process. These two domains are inherently independent of each other. So, design may be formally defined as creating an artifact's description, which meets perceived goals through mapping between the functional domain and design domain [19, 21].

Design involves three distinct aspects: a *problem definition* from uncertain facts into a coherent statement; the *creative process* of devising a proposed solution; and the *analytical process* of determining whether the proposed solution is correct and rational. These aspects are illustrated in Figure 1.1. Here it shows that the design process (shown in dotted lines) begins with expectations of social needs and goals. These goals are converted into engineering specifications, such as design parameters (DPs), functional requirements (FRs) and so on. Once the need is formalized, ideas and methods are generated to create a prototype. This prototype is then analyzed and compared with the original set of functional requirements. When the prototype does not fully satisfy the specified requirements, either a new idea must be suggested or the functional requirements must be modified. This iterative process continues until the designer produces an acceptable product.

1.1.1 DPs and FRs

During the design process, the important step is to generate specifications for an artifact. Two specifications to represent an artifact are commonly used. They are design parameter (DP) and functional requirement (FR).

A design parameter is a variable to be determined in the beginning of the design process. These parameters are independent of each other. Normally they are a minimum set to present the system.

A functional requirement is another form of the statement to describe a design task. It refers to goals them that the final product must satisfy.

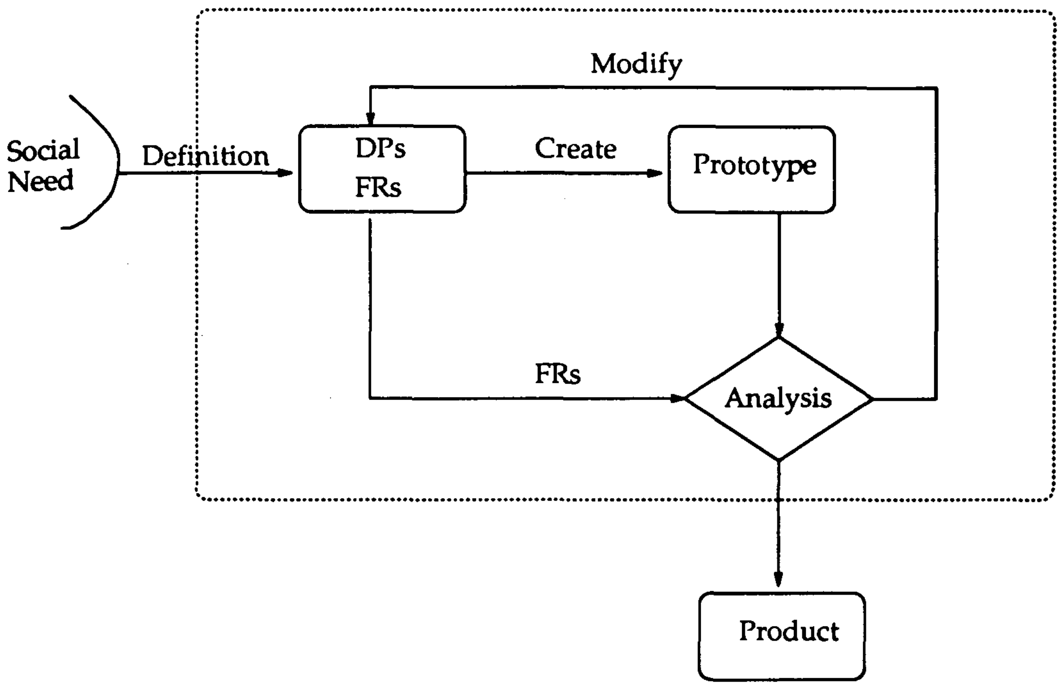

Figure 1.1: Design Process

In heat exchanger design, for example, the functional requirement is "transfer heat from one fluid stream to another". This is a formal statement of the social need. The designer may choose a shell-and-tube heat exchanger to meet the requirement. By making this choice, he introduces the design parameters "number-of-tube", "tube-length", etc. These design parameters are not given by the social need. They are a minimum set to represent the shell-and-tube heat exchanger.

1.1.2 Hierarchy of FRs and DPs

Every design has a hierarchical nature. That is, the design problem can be decomposed into new, small problems including information. The hierarchical decomposition makes the design process possible since it would otherwise be too complicated. The hierarchical nature of design can be concluded from two facts [19]:

1. FRs and DPs have hierarchies, and they can be decomposed;

2. FRs at the ith level can be decomposed into the next level of the FR hierarchy by first going over to the physical domain and developing a solution that meets the ith level FRs.

We provide personnel transportation as a top-level functional requirement. The motorized vehicle is one top-level solution. Decomposing the requirement into more detailed sub-goals, we can provide protection from weather, supply motive power, provide storage for luggage. Choosing solutions to each sub-goal, we can select either a gasoline engine or a diesel to supply motive power. We can continue this process until we get the final solution. This hierarchy is shown in Figure 1.2.

1.1.3 Creative and Analytic Design

The creative design process is a process of translating FRs into a design solution. It requires synthesis of new ideas and methods, and may involve untried combinations of them. The creative ideas and synthesis process depend on the specific knowledge possessed by the designer, and on his ability to integrate knowledge.

The creative process in design is complemented by the analytical process, which is a verification with design principle. In this process, design decisions are made by evaluating proposed prototype. It implies making correct decisions as well as evaluating the details of specific design features.

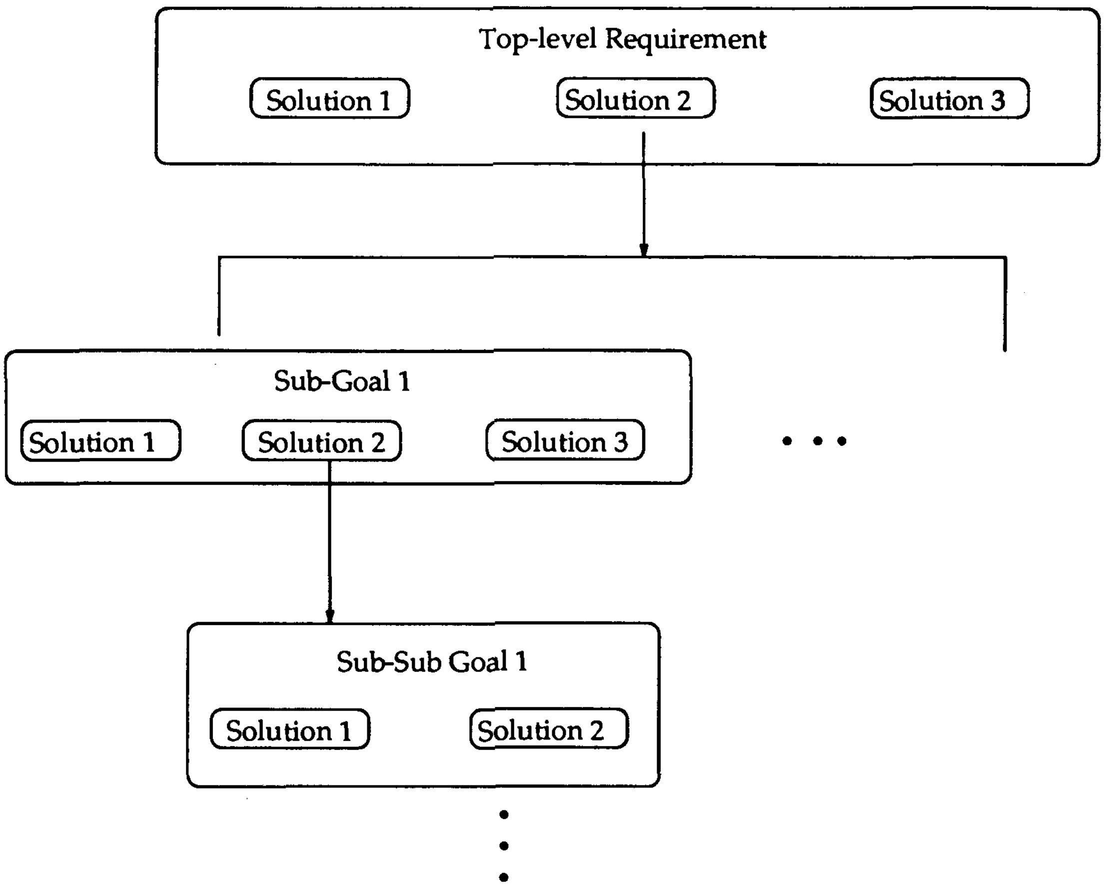

Figure 1.2: Hierarchical Structure of the functional requirement

These two processes are interrelated, since one must be able to abandon or discard bad ideas by exploring different possibilities. The creative process must be checked through analysis and corrected for differences between the perceived problem definition and the proposed solution, according to design principle.

1.1.4 Design Axioms

The design axioms provide basic standards or principles for analysis and decision making. They guide the creative process to identify good designs from a infinite number of plausible designs. Hence, they offer a basis for comparing and selecting designs. According to Suh, there are two design axioms [19] that govern a good design. The declarative form of the axioms is:

1. Axiom 1: The Independence Axiom
   Maintain the independence of the FR's

2. Axiom 2: The Information Axiom
   Minimize the information content of the design

Axiom 1 deals with the relationship between functions and physical variables. We can say that in an acceptable design, the DPs and the FRs are related so that any specific DP can be adjusted to satisfy its corresponding FR without affecting other functional requirements, while Axiom 2 deals with the complexity of a design. It states that among all the designs satisfying the Independence Axiom (Axiom 1), the one with minimum information is the best design.

Design axioms are not only a criterion for evaluating decision but also can be thought of as an objective function that directs the generation of design alternatives. They can greatly simplify the design process by eliminating many alternatives at an early stage of the process that ultimately could prove to be unsatisfactory. They establish a scientific foundation for the design field.

To conclude, the engineer's job is to design an objective solution to a problem. This process may be motivated to identify the need for a product. The need must be synthesized into a small group of requirements and solved by a mathematical or some other method. The prototype should follow design axioms in order to be tested against a feasibility analysis to confirm or disqualify the product design.

1.2 Intelligent Design

Engineering practice and scientific enquiry are undergoing a profound change, due to the advent of the computer. Computers have a significant impact in the design field. Since the mid-1980s, artificial intelligence (AI) techniques have begun to be used in design.

AI is the study of mental faculties using computational models. It can embody the knowledge of experts, who engage in thinking alternating with calculations. It is the study of ideas that may enable computers to be intelligent. A great deal of human thinking and experimentation has been devoted to the use of AI techniques.

Intelligence is the capability to acquire knowledge, the faculty of thought and reason, and the ability to perceive and manipulate things. It is an amalgam of information representation and information processing talents. Intelligent behaviour is characterized by a collection of general strategies that use knowledge. Intelligent design [3] is intelligent behaviour in design and can be described as a knowledge-based reasoning task [6]. It is composed from suitable elements represented in a knowledge base, subject to rules expressing the characteristics of the elements. Using general strategies, it exploits qualitative heuristic knowledge about the physical domain.

1.2.1 Fuzzy Logic in Intelligent Design

Advances in science and technology have made our modern society very complex and our design process increasely uncertain. As the complexity of a system increases, our ability to make precise statements about its behaviour diminishes. To simplify this complexity, an option is to increase the amount of allowable uncertainty by sacrificing some of the precise information. Rarely is a problem initially presented to an engineer in exam-paper precision of definition. Some words, such as "quite", "cheap", "very tall", and so on, are merely subjective and vague as design requirements until they have been translated into figures [9, 25]. As human beings, we must learn to accept that uncertainty is part of our life and will continue to be part of it in the future as well. The human being possesses some special characteristics that make it possible to learn and reason in a vague and fuzzy environment. He can arrive at decisions based on imprecise, qualitative data, in contrast to formal mathematics and formal logic demanding precise and quantitative data. Most human thought is approximate in nature, while the conventional approaches to knowledge representation lack the means for representing the meaning of everyday type facts exemplified by

1. "Usually" it takes "about" "an hour" to drive from SFU to Downtown Vancouver in "heavy" traffic

2. Unemployment is not "likely" to undergo a "sharp" decline during the next "few" months

3. "Most" experts believe that the likehood of a "severe" earthquake in the "near" future is "very" low

where words in the quotation involve uncertainty.

The qualitativeness of decisions implies that intelligence merely obtains an approximate solution from uncertain physical domain. Qualitative and approximate reasoning, which is close to human being's thinking, can produce intelligent design.

Fuzzy logic, as its name suggests, is one logic underlying approximate reasoning. It is much closer to human reasoning and commonsense. Using fuzzy logic, design can simulate human reasoning and cognition. It can develop novel methods for systems that are too complex for analysis by conventional quantitative techniques. It can describe different aspects of imprecision, inexactness and uncertainty of the real world, which are difficult, sometimes impossible to describe in terms of traditional methods. It can provide a direct representation for knowledge, allowing closer and closer approximations to reality.

Fuzzy logic uses fuzzy set theory as a basis for reasoning with imprecise concepts. In following chapters, we will give more detailed information about fuzzy theory and its application.

CHAPTER 2

BACKGROUND OF FUZZY THEORY

2.1 Ambiguity and Uncertainty

The sciences construct exact mathematical models of empirical phenomena and use these models to make predictions. Most aspects of the real world escape such precise mathematical models, and usually there is an elusive inexactness as part of the original phenomena. For example, if we say "tall person", we cannot always clearly determine how high it is. The ambiguity of "tall person" arises when we try to turn a qualitative statement into a quantitative range. In engineering, the adjectives that describe the states and conditions of various things are almost always connected to amounts in this way. Fuzzy mathematical properties provide a practical guide for this kind of the model.

Essentially, fuzziness is a type of imprecision that is found in such expressions as "quite easy". Such expressions correspond to fuzzy sets that do not have sharply defined boundaries [30]. These sets arise whenever we attempt to describe ambiguity, vagueness, and ambivalence in mathematical models of empirical phenomena. As one of its aims, fuzzy set theory develops a methodology for the formulation and solution of problems that are too complex or too ill-defined to analyze by conventional techniques.

In this chapter, we first introduce fuzzy set concepts and properties. Then we present fuzzy numbers and an algorithm for calculating with them. Finally, we study the theory of fuzzy logic.

2.2 Fuzzy Sets

The theory of fuzzy sets deals with subsets (represented as $\tilde{A}$) of a universe of discourse, X. The transition between full membership (1) and non-membership (0) in a fuzzy set is gradual rather than abrupt, as shown in Figure 2.1 (a), where μ represents the degree of membership in $\tilde{A}$ and x is an element in the X. By contract, the crisp set in Figure 2.1 (b) has sharp bounds at 2 and 10. The fuzzy subset has no well-defined boundaries, although the universe of discourse, X, covers a defined range of objects. This gradual boundary dividing membership in the class from nonmembership allows us to represent vagueness.

A fuzzy set can be defined mathematically by assigning a membership value to each possible individual in the universe of discourse, X. This value represents the individual's degree of membership in the set, ranging from 0 to 1:

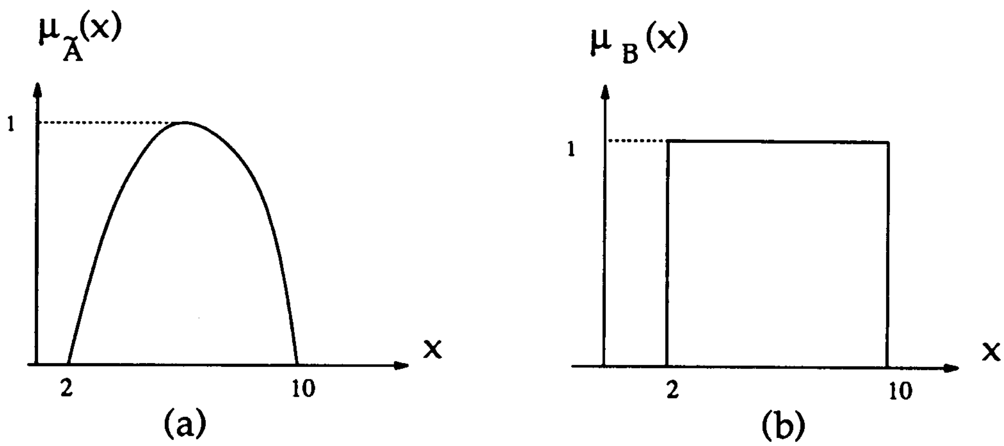

Figure 2.1: Fuzzy Set (a) and Crisp Set (b)

$$\mu_{\tilde{A}} \in [0,1]$$

If this universal set is countable, it can be defined by listing each member and its degree of membership in the set Ã

$$\tilde{A} = \sum_{i=1}^{n} \frac{\mu_{\tilde{A}(x_i)}}{x_i}$$

Similarly, when X is continuous, then, a fuzzy set Ã can be defined in the form

$$\tilde{A} = \int_{x \in X} \frac{\mu_{\tilde{A}}(x)}{x}$$

Note that in the above definitions, "–" does not refer to a division and is used as a notation to separate the element from its degree of the membership.

For instance, suppose the universe of discourse, B, is a set of people's ages , and Ã is a set of the ages of "young" people in a town. Then B can be represented as

$$B = \{b_1, b_2, b_3, b_4, b_5, b_6, b_7, b_8, b_9, b_{10}\}$$
$$= \{1, 5, 10, 20, 30, 40, 50, 60, 70, 80\}$$

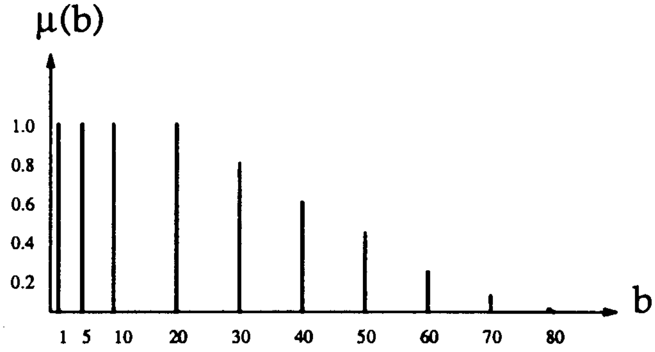

Figure 2.2: Distribution of "young" People

and Ã, the fuzzy set shown in Figure 2.2, is represented as

$$\tilde{A} = \{\frac{1.}{1}, \frac{1.}{5}, \frac{1.}{10}, \frac{1.}{20}, \frac{0.8}{30}, \frac{0.5}{40}, \frac{0.3}{50}, \frac{0.2}{60}, \frac{0.1}{70}, \frac{0.}{80}\}$$

In the set of "young" people, the age of 30, element b_5, has membership value of 0.8; the age of 50, element b_7, has a membership value of 0.3, and so on.

In general, we distinguish three kinds of inexactness: generality, that a concept applies to a variety of situations and occurs when the universe is not just one point; ambiguity, that it describes more than one distinguishable situation and occurs when there is more than one local maximum of a membership function; and vagueness, that precise boundaries are not defined and occurs when membership function takes values rather than just 0 and 1. We mainly discuss the vagueness.

2.3 Properties of Fuzzy Sets

Four important properties of fuzzy set theory are the resolution principle, normality, convexity and the extension principle. But in order to discuss these, we must introduce the concept of α-cuts [32].

2.3.1 α-cuts

When we want to exhibit an element $x \in X$ belonging to a fuzzy set $\tilde{A}$, we may demand that its membership value be greater than some threshold $\alpha \in [0, 1]$, which leads to a crisp set. This crisp set of elements in the universe X is called the α-cut, A_α, of $\tilde{A}$. Each element of A_α has a membership value in $\tilde{A}$ greater than or equal to the specified value of α, where

$$A_\alpha = \{x \in X|_{\mu_{\tilde{A}}(x) \geq \alpha}\} \tag{2.1}$$

The subscript α standing for α-cuts can be chosen arbitrarily in the interval [0, 1]. This value is often designated at some value of membership function appearing in the fuzzy set. For instance, the α-cuts equal to 0.2, 0.8 and 1.0, respectively, of the fuzzy set "young" in Figure 2.2 can be written as

$$\text{"young}_{0.2}\text{"} = \{1, 5, 10, 20, 30, 40, 50, 60\}$$
$$\text{"young}_{0.8}\text{"} = \{1, 5, 10, 20, 30\}$$
$$\text{"young}_{1.0}\text{"} = \{1, 5, 10, 20\}$$

A fuzzy set $\tilde{A}$ may be constructed from its α-cuts through the resolution principle:

$$\tilde{A} = \bigcup_\alpha \alpha\, A_\alpha$$

Where $\alpha\, A_\alpha$ is the product of a scalar α with set A_α. In other words, a fuzzy set $\tilde{A}$ is decomposed into $\alpha\, A_{\tilde{\alpha}}$, $\alpha \in [0, 1]$ and is expressed as the union of these. It comes out as shown in Figure 2.3. As can be seen from the figure, if $\alpha_2 < \alpha_1$, $A_{\alpha_2} \supset A_{\alpha_1}$.

Given sets such as A_{α_1} and A_{α_2}, we can retrieve the original membership function of fuzzy set Ã by connecting their membership functions. Therefore, a fuzzy set can be expressed in terms of the concept of α-cuts without resorting to the membership function. It is very convenient for the calculation of operations on fuzzy sets.

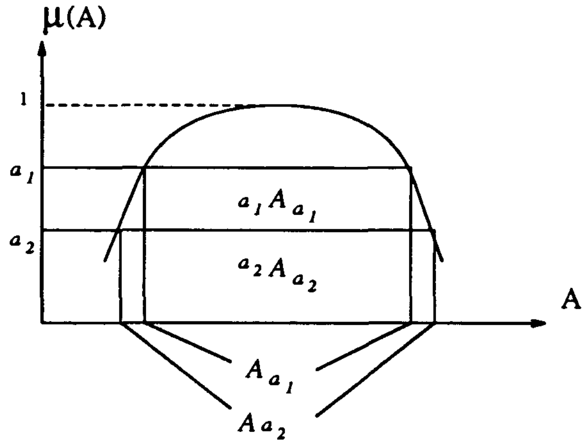

Figure 2.3: Decomposition of a Fuzzy Set

2.3.2 Normal and Convex Fuzzy Sets

We now define normality and convexity. A fuzzy set Ã is normal if and only if

$$\bigvee_{x} \mu_{\tilde{A}(x)} = 1 \tag{2.2}$$

for all $x \in$ X, where $\bigvee$ stands for maximum. This means that the highest membership value of $\mu_{\tilde{A}(x)}$ equals 1.

A fuzzy set is convex if and only if each of its α-cuts is a convex set. That is: Ã is convex if

$$\mu_{\tilde{A}}(\gamma x_1 + (1-\gamma)x_2) \geq \min[\mu_{\tilde{A}}(x_1), \mu_{\tilde{A}}(x_2)], \tag{2.3}$$

for all $x_1, x_2 \in X$ and $\gamma \in [0,1]$.

2.3.3 Extension Principle

One of the basic principles of fuzzy set theory is the extension principle [5]. It extends calculation in crisp sets to fuzzy sets. This allows the fuzzy domain of a function to be extended to the conventional field. Suppose that we have an n-ary function, f, which is a mapping from the Cartesian product $X_1 \cdot X_2 \cdots X_n$ to a universe Y such that $y = f(x_1, x_2, \cdots, x_n)$. And $\tilde{A}_1, \tilde{A}_2, \cdots, \tilde{A}_n$, which are n fuzzy sets in $X_1, X_2, \cdots, X_n$, respectively, are characterized by a set of membership functions $\mu_i(x_i)$, $i = 1, 2, \cdots, n$. Then, the extension principle allows us to induce from n-ary function, f, a fuzzy set, $\tilde{A}$, on Y such that

$$\mu(y) = \bigvee_{y=f(x_1,\cdots,x_n)} \Big(\bigwedge_{i\in[1,n]} \mu(x_i) \Big) \tag{2.4}$$

where the $\vee$ stands for the maximum and $\wedge$ minimum.

2.4 Fuzzy Numbers

If the universe of discourse is the real number, a normal and convex fuzzy set is called a fuzzy number [10, 12, 13]. Fuzzy numbers have membership functions like the ones in Figure 2.4 such as $\tilde{5}$ in this figure (a) and $\tilde{8}$ in the figure (b). By α-cuts, a fuzzy number is changed to a set of closed intervals with lower bound, a_1, and upper bounds, a_2. These values greater than or equal to a_1 and smaller than or equal to a_2 can be expressed as the symbol

$$A_\alpha = [a_1^\alpha, a_2^\alpha]$$

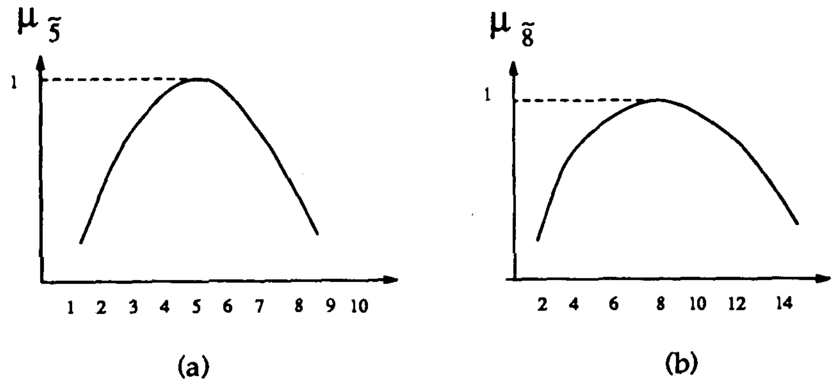

Figure 2.4: Fuzzy Numbers

We approximate the fuzzy number by a series of α-cuts. By the extension principle, we can define addition, subtraction, multiplication, division, minimum and maximum on the fuzzy number. Using these definitions, we get the solution for each α-cut. Then by the resolution principle, we reassemble the α-cuts to get the solution as a fuzzy number.

2.4.1 Addition

Let $\tilde{A}$ and $\tilde{B}$ be two fuzzy numbers and A_α and B_α their intervals for an α-cut, $\alpha \in [0, 1]$. We can write the addition operation for the α-cut as

$$C_\alpha = A_\alpha + B_\alpha = [a_1^\alpha, a_2^\alpha] + [b_1^\alpha, b_2^\alpha] = [a_1^\alpha + b_1^\alpha, a_2^\alpha + b_2^\alpha] \tag{2.5}$$

The addition operation can be expressed as

$$\tilde{C} = \tilde{A} + \tilde{B}$$

where

$$\tilde{C} = \bigcup_\alpha \alpha\, C_\alpha$$

2.4.2 Subtraction

Considering the following definitions and symbols, for all $\alpha \in [0,\ 1]$, subtraction can be defined by

$$C_\alpha = A_\alpha - B_\alpha = [a_1^\alpha, a_2^\alpha] - [b_1^\alpha, b_2^\alpha] = [a_1^\alpha - b_2^\alpha, a_2^\alpha - b_1^\alpha] \tag{2.6}$$

The method for addition can also be extended to subtraction. Subtraction is, in fact, the addition of the image of $\tilde{B}^-$ to $\tilde{A}$, where

$$B_\alpha^- = [-b_2^\alpha, -b_1^\alpha]$$

for all $\alpha \in [0,\ 1]$.

2.4.3 Multiplication

At this point we consider multiplication in R^+. Let us consider two fuzzy numbers Ã and B̃ in R^+. From the α-cuts, we can write

$$C_\alpha = A_\alpha \cdot B_\alpha = [a_1^\alpha, a_2^\alpha] \cdot [b_1^\alpha, b_2^\alpha] = [a_1^\alpha \cdot b_1^\alpha, a_2^\alpha \cdot b_2^\alpha] \tag{2.7}$$

for all $\alpha \in [0,\ 1]$.

2.4.4 Division

•
Division of two fuzzy numbers is defined in R^+ by

$$C_\alpha = A_\alpha : B_\alpha = [a_1^\alpha, a_2^\alpha] : [b_1^\alpha, b_2^\alpha] = [\frac{a_1^\alpha}{b_2^\alpha}, \frac{a_2^\alpha}{b_1^\alpha}] \tag{2.8}$$

Division is a multiplication by the inverse; that is by

$$B_\alpha^{-1} = [\tfrac{1}{b_2^\alpha}, \tfrac{1}{b_1^\alpha}]$$

for all $\alpha \in [0, 1]$.

2.4.5 Minimum and Maximum

Consider two fuzzy numbers Ã and B̃,

$$A_\alpha = [a_1^\alpha, a_2^\alpha], \qquad B_\alpha = [b_1^\alpha, b_2^\alpha],$$

$\forall \alpha \in [0, 1]$, we define the fuzzy minimum of Ã and B̃ as

$$A_\alpha \wedge B_\alpha = [a_1^\alpha, a_2^\alpha] \wedge [b_1^\alpha, b_2^\alpha] = [a_1^\alpha \wedge b_1^\alpha, a_2^\alpha \wedge b_2^\alpha] \tag{2.9}$$

$$\tilde{A} \wedge \tilde{B} = \bigcup_\alpha \alpha(A_\alpha \wedge B_\alpha)$$

and the fuzzy maximum of Ã and B̃ as

$$A_\alpha \vee B_\alpha = [a_1^\alpha, a_2^\alpha] \vee [b_1^\alpha, b_2^\alpha] = [a_1^\alpha \vee b_1^\alpha, a_2^\alpha \vee b_2^\alpha] \tag{2.10}$$

$$\tilde{A} \vee \tilde{B} = \bigcup_\alpha \alpha(A_\alpha \vee B_\alpha)$$

The symbols $\wedge$ and $\vee$ will be used for representing the minimum and maximum of the fuzzy numbers Ã and B̃.

2.5 Common Types of Fuzzy Numbers

2.5.1 Triangular Fuzzy Numbers

Triangular fuzzy numbers (TFN's) shown in Figure 2.5 are a particular type of fuzzy numbers. It can be represented as:

$$
\begin{array}{ll}
\mu_{\tilde{A}} = 0 & x < a_1 \\
\mu_{\tilde{A}} = \frac{x-a_1}{a_2-a_1} & a_1 \le x \le a_2 \\
\mu_{\tilde{A}} = \frac{a_3-x}{a_3-a_2} & a_2 \le x \le a_3 \\
\mu_{\tilde{A}} = 0 & x > a_3
\end{array}
$$

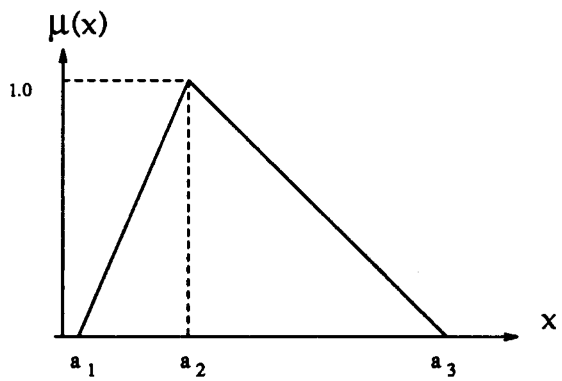

Figure 2.5: The Shape of TFN

Assume that a_1, a_2 and a_3 are finite. A TFN is completely represented by a triplet as

$$\tilde{A} = (a_1, a_2, a_3)$$

At the α-cut, the interval is given by

$$A_\alpha = [A_1, A_2] = [a_1 + \alpha(a_2 - a_1), a_3 - \alpha(a_3 - a_2)]$$

2.5.2 Trapezoidal Fuzzy Numbers

Trapezoidal fuzzy numbers (TRFN) shown in Figure 2.6 are another special case of a fuzzy number.

$$\begin{array}{ll} \mu_{\tilde{A}} = 0 & x < a_1 \\ \mu_{\tilde{A}} = \frac{x-a_1}{a_2-a_1} & a_1 \leq x < a_2 \\ \mu_{\tilde{A}} = 1 & a_2 \leq x \leq a_3 \\ \mu_{\tilde{A}} = \frac{a_4-x}{a_4-a_3} & a_3 < x \leq a_4 \\ \mu_{\tilde{A}} = 0 & x > a_4 \end{array}$$

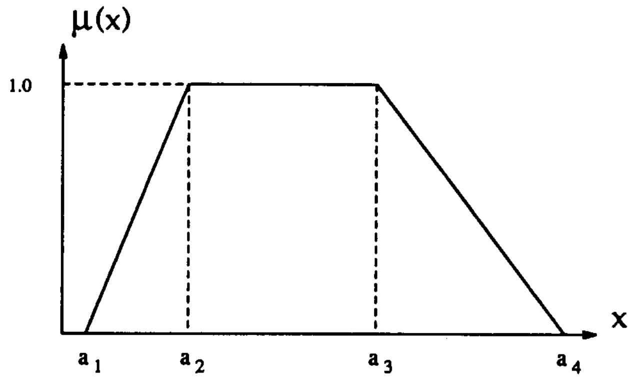

Figure 2.6: The Shape of TRFN

Assume that a_1, a_2, a_3 and a_4 are finite. A trapezoidal fuzzy number is completely represented by four elements as

$$\tilde{A} = (a_1, a_2, a_3, a_4)$$

At the α-cut, the interval is given by

$$A_\alpha = [A_1, A_2] = [a_1 + \alpha(a_2 - a_1), a_4 - \alpha(a_4 - a_3)]$$

2.6 Linguistic Variables

The core of fuzzy set theory models an imprecise situation. However, the theory can be difficult to use directly without linguistic variables. The linguistic variables [18] built on fuzzy set theory offer imprecise statements like "low", "somewhat low", "very low" and "fairly low". They allow a natural specification of values for imprecise concepts and can be used as a quantitative expression of the statement.

A linguistic variable is a variable whose values are natural language expressions referring to some quantity of interest. These natural language expressions are names for the fuzzy sets. These fuzzy sets are composed of the possible numerical values that the quantity of interest can assume. An example of a linguistic variable demonstrates its structure. Let us define a linguistic variables as "NUMBER". The quantity of interest "NUMBER" will be an integer between 1 and 10. The sets of natural expressions of "NUMBER" are "few", "several" and "many". These in turn are the following fuzzy sets:

$$\begin{aligned}
\text{"NUMBER"} &= \{\text{"few"}, \text{"several"}, \text{"many"}\} \\
\text{"few"} &= \{\tfrac{0.4}{1}, \tfrac{0.8}{2}, \tfrac{1.0}{3}, \tfrac{0.4}{4}, \tfrac{0}{5}, \tfrac{0}{6}, \tfrac{0}{7}, \tfrac{0}{8}, \tfrac{0}{9}, \tfrac{0}{10}\} \\
\text{"several"} &= \{\tfrac{0}{1}, \tfrac{0}{2}, \tfrac{0.5}{3}, \tfrac{0.8}{4}, \tfrac{1.0}{5}, \tfrac{1.0}{6}, \tfrac{0.8}{7}, \tfrac{0.5}{8}, \tfrac{0}{9}, \tfrac{0}{10}\} \\
\text{"many"} &= \{\tfrac{0}{1}, \tfrac{0}{2}, \tfrac{0}{3}, \tfrac{0}{4}, \tfrac{0}{5}, \tfrac{0.4}{6}, \tfrac{0.6}{7}, \tfrac{0.8}{8}, \tfrac{0.9}{9}, \tfrac{1.0}{10}\}
\end{aligned}$$

where the numerator of each fraction is the degree of membership and the denominator an element in the set. A linguistic variable NUMBER has three values, "few", 'several" and "many", which are natural language expressions and represented by their fuzzy sets.

A linguistic variable differs from a numerical variable in that its values are not numbers but words or sentences in a natural or artificial language. Since words, usually, are less precise than numbers, the concept of a linguistic variable can provide a means to characterize phenomena which are too complex or too ill-defined. Since these are represented in fuzzy logic by fuzzy numbers, linguistic variables can be manipulated using the operations of fuzzy algorithm [20].

CHAPTER 3

POLE COMPUTATION ALGORITHM (PCA)

3.1 Prior Work

The application of fuzzy logic to engineering design described in this thesis requires the frequent calculation of functions of fuzzy numbers. Algebraic operations on real numbers can be extended to fuzzy numbers by means of the extension principle of Zadeh. Therefore, fuzzy numbers can be processed similar to the non-fuzzy case, and the operations are sometimes called the extended operations (extended addition, extended subtraction, etc.). Although the solution of these operations is defined by the extension principle of Eq(2.4), the implementation of the solution procedure is not trivial. The reason is that the solution procedure corresponds to a nonlinear programming problem which is very complex except for the simplest mapping function

[2].

Prior methods for calculating the results of mappings on fuzzy numbers can be classified according to whether they seek approximate or exact solutions. Examples of the class of approximate methods include the numerical procedure suggested by Schmucker [18], the analytic method of Dubois and Prade [8] and the method of Dong and Wong [7]. In [7], Dong and Wong show that Schmucker's discretization method can give quite irregular and incorrect membership functions. Dubois and Prade's method requires that the function f be increasing over the solution space; this is equivalent to requiring that the solution space be uniform as introduced below. An example of an exact solution method is the non-linear programming technique of Baas and Kwakernaak [2]. The applicability of this method depends on certain restrictive conditions, given as Lemma 1 in [2]. These conditions are equivalent to the requirement that the solution space be uniform or quasi-uniform [29].

In this chapter, we describe an original approach for calculating the result of applying any function to fuzzy numbers [10, 12, 13]. A classification scheme for functions on fuzzy numbers is presented, based on the behaviour of the partial derivatives of the function. It will be shown that there is a class of problems - these having interior extrema - for which no existing method can give reliable results. A generally applicable alternative is suggested. We will also show that there is a class of problems - those having uniform solution spaces - for which the output may be calculated very rapidly, and a calculation method is given. Finally, for the class of problems having quasi-uniform solution spaces, we recommend the method of Wong and Dong. Minor modifications to their method can take advantage of special features of the problem to reduce computational complexity.

3.2 FWA Algorithm

The FWA [7] algorithm is a way of computing the output of a function supplied with fuzzy inputs, according to Zadeh's extension principle. The method is based on the α-cut representation of fuzzy sets and interval analysis. This method provides a discrete solution to extended weighted averages.

For N real fuzzy numbers, $\tilde{U}_1, \cdots, \tilde{U}_N$, let u_i be an element of $\tilde{U}_i$, $(i \in [1, N])$. Given a performance parameter represented by the expression $z = f(u_1, \cdots, u_N)$, let $\tilde{Z}$ be the fuzzy output of the expression. The following steps lead to the solution of $\tilde{Z}$.

1. For each $\tilde{U}_i$, discretize the preference function into a number of α values, $\alpha_1, \cdots, \alpha_M$, where M is the number of steps in the discretization.
2. Determine the intervals $[u_i^L, u_i^R]$ for each parameter $\tilde{U}_i$, i = 1, $\cdots$, N, for each α-cut, α_j, j = 1, $\cdots$, M.
3. Using one end point from each of the N intervals for each α_j, combine the end points into an N-ary array, $[u_1^{P_1}, \cdots, u_i^{P_i}, \cdots, u_N^{P_N}]$, where $\forall$ i, $P_i \in \{L, R\}$. A total of 2^N distinct permutations exist for the array.
4. For each of the 2^N permutations, determine $z_k = f(u_1^{P_1}, \cdots, u_N^{P_N})$, k = 1, $\cdots$, 2^N.
5. Give Z_j=$[\bigvee(z_k), \bigwedge(z_k)]$ as the resultant interval for the α-cut, α_j.
6. Repeat step 3 through 5 for other α-cuts, then apply the resolution principle to obtain $\tilde{Z}$.

For N fuzzy inputs and M discrete points on the preference function, the algorithmic complexity of the FWA implementation is on the order of

$$H \sim M \cdot 2^N$$

where H equals the number of operations.

The most serious drawback of the FWA algorithm arises when f is non-monotonic in one or more of the fuzzy variables $\tilde{A}_1, \cdots, \tilde{A}_n$. A second drawback of the FWA is the necessity of performing 2^N calculations for every value of α. This becomes computationally expensive when N is large, as will often be the case in engineering applications.

3.3 Definitions and Notations

Let X be a Cartesian product of universes: $X = X_1 \times \cdots \times X_n$ and let $\tilde{A}_1, \cdots, \tilde{A}_n$ be fuzzy sets in $X_1, \cdots, X_n$, respectively. Let f be a bounded, continuously differentiable function from X to a universe Y. The fuzzy set $\tilde{B}$ induced on Y by applying f to the sets $\tilde{A}_i$ is calculated. If we write $y = f(x_1, \cdots, x_n)$ where $x_i \in \tilde{A}_i$ and $y \in \tilde{B}$, then the set $\tilde{B}$ is defined by the extension principle in Eq(2.4). We know that if the $\tilde{A}_i$ are normal and convex, and f is bounded, then $\tilde{B}$ is also normal and convex. A series of α-cuts of fuzzy sets is considered. For each choice of α, there are n intervals $[x_1^L, x_1^R]$, $\cdots$, $[x_n^L, x_n^R]$, with Cartesian product A_α. Calculating the corresponding interval $[y^L, y^R]$, $B_\alpha = f(A_{1\alpha}, \cdots, A_{n\alpha}) = [y^L, y^R]$ can be obtained. The y^L and y^R correspond respectively to the global minimum and maximum of f over the space A_α. So the problem here is to locate these minimum and maximum values. In broad outline, we do this by identifying a short list of candidates, called poles, for these values, then selecting the highest and lowest values from this list.

The algorithm described below does not depend on any continuous function associated with the variables used. We will be working with general sets and membership functions. We introduce some definitions and notations first in order to make the description easier.

3.3.1 Representation of a point

Consider the n-dimensional space $A_{1\alpha} \times \cdots \times A_{n\alpha}$. A point in this space can be written as $X = (x_1, \cdots, x_n)$. For any given point X, three sets are defined: the left set, L, containing those values of i for which $x_i = x_i^L$; the right set, R, containing those values of i for which $x_i = x_i^R$; and the interior set, M, containing the remaining values of i. Then any point is described in the space as a triple (x_l, x_r, x_m), where x_l represents the set of variables x_i for which $i \in L$, and similarly for x_r and x_m.

3.3.2 Weighting Function

Consider the partial derivatives of f with respect to each of the variables x_i

$$\phi_i(X) = \frac{\partial f(X)}{\partial x_i}$$

A weighting function θ is defined as follows:

$$\theta(x_i) = \begin{cases} 1 & \text{if } x_i = x_i^L; \\ -1 & \text{if } x_i = x_i^R; \\ 0 & \text{otherwise.} \end{cases}$$

The weighted partial derivative of the function f at a point X is then written as:

$$\psi_i(X) = \theta(x_i)\,\phi_i(X)$$

3.3.3 Poles

A point $X_p = (x_l, x_m, x_r)$ is a first class of the pole if

1. $\psi_l(X_p) > 0 \qquad \forall l \in L;$
2. $\psi_r(X_p) > 0 \qquad \forall r \in R;$
3. $\phi_m(X_p) = 0 \qquad \forall m \in M;$
4. $\frac{\partial \phi_m}{\partial x_m} = \phi'_m(X_p) > 0 \qquad m \in M$, (if $\phi'_m = 0$, then $\phi''_m > 0$, etc.)

and a second class of the pole, if

1. $\psi_l(X_p) < 0 \qquad \forall l \in L;$
2. $\psi_r(X_p) < 0 \qquad \forall r \in R;$
3. $\phi_m(X_p) = 0 \qquad \forall m \in M;$
4. $\frac{\partial \phi_m}{\partial x_m} = \phi'_m(X_p) < 0 \qquad m \in M$, (if $\phi'_m = 0$, then $\phi''_m < 0$, etc.)

Both of them are called poles for short.

3.3.4 Particular Zone

A particular zone, Z_i, of A_α is an $(n-1)$-dimensional subspace of A_α for which $\phi_i = 0$, where the i identifies each particular zone with the corresponding axis x_i of the solution space. If A_α has no particular zones, it is said to be a uniform solution space. If A_α has one or more particular zones, and none of these zones Z_j intersects the corresponding axis, x_j, or any line parallel to this axis, then A_α is said to be a quasi-uniform solution space. Otherwise, A_α is said to be a non-uniform solution space.

3.3.5 Normal Pole and Particular Pole

If a point X_p is both a pole and a corner of the solution space A_α, then X_p is said to be a normal pole. If X_p is a pole and is located at an intersection between m particular zones and $n-m$ boundaries of A_α, $1 \leq m \leq n$, then X_p is a particular pole.

3.4 Solution Space

In the solution space, any extremum of $f(x)$ must be either a normal or a particular pole. Here, we first discuss the method for two special cases: uniform and quasi-uniform solution space. In the next section, we develop a general method for computation.

3.4.1 Uniform Solution Space

In a uniform solution space A_α over the given interval $[x_1^L, x_1^R], \cdots, [x_n^L, x_n^R]$, with $\phi_i > 0$, $1 \leq$ i $\leq$ (k-1) for some integer k, $2 \leq$ k $\leq$ n and $\phi_i(x) < 0$, $k \leq i \leq n$, there are exactly two poles,

$$X^L = (x_1^L, \cdots, x_{k-1}^L, x_k^R, \cdots, x_n^R) \tag{3.1}$$

$$X^R = (x_1^R, \cdots, x_{k-1}^R, x_k^L, \cdots, x_n^L) \tag{3.2}$$

Both poles are at corners of A_α; the minimum pole, X^L, is a global minimum, the maximum, X^R, is a global maximum. Hence $y^L = f(X^L)$ and $y^R = f(X^R)$ [1].

Example 3-1: Consider the function $y = f(x_1, x_2) = x_1\, x_2$, $x_1 \in A_{1\alpha} = [1, 5]$, $x_2 \in A_{2\alpha} = [-4, -3]$. We note that $\phi_1 = x_2 < 0$ and $\phi_2 = x_1 > 0$, so $A_\alpha = A_{1\alpha} \times A_{2\alpha}$ is uniform. Therefore, there are no particular poles and only two normal poles. These two poles are shown in Figure 3.1. The arrows on the boundaries of the solution space in the figure denote the directions in which the ϕ_i are increasing. The dots represent poles, where P_1 is the lowest position for the point of (x_1, x_2) and P_2 is the highest position for the point of (x_1, x_2). From the Equation (3.1) and (3.2), the two poles can easily be got,

$$P_1 = (x_1^R, x_2^L) = (5, -4)$$

$$P_2 = (x_1^L, x_2^R) = (1, -3)$$

Hence $y^L = f(P_1) = 5 \cdot (-4) = -20$ and $y^R = f(P_2) = 1 \cdot (-3) = -3$

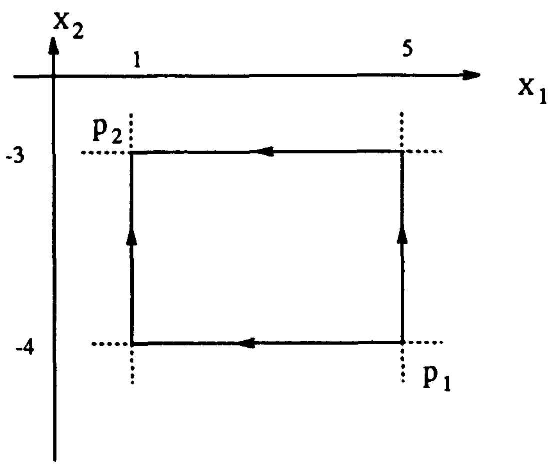

Figure 3.1: Normal Poles in Uniform Space

3.4.2 Quasi-Uniform Solution Space

In a quasi-uniform solution space, there are only normal poles. So, the result can be obtained through comparing these normal pole's values

Example 3-2: Consider the function $y = f(x_1, x_2) = x_1 x_2$, $x_1 \in A_{1\alpha} =$ [-1, 5] and $x_2 \in A_{2\alpha} =$ [-4, 3].

$$\phi_1 \begin{cases} = 0 & \text{if } x_2 = 0; \\ > 0 & \text{if } x_2 > 0; \\ < 0 & \text{if } x_2 < 0. \end{cases}$$

and

$$\phi_2 \begin{cases} = 0 & \text{if } x_1 = 0; \\ > 0 & \text{if } x_1 > 0; \\ < 0 & \text{if } x_1 < 0. \end{cases}$$

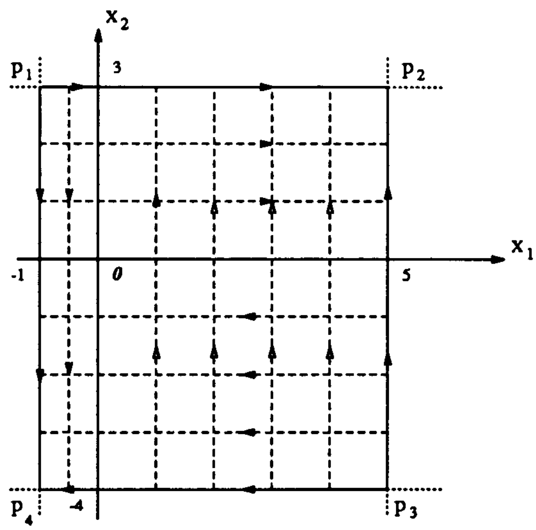

Figure 3.2: Normal Poles in Quasi-Uniform Space

From Figure 3.2, we note that $\phi_1 = x_2$, which is zero when $x_2 = 0$, and $\phi_2 = x_1$, which is zero when $x_1 = 0$, neither of which intersects with its axis or any line parallel to that axis. Therefore, $A_\alpha = A_{1\alpha} \times A_{2\alpha}$ is quasi-uniform. Then we need to examine the weighted partial derivatives at each vertex to find normal poles. In this example, the partial derivatives are calculated at each corner of the space and tabulated in the Table 3.1. There are four normal poles, P_1, P_2, P_3, and P_4, which are shown in Figure 3.2. The arrows in the figure are for the position trend. Comparing the values of the function at these four poles gives us: y^L = f(P_3) = -20 and y^R = f(P_2) = 15

Table 3.1: Finding Normal Poles

x_1	x_2	$\phi_1(x)$	$\theta_1(x)$	$\psi_1(x)$	$\phi_2(x)$	$\theta_2(x)$	$\psi_2(x)$	Pole
-1	-4	< 0	1	< 0	< 0	1	< 0	Yes
5	-4	< 0	-1	> 0	> 0	1	> 0	Yes
-1	3	> 0	1	> 0	< 0	-1	> 0	Yes
5	3	> 0	-1	< 0	> 0	-1	< 0	Yes

3.5 Pole Computation Algorithm

In the last section, we discussed methods for uniform and quasi-uniform solution spaces. But the non-uniform solution space is left. For this problem, the function may have extrema at points other than the corners of the space, ie., particular poles. Neither of the methods defined so far can locate these extrema. Hence, neither can have a correct result. Here we introduce a general method for solution of any type of bounded function.

Because they are located at either corners or intersections, all the poles, not only normal poles but also particular poles, can be found. The function is evaluated in two classes of poles, where the first class includes the poles, $P_1, \cdots, P_k$, and the second class the poles, $P_{k+1}, \cdots, P_n$. After comparison, the global minimum and maximum values, y^L and y^R, can be obtained.

$$y^L = \bigwedge_{i=1}^{k} f(P_i) \tag{3.3}$$

$$y^R = \bigvee_{i=k+1}^{n} f(P_i) \tag{3.4}$$

Example 3-3: Consider the function $f(x_1,\ x_2,\ x_3) = x_1^2 + x_2^2 + x_3^2 + x_1 x_2 + x_2 x_3$, where $x_1 \in$ [-5, 1], $x_2 \in$ [-4, 2] and $x_3 \in$ [6, 8]. We record expressions for the derivatives of the function with respect to each variable:

$$\phi_{x_1} = 2x_1 + x_2$$

$$\phi_{x_2} = 2x_2 + x_1 + x_3$$

$$\phi_{x_3} = 2x_3 + x_2$$

This is a non-uniform solution space, where the particular zone $\phi_{x_1} = 0$ intersects with the surfaces x_1 = -1, x_2 = 2, x_3 = 6 and x_3 = 8. These two points (-1, 2, 6) and (-1, 2, 8) are intersections. Similarly, we can find another three points(-5, -0.5, 6), (-5, -1.5, 8) and (1, -3.5, 6) from ϕ_{x_2} and ϕ_{x_3}. Besides, there are eight corners in this example. We form a table for these points, where c_i, i =1, $\cdots$, 8 for corners, and s_j, j = 1, $\cdots$, 5 for intersections.

Table 3.2: Normal Poles and Particular Poles in Non-Uniform Space

[-5,1]	[-4,2]	[6,8]	ϕ_{x_1}	ϕ'_{x_1}	ϕ_{x_2}	ϕ'_{x_2}	ϕ_{x_3}	ϕ'_{x_3}	N.P.	P.P	$f(x)$	y^L	y^R
-5	-4	6	< 0		< 0		> 0						
-5	-4	8	< 0		< 0		< 0		Yes		93		
-5	2	6	< 0		< 0		> 0						
-5	2	2	< 0		< 0		< 0		Yes		99		99
1	-4	6	> 0		< 0		> 0						
1	-4	8	> 0		> 0		< 0						
1	2	6	< 0		< 0		> 0						
1	2	8	< 0		< 0		< 0		Yes		87		
-1	2	6	=0	> 0	< 0		> 0						
-1	2	8	=0	> 0	< 0		< 0						
-5	-0.5	6	< 0		=0	> 0	> 0						
-5	-1.5	8	< 0		=0	> 0	< 0						
1	-3.5	6	> 0		=0	> 0	> 0			Yes	24.75	24.75	

In the Table 3.2, N.P. and P.P stand for the normal poles and particular poles, respectively. From the weighted partial deviative functions, a particular pole, P_1, (1, -3.5, 6) and three normal poles, P_2 at (-5, -4, 6), P_3 at (-5, -4, 8) and P_4 at (1, 2, 8) are calculated. P_1 belongs to the first class of the pole, and its function is a global minimum value. P_2, P_3 and P_4 belong to the second class. Comparing the value of the function at these three poles, the global maximum value is attained at P_3. The result value is [24.75, 99].

3.6 Bounded Function

We want to be able to classify functions as having a uniform, quasi-uniform, or non-uniform solution space in advance of knowing their arguments. Three types of functions are listed below:

Type 1

$$y = f(x) = \sum_{i=1}^{n} c_i\, x_i$$

Type 2

$$y = f(x) = \frac{\sum_{i=1}^{n} c_i \prod_{j=1}^{i} x_{i,j}}{\sum_{l=1}^{n} d_l \prod_{m=1}^{l} z_{l,m} + k}$$

Type 3

$$y = f(x) = \frac{\sum_{i=1}^{n} w_i\, x_i}{\sum_{i=1}^{n} w_i}, \qquad \sum_{i=1}^{n} w_i \neq 0$$

Where the c_i, d_l and k are real numbers and the x_i, $x_{i,j}$, $z_{l,m}$ and w_i are members of fuzzy numbers $\tilde{X}_i$, $\tilde{Z}_i$ and $\tilde{W}_i$. Any function not falling under one of the above types can be classed as Type 4.

There are some characteristics for different types. Functions of Type 1 will always induce a uniform solution space; functions of Type 2 and 3 will induce solution spaces which are either uniform or quasi-uniform, depending on input fuzzy numbers. Type 4 functions may induce non-uniform solution spaces. Several fast methods of calculation, which together will allow us to handle all three types of bounded continuously differentiable function, are introduced.

3.6.1 Deletion Rule

Consider the type 2 functions obtained when $d_l = 0$, $k = 1$, $c_i = 1$, $i = 1, \cdots, n$:

$$y = x_1 \cdots x_n$$

Given the fuzzy numbers $\tilde{X}_1, \cdots, \tilde{X}_n$ to which the arguments belong, and given a particular value of α, the arguments can grouped into three disjoint classes. Thus it can be written

$$\begin{aligned} x_i &\in [-x_i^L, x_i^R], & \forall i \in Z \\ x_j &\in [x_j^L, x_j^R], & \forall j \in J \\ x_k &\in [-x_k^R, -x_k^L], & \forall k \in K \end{aligned}$$

where Z, J and K are indicial sets which label the variables (x_i^L, x_i^R, x_j^L, x_j^R, x_k^L and x_k^R are all positive) as belonging to "zero", "positive" and "negative" intervals respectively, these intervals corresponding to the "zero", "positive" and "negative" fuzzy numbers as defined by Mizumoto and Tanaka in [16].

It is then easy to show that the extrema of the function are given by

$$y^L = \bigwedge_{i \in Z} (-x_i^L \prod_{l=1 \neq i}^{n} x_l^R), \qquad y^R = \prod_{l}^{n} x_l^R, \quad \text{if } |K| \text{ even}$$
$$y^R = -\bigwedge_{i \in Z} (-x_i^L \prod_{l=1 \neq i}^{n} x_l^R), \quad y^L = -\prod_{l}^{n} x_l^R, \quad \text{if } |K| \text{ odd}$$

This is referred to as the Deletion Rule.

Example 3-4: Given $y = x_1 x_2 x_3 x_4 x_5$, where $x_1 \in$ [-1, 2], $x_2 \in$ [-2, 1], $x_3 \in$ [-3, -1], $x_4 \in$ [-5, -2], $x_5 \in$ [4, 6]. We note that this function belongs to special case in

Type 2, where $Z = \{1, 2\}$, $J = \{5\}$ and $K = \{3, 4\}$. K is even, so the deletion rule is applied to obtain

$$\begin{aligned} y^L &= \bigwedge_{i \in Z} (-x_i^L \prod_{l=1 \neq i}^{5} x_l^R) \\ &= \bigwedge((-1) \cdot 1 \cdot 3 \cdot 5 \cdot 6,\ (-2) \cdot 2 \cdot 3 \cdot 5 \cdot 6) = -360 \\ y^R &= \prod_{l=1}^{5} x_l^R = 2 \cdot 1 \cdot 3 \cdot 5 \cdot 6 = 180 \end{aligned}$$

3.6.2 Selection Rule

Consider a function of Type 3, which is a fuzzy weighted average:

$$y = \frac{\sum_{i=1}^{n} x_i w_i}{\sum_{i=1}^{n} w_i}$$

The derivatives of this function are

$$\begin{aligned} \phi_{x_i} &= K w_i \\ \phi_{w_i} &= K^2 \sum_{j=1}^{n} (x_i - x_j) w_j \end{aligned}$$

where

$$K = \frac{1}{\sum_{i=1}^{n} w_i}$$

We can always choose to define our weights such that K is positive. Therefore, the partial derivative, ϕ_{x_i}, will have the same sign as w_i. If the lower bound and upper bound of w_i have the same sign, (i.e., both positive or both negative), the minimum of the function will be attained when $x_i = x_i^L$ or x_i^R, immediately. If, on the other hand, the support of w_i includes zero, the sign of ϕ_{wi} must be determined first.

This observation leads to an efficient way of locating the extrema: suppose the support of w_i includes zero for $1 \leq i \leq l$, where $1 \leq l \leq n$, while for $l < i \leq k$,

$l < k \leq n$, the support of w_i is wholly positive and for the remaining values of i, the support of w_i is wholly negative. Then the extrema of the function are

$$y^L = \bigwedge_{t_i \in [0,1], 1 \leq i \leq n} \frac{Z^L(t_i) + P^L(t_i) + N^L(t_i)}{w(t_i)} \tag{3.5}$$

where

$$Z^L(t_i) = \sum_{i=1}^{l} [x_i^R w_i^L t_i + x_i^L w_i^R (1 - t_i)]$$
$$P^L(t_i) = \sum_{i=l+1}^{k} x_i^L [w_i^L t_i + w_i^R (1 - t_i)]$$
$$N^L(t_i) = \sum_{i=k+1}^{n} x_i^R [w_i^L t_i + w_i^R (1 - t_i)]$$

and

$$w(t_i) = \sum_{i=1}^{n} [w_i^L t_i + w_i^R (1 - t_i)] \tag{3.6}$$

$$y^R = \bigvee_{t_i \in [0,1], 1 \leq i \leq n} \frac{Z^R(t_i) + P^R(t_i) + N^R(t_i)}{w(t_i)} \tag{3.7}$$

where Z^R, P^R and N^R are defined exactly as Z^L, P^L and N^L, except with the superscripts of x^L and x^R interchanged. Evaluating these extrema requires a total of 2^{n+1} function evaluations, while FWA would require 2^{2n} for the same problem.

Example 3-5: Given the function

$$y = \sum_{i=1}^{3} \frac{x_i w_i}{w_i}$$

where x_1: [1, 8], x_2: [2, 9], x_3: [3, 10], w_1: [-1, 2], w_2: [-2, 3] and w_3: [4, 8]. The selection rule is applied to get

$$y^L = \bigwedge \frac{8\,(-1)\,t_1 + 1 \cdot 2\,(1-t_1) + 9\,(-2)\,t_2 + 2 \cdot 3\,(1-t_2) + 3 \cdot (4\,t_3 + 8\,(1-t_3))}{(-1)\,t_1 + 2\,(1-t_1) + (-2)\,t_2 + 3\,(1-t_2) + 4\,t_3 + 8\,(1-t_3)}$$

$$y^R = \bigvee \frac{1(-1)t_1+8\cdot2(1-t_1)+2(-2)t_2+9\cdot3(1-t_2)+9\cdot3(1-t_3)+10(4t_3+8(1-t_3))}{(-1)t_1+2(1-t_1)+(-2)t_2+3(1-t_2)+4t_3+8(1-t_3)}$$

Arrange the t_i, i =1, 2, 3, shown in Table 3.3, to obtain y^l and y^r. Minumum and maximum results for lower and upper bounds are y^L = -14 and y^R = 35, respectively.

Table 3.3: Finding Extrema Using Selection Rule

t_3	t_2	t_1	y^l	y^r	y^L	y^R
1	1	1	-14	35	-14	35
1	1	0	-1	15		
1	0	1	1.67	11		
1	0	0	2.2	10.6		
0	1	1	-1	13		
0	1	0	1	11		
0	0	1	2.22	9.22		
0	0	0	2.46	9.46		

CHAPTER 4

ONE APPROACH TO IMPRECISION IN DESIGN

4.1 Introduction

The engineering design process can be characterized as a collection of phases. Each phase can transform a description of a need, from a highly imprecise preliminary stage to a final configuration in the form of a precise, physical design description. At the beginning stage, the designer is not certain what values will be used for each design parameter, or at least, he can choose several different values for the design parameter. In this stage, the level of imprecision in the description of design parameters is typically high. As the design process proceeds, the imprecision with each known design parameter is reduced. Because of the imprecision inherent in the preliminary design stage, computational tools for this phase must be able to manipulate imprecise representations of design alternatives, while directly incorporating the engineer's judgement. In this chapter, we introduce one method for engineering design, developed by Wood and Antonsson. The next chapter will describe an alternative method.

4.2 Wood & Antonsson's Approach

In the preliminary stage, the designer determines the approximate or exact relationships governing the design configuration, and specifies the mathematical expressions of the design. He selects imprecise inputs to these expressions as fuzzy numbers. These fuzzy numbers describe the set of design parameters.

Once the input parameters and performance parameter expressions have been determined, an analytical or numerical method such as FWA is used to solve for the imprecise performance parameters. These results are also fuzzy numbers, representing the performance of the design. Then, these parameters are compared to the design criteria, and the "backward path" [24, 25, 26, 27], to be defined below, can be utilized to find out each input parameter. Peak values of inputs correspond to the peak in the output sets. Off-peak values of the output at a particular membership function level correspond to one or more off-peak inputs.

In the design, some input parameters are very strongly coupled to the outputs, and others are nearly independent. A mathematical measure (γ-level) will provide the means to determine this coupling. If the measure indicate that the input parameter contributes very little to the analysis, they may be fixed to the most desired value, resulting in a simplification of the design problem.

4.3 Interpretation of Parameters

A designer would prefer not to fix parameters in the early stages [23], but rather to evaluate imprecisely defined alternatives. Fuzzy set theory forms the basis for interpreting these input parameters in a design. We represent imprecision by a fuzzy set, a range associated with a membership function. The membership function describes the desirability of using these particular values. A most desired value or interval is assigned to every input parameter. The more confident, or the more the designer desires to use an input value, the higher its membership in the parameter's set. Specifically, the parameter's values with membership of one are those in which the designer has the highest confidence; whereas parameter values with membership of zero are those he feels sure can be excluded from consideration.

Just as the input design parameters may have fuzzy sets associated with them, the output of the design calculation also has a direct interpretation. The performance parameter will be in the form of a function, with an interval encompassing all possible output performance values, and with a preference function ranging between 0 and 1. The values of the set which takes on membership values of one can be directly traced back to the input parameter values which have a membership of one. Other output values with a membership value α which is less than one similarly correspond to at least one input parameter's having a membership value equal to α, the remaining parameters' membership being greater than α. This reasoning is called "the backward path", which is a natural consequence of calculations using the fuzzy method. If the resulting output value for the membership equal to one satisfies the functional requirement in the performance space of the design, the design may proceed without further detailed numerical analysis. However, if the functional requirement is not satisfied, the output membership function is examined to determine where the acceptable output value falls. The input parameter may then be traced for all values on the characteristic curve itself, according to the backward path.

4.4 Design Measure

The Measure of Fuzziness [24] expresses the coupling of input parameters and output parameters in the design. Here, we define a γ-level measure, D, for the engineering design as

$$D(\tilde{C}) = \sum_{i=1}^{n}(e^{\alpha \tilde{c}(x_i)} - 1) \tag{4.1}$$

where $\tilde{C}$ is a fuzzy number.

Let $\tilde{C}_1, \cdots, \tilde{C}_N$ be N imprecise inputs, and let $\tilde{D}$ be the output of the computation $y = f(x_1, \cdots, x_n)$. The γ-level value is calculated from these steps:

1. Determine $\tilde{D}$ using numerical calculation, for instance, FWA.
2. Discrete the intervals into n equally spaced steps when $\alpha_{\tilde{D}}$ equals to 0.
3. For each input parameter , $\tilde{C}_i$, $i = 1, \cdots, N$, set all other $\tilde{C}_j$, $j \neq i$, to their nominal crisp value (where $\alpha_{\tilde{C}}$ equals to 1), use FWA to calculate the output θ_i.
4. Calculate the γ-level measure for $\tilde{D}$ and all θ_i from Equation (4.1).
5. Normalize the $D(\tilde{D})$'s with respond to $D(\tilde{\theta}_i)$. The result is an ordering of the inputs according to the strength of their effect on the performance parameter.

The design measure provides the ability to find out information on the coupling between the inputs and outputs of the design. It can determine which parameters produce little or no effect on the performance and which parameters will alter the output the most. Those having little effect on the performance can be fixed on their peak values, which reduces the complexity of the deisgn.

4.5 Heat Exchanger Design Example

4.5.1 Requirement Statement

A shell-and-tube heat exchanger consists of a number of tubes supported by two end plates (shown in Figure 4.1). The outer surface of the tubes is maintained at $100^{o}C$ by condensing steam, and the tubes are made of a metal whose conductivity is high compared with the heat transfer coefficient between the inner surface of the tubes and the water flowing in them. In designing the heat exchanger, we can vary the number of tubes, n; the tube diameter, d_1; and the tube length, l. We assume the tubes are thin-walled, so the internal and external diameters are approximately the same. Our objective is to transfer about 5 kW of heat into water flowing through the tubes of the heat exchanger. The water will enter the heat exchanger at about $90^{o}C$, at a flow rate between 1.5 and 2.0 kg/s.

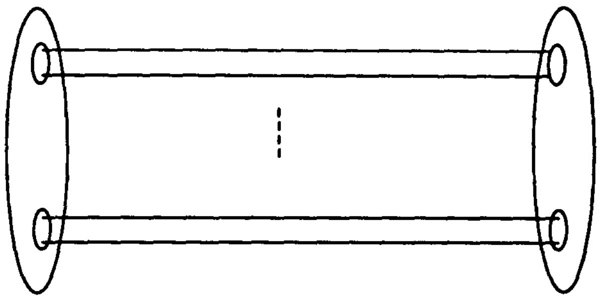

Figure 4.1: Structure of a Shell-and-Tube Heat Exchanger

4.5.2 Performance Parameter Expression (PPE)

From the structure of the heat exchanger, shown in Figure 4.1, we can choose the number of tubes, n; the tube diameter, d_1, and the tube length, l, as design parameters. Also, the temperature of water flowing through the tubes of the heat exchanger and a flow rate of water are design parameters. The heat exchanger is designed to transfer $5kW$ of water through it. According to the heat transfer theory, the PPE [5] can be expressed as

$$q = h \cdot a \cdot (t - t_1) \tag{4.2}$$

where t is the temperature of outer surface of tubes, t_1 temperature of water, h convective heat transfer coefficient, a area of the surface of tubes. To completely describe this problem, the expressions for h, a, and relative equations, such as Nusselt number Nu, Reynold's number Re, and a flow rate $\dot{m}$, are needed as follows:

$$h = \frac{Nu \cdot k}{d_1} \tag{4.3}$$

$$Nu = 0.022 P_R^{0.6} \cdot R_e^{0.8} \tag{4.4}$$

$$Re = \frac{u_{av} \cdot d_1}{\nu_e} \tag{4.5}$$

$$\dot{m} = \pi \cdot \rho \cdot (d_1/2)^2 \cdot u_{av} \cdot n \tag{4.6}$$

$$a = \pi \cdot d_1 \cdot n \cdot l \tag{4.7}$$

where k is the thermal conductivity of the material, P_r Prandtl number, u_{av} the flow rate of water, ν_e kinematic viscosity and ρ fluid density.

4.5.3 Functional Requirement (FR)

There is a requirement in this heat exchanger. The heat transferring through the tubes must be between the minimum value q_{r_1} and the maximum value q_{r_2}. Thus, the functional requirement for the heat exchanger in this example is:

$$q_{r_1} = 4.5$$

$$q_{r_2} = 5.5$$

$$q_{r_1} \leq q \leq q_{r_2}$$

where the subscripts r_1 and r_2 denote "requirement".

4.5.4 Design Parameters (DPs)

The designer specifies the input parameters as preference functions. These parameters in the heat exchanger are the temperature of entering water, t_1; the number of tubes, n; the tube diameter, d_1; the tube length, l; and the mass flow rate of water, $\dot{m}$. Note that the flow rate of water u_{av} in Equation (4.5) and (4.6), is not an independent DP, since it depends directly upon the mass flow rate of water $\dot{m}$.

In the design, the subjective knowledge, experience, and desires of the engineer are used to imprecisely determine these input parameters. For example, the tube length,

l, is constrained by a maximum length proposed in the configuration of Figure 4.2 (a). The input parameter l is imprecisely defined in a range of possible values where the desirability increases from the minimum preference value ($\alpha = 0$) in the range to the maximum preference value ($\alpha = 1$) and decreases to the minimum preference value ($\alpha = 0$) shown in Figure 4.2 (a). In this case, the length 10 cm, which corresponds to the peak of the fuzzy number, is the most desired, while values of the length greater than or less than 10 cm are less desirable. Therefore, for support values to the right of the input fuzzy number's peak or left, the degree of the membership function is specified to have lower values. Design lengths of the minimum performance, in this particular example are 1 cm and 100 cm, respectively. We can apply α-cuts to this parameter, for example, the interval [1, 100] under $\alpha = 0$, to calculate its performance.

The diagrams shown in Figure 4.2 (b), (c), (d) and (e) are the remaining design parameters: temperature, t_1; the number of tubes, n; flow rate of water, $\dot{m}$ and the tube diameter, d_1, respectively. Triangular fuzzy numbers are chosen to represent these input parameters. So, these fuzzy design parameters can be represented by three values: left-extreme value, the peak value and right-extreme value. The formulae below give numerical forms of each design parameter, according to the order of Figure 4.2.

$$l = \begin{cases} 0.01 + 0.09\, y_l \\ 1 - 0.9\, y_l \end{cases}$$

$$t_1 = \begin{cases} 70 + 20\, y_{t_1} \\ 100 - 10\, y_{t_1} \end{cases}$$

$$n = \begin{cases} 1 + 9\, y_n \\ 100 - 90\, y_n \end{cases}$$

$$\dot{m} = \begin{cases} 1.5 + 0.3\, y_{\dot{m}} \\ 2 - 0.2\, y_{\dot{m}} \end{cases}$$

$$d_1 = \begin{cases} 0.001 + 0.049\, y_{d_1} \\ 0.050 \end{cases}$$

where y_l, y_{t_1}, y_n, $y_{\dot{m}}$ and y_{d_1} are the degree of membership functions between zero and one of l, t_1, n, $\dot{m}$ and d_1, respectively. Table 4.1 provides the properties for this example, which are the thermal conductivity of the material, k; fluid density, ρ; kinematic viscosity, ν_e; Prandtl number, P_R and temperature of outer surface, t. Combining these inputs and Equation (4.2–4.7), the performance value can be calculated.

Table 4.1: Constant Data in the Heat Exchanger

name	constant
$k(W/m^2.^oc)$	0.668
$\rho(kg/m^3)$	974.08
$\nu_e(m^2/s)$	0.364×10^{-6}
P_R	2.22
$t(^oC)$	100

4.5.5 Performance Parameter

With the input parameters specified in the last section and Equations (4.2–4.7), the preference function can be determined by the FWA algorithm. After the calculations have been performed, the fuzzy output for the rate of heat transfer to the water q, is shown in Figure 4.3. This figure shows the imprecise performance parameter result for the heat exchanger.

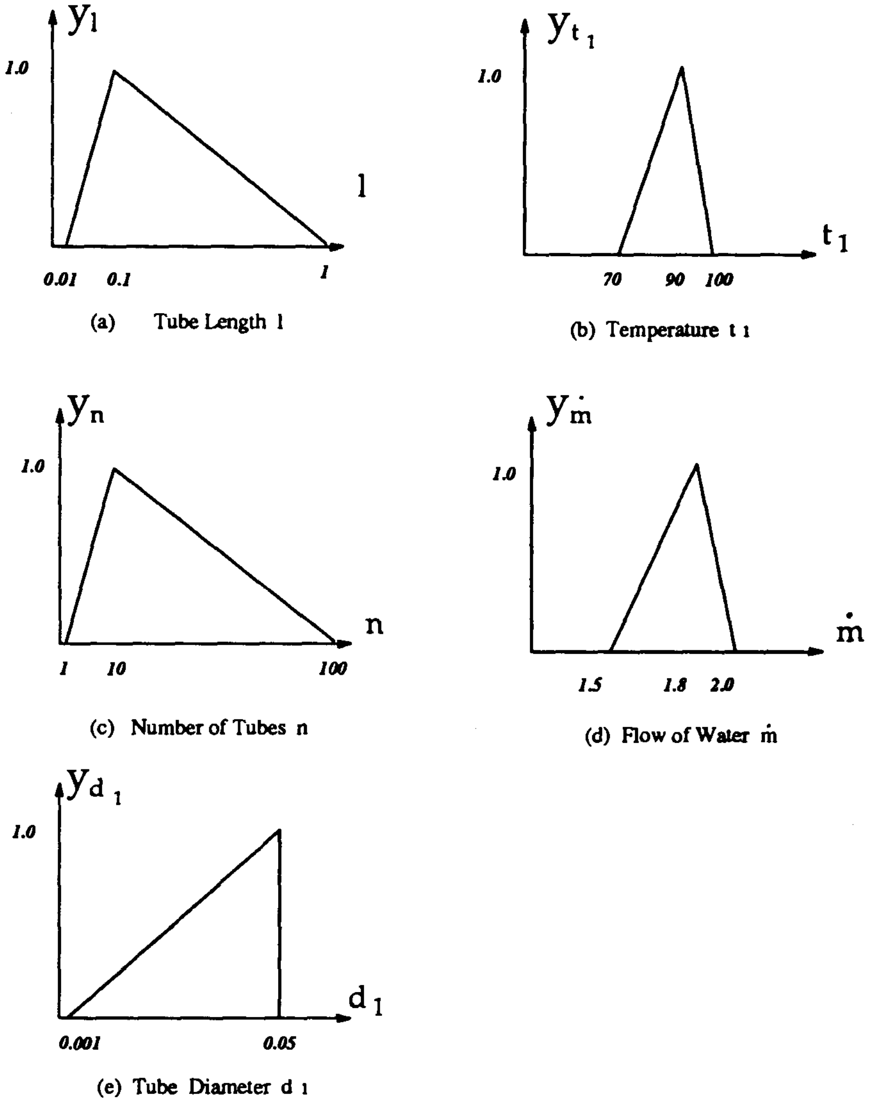

Figure 4.2: Design Parameters for a Heat Exchanger

The next step is to compare the output set with the performance criterion. Comparing the output q with the requirement values q_{r_1} and q_{r_2}, the output value at the peak of q under $\alpha = 1$ is not between q_{r_1} and q_{r_2}. Thus the output of the heat for most confident inputs does not satisfy the FR, while some values to the right of the peak in Figure 4.3 are found to satisfy the FR, in fact, for q between 6.3 kW under $\alpha = 0.4$ and 4.4 kW under $\alpha = 0.3$. To respond to the FRs, the input parameters must deviate from the peak values. According to the γ-level measure shown in Table 4.2, the values of the tube length, l, and diameter, d_1, have strongest coupling with the output. These two design parameters are changed to the values 0.41 m and 0.04 m, respectively, while other DPs are very little effect on their output and are set to their peak memberships. Thus, the heat transfer rate $q = 5.25$ kW can be obtained. This value corresponds to the most desirable set of values for the DP's and falls in the functional requirement.

Table 4.2: γ-Level for Heat Transfer Rate q

name	γ-level
t	0.01
l	0.17
n	0.01
m	0.01
d_1	0.2

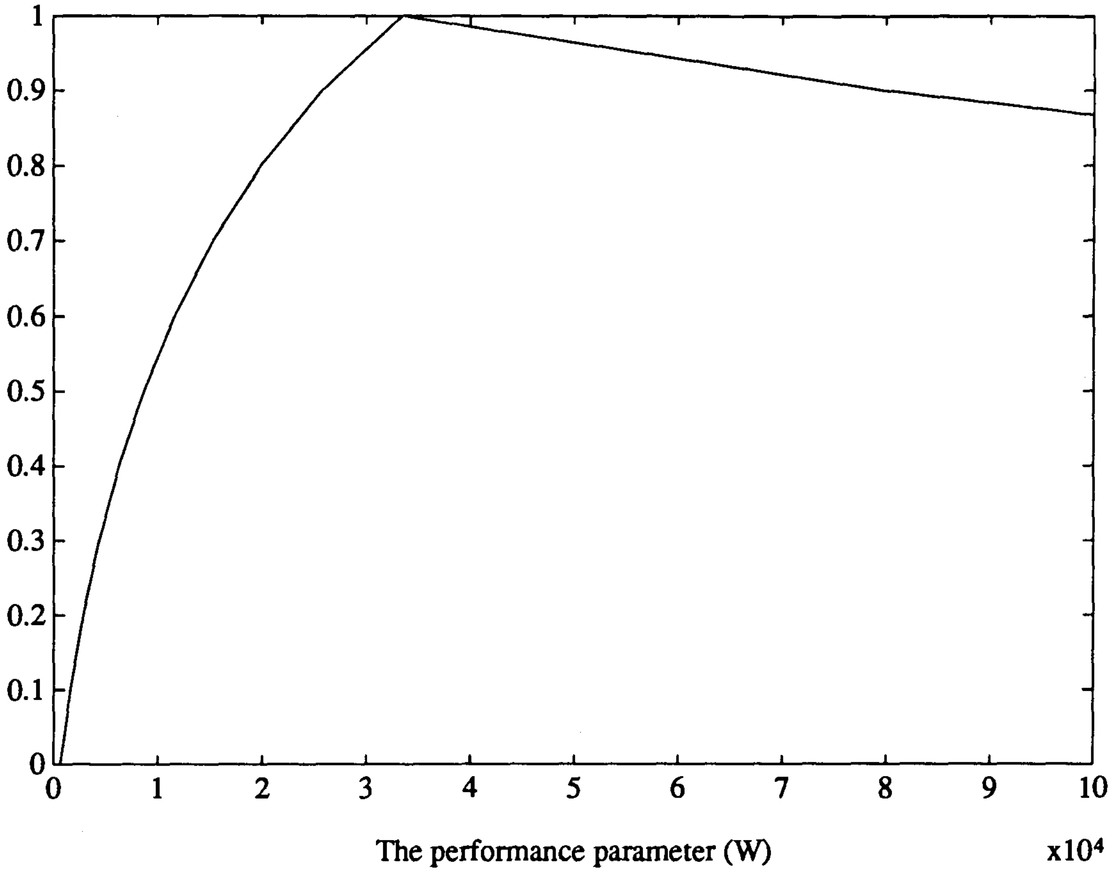

Figure 4.3: Performance Parameter for the Heat Exchanger

CHAPTER 5

IMPLEMENTING FUZZY DESIGN

5.1 Introduction

While making use of Wood & Antonsson's approach, we found that their method has some shortcomings. Firstly, it appears to us difficult to attach rational preferences to the values of isolated design parameters; for example, given a choice among a range of steel alloys, should we prefer the cheaper alloys or the stronger ones? We can see no sensible way of answering this question before seeing the impact on the final design. Secondly, there are no standards for how we can get the design parameters and performance goals. It is not clear whether the preference function for the DP denotes the designer's *desire* to use a particular values, or his *expectation* that the value will lie in a certain range. Thirdly, they have not used inexact, linguistic terms to describe a library of components. Therefore the designer will have to select ranges for the DP for each new design problem.

In this chapter, we develop an alternative method to manipulate the engineering design: introduce a vocabulary of component types, associating membership functions with design parameters and building knowledge bases; suggest a metric for trade-off strategies between the multiple performance parameters; and use algorithms described in chapter 3 for the fuzzy calculation.

5.2 A Sense of Scale

In developing a knowledge base to support engineering design, we face a trade-off between efficiency and generality. It becomes impossible sometimes to identify a sharp threshold between impossibility and possibility. Here is an example:

"Can you see a rubber band being used to power an aircraft weighting 10 gms?"
"Yes, certainly."
"And if the aircraft weights 1 kg?"
"Perhaps; I don't think that would be the best way."
"How about a 747?"
"That's ridiculous!"

There is a maximum size beyond which rubber-powered flight becomes impractical. Yet it would be very difficult to say what this size is. There might be no shorter way of proving it impractical than by examining a variety of attempts to reach the goal. It is unlikely to identify a sharp threshold between impossibility and possibility. As we lower the power requirement, we eventually come to a rubber-power airplane that can be built.

A sense of scale classifies experience into qualitatively different regimes. Within a regime, some effects are important, other can be ignored. Making distinctions of this sort is one of the respects in which an engineer differs from a physicist. This is illustrated by the engineer's use of non-dimensional numbers. In the Navier-Stokes equations, physics provides a complete and accurate description of fluid flow. But for design work, engineers rely on empirical correlations between the Reynolds number, the friction factor and Nusselt number–correlations which are only approximately true, but which provide a basis for making decisions.

Originally, our sense of the scale comes from contingent details of our lives. Our bodies are a particular size, and the tasks important to us require particular amounts of effect. By contrast, scientific laws represent an attempt to describe the universe independent of these contingent details. It would be surprising, then, if our judgements of scale could be deduced from the laws of physics.

When working in the neigbourhood of the threshold between possibility and impossibility, we would expect to take a long time to decide that the requirements were unachievable based on considering very unusual designs. But most design decisions are not like this. Most of the time we are far from the threshold, and can say very quickly whether or not a proposal is feasible. This is the characteristic we should like to see in a fuzzy intelligent design system.

5.3 Design Match

Trade-off strategies are always present in the design process. We must balance different aspects. We can compare the fuzzy numbers representing the predicted and desired performance for a given component or artifact, to implement an overall approach of trade-off.

Let the performance variable have predicted value (X, μ) where X is the range of possible values and μ the preference function over that range, and let the desired performance for the same variable be (Y, ν). Then match between the prediction and the requirement is a real number d in the range between zero and one, given as follows:

$$d = f_d(X, Y, \mu, \nu)$$

where

$$f_d = \max_{x \in X \cap Y} \{\min(\mu(x), \nu(x))\}$$

The diagram shown in Figure 5.1 illustrates the predicted value, X, the desired value, Y, and their relation, f_d. Since f_d is 0.6 in this figure, the match between X and Y is 0.6. The formulae of design match, f_d, can be chosen in order to match the design closely.

The match has the following characteristics:

1. Given a second artifact with performance (X, μ') and $\mu' \geq \mu$, $\forall x \in X \cap Y$, f_d(X, Y, μ', ν) $\geq f_d$(X, Y, μ, ν).

2. Given a second requirement with performance (Y, ν') and $\nu' \geq \nu$, $\forall y \in X \cap Y$, $f_d(\mathrm{X}, \mathrm{Y}, \mu, \nu') \geq f_d(\mathrm{X}, \mathrm{Y}, \mu, \nu)$

3. If X ∩ Y = 0, $f_d(\mathrm{X}, \mathrm{Y}, \mu, \nu) = 0$.

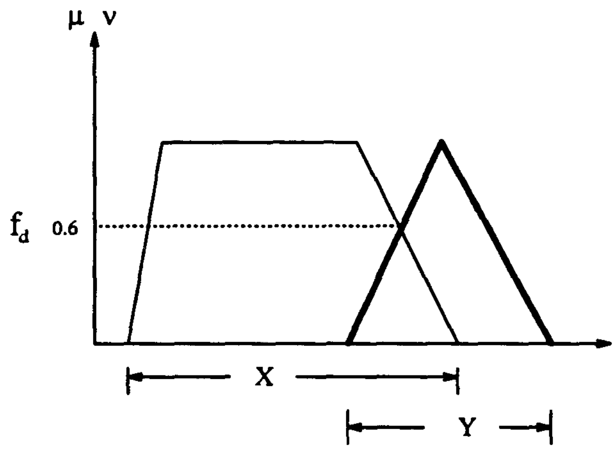

Figure 5.1: Performance Parameter Values

In general, we have a number of requirements. We can combine the matches for each requirement to get an overall measure. Let us denote this combined metric by $f_D = f_D(d_1, \cdots, d_n)$. The operation f_D should have the following characteristics:

1. $f_D(d_1, \cdots, d_n) = 0$, if $\forall\ i$: $d_i = 0$, $i \in [1, n]$.

2. $f_D(d_1, \cdots, d_n) = d$, if $d_1 = \cdots = d_n = d$.

3. $f_D(d_1, \cdots, d_i, \cdots, d_n) \leq f_D(d_1, \cdots, d_i', \cdots, d_n)$ iff $d_i \leq d_i'$

One definition of f_D that would satisfy these characters is

$$f_D(d_1, \cdots, d_n) = \min_{1 \leq i \leq n} d_i$$

Other definitions of f_D are possible. The choice of f_D corresponds to the choice of a particular design strategy.

5.4 Building a Vocabulary of Components

To develop a fuzzy knowledge base to support a preliminary design, we create and maintain a set of components. These are mechanisms, devices and artifacts qualitatively different from one another. These are characterized by a number of parameters described by linguistic variables. A shell-and-tube heat exchanger, for example, can be classified as a "small", "medium" or "big" heat exchanger, whose parameters can be represented by fuzzy numbers. The preference functions for the diameters of these shell-and-tube heat exchangers are illustrated in Figure 5.2. The membership functions defined for these parameters represent the imprecision in the definition of these shell-and-tube heat exchangers. It is clear that there is no precise upper bound to the diameter of the exchanger, yet it is also clear that a tube having diameter 1 μm is not a heat exchanger.

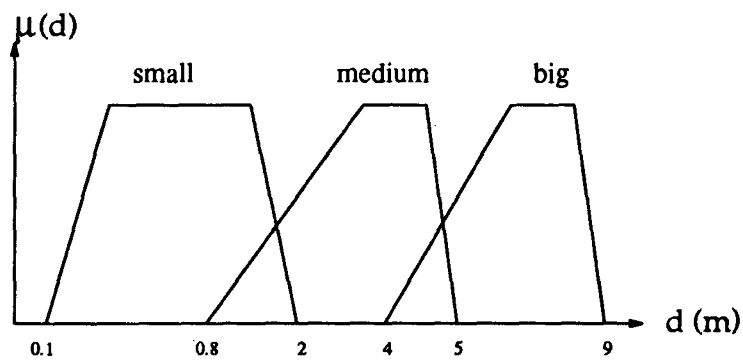

Figure 5.2: Diameters for Shell-and-Tube Heat Exchangers

When the knowledge base is used for design, the user will supply a number of design requirements. These will also be naturally represented as linguistic variables. An engine designer, for example, is unlikely to be asked for an engine with a power output of 500 kW. More probably the design requirement will be for "a sporty four-cylinder engine" or perhaps an engine with output "about 480 kW". "about 480 kW" is a fuzzy number, with gradual bounds. There will typically be a number of requirements on a design. Some may be expressed as fuzzy targets for performance, others as fuzzy upper or lower bounds on variables.

There is a difference in interpretation of membership functions between design parameters and performance requirements. In the former case, the degree of membership function reflects our estimate of how well a given value falls within the meaning of an imprecisely defined word in the vocabulary; in the latter, the degree of membership function represents the desirability of a performance parameter achieving a given value.

5.5 Parameters & Boundary Conditions

The knowledge base is built up through device modeling, guided by an expert in the field. We put imprecise bounds on the geometric parameters of a device. From these we deduce the corresponding limits on its performance. The performance of a device is a function of its parameters and of its boundary conditions.

A parameter is "internal" to the device. It describes the geometry and material properties of a device. For example, the parameters for a shell-and-tube heat exchanger would include the number of tubes, the tube length and diameter, and the tube material.

The boundary conditions are imposed by the context in which the device is used.

It may be a consequence of choices made in the design of neighboring components, or may be given explicitly or implicitly in the design requirements. For example, the boundary conditions for a shell-and-tube heat exchanger could include the volumetric flow rates of the two fluid streams and the physical properties of the two fluids.

5.6 Refining the Design

In the first stage of the design process, the designer faces a choice between a number of different mechanisms to meet the given requirements. For example, we might want to choose a small tubular, a medium tubular, a large finned or a extra large plain heat exchanger to meet the criterion. An initial selection can be made by using the match f_d and f_D to compare the imprecise performance with the given requirements for each alternative.

This early stage of design may serve to rule out some candidate solutions, but the imprecise performance parameters of most candidates will have a broad plateau. This is what we would expect, reflecting the fact that many alternatives can be pursued to meet the requirements. To proceed with the design, it is necessary to refine the representation.

There are several hierarchies in the design. We can distinguish different categories of potential design components, corresponding to the technical vocabulary of the field – for example, we might have a fuzzy model for a shell-and-tube heat exchanger, a finned heat exchanger and a plain heat exchanger. The first refinement occurs as the designer selects one of these categories. Each design choice reduces the imprecision in the predicted performance. At the next level of refinement, we can distinguish different classes within the chosen category, such as a "small" heat exchanger, a "medium" heat exchanger, a "large" heat exchanger and so on. These distinctions can be made by using linguistic variables. The value of the imprecise representation at this stage is that it allows the infeasibility of a particular design choice to be discovered at an early stage, before the design has been elaborated in full detail. After the results from the second level of refinement, the design process proceeds to the last level and gets a final value for each design parameter.

5.7 Example

A design is required for the hot-end heat exchanger in a Stirling engine. Here are three candidate designs: shell-and-tube, finned, and plain. We have some requirements for this design:

1. The internal volume of the heat exchanger should be as small as possible, and certainly less than 300 cubic centimeters.

2. The heat flow per unit degree temperature difference through the heat exchanger walls should be about 50 Watts per degree; lower than 10 Watts per degree is definitely not good enough, and greater than 500 Watts per degree would be overdesigned.

3. The pressure drop across the exchanger should be as low as possible, and certainly less than 100 KPa.

Three stages are made for the final design: the preliminary stage, the refining stage and the precise stage. During the first stage, the preliminary stage, the designer can get very broad ranges for each parameter to decide which category can be chosen among the shell-and-tube, finned and plain heat exchangers. Then, design moves to the second stage, refining stage, to select suitable size in certain heat exchanger types, such as "VerySmall" (standing for very small size of the heat exchanger), "Small", "Medium", etc. In this stage, the designer can focus on a smaller group and obtain narrow curves. Finally going to the third stage, precise stage, the designer can get accurate results under given conditions and criterion.

5.7.1 Preliminary Stage

In the preliminary stage design, the design parameters and functional requirements are fuzzy numbers. After calculation through the mathematical model - performance parameter expressions - the results, performance parameters, will also be fuzzy numbers. We will choose one or more categories which fall in the functional requirements for the continuing design.

Performance Parameter Expressions (PPEs)

According to the theory of heat exchange, the design parameters for the heat exchangers can be chosen for the heat flow per unit degree temperature, Q, pressure drop across the exchanger, P, and the internal volume of the exchanger, V. In equation form, these PPEs for the shell-and-tube heat exchanger can be expressed as:

$$Q_{tube} = h \cdot a \tag{5.1}$$

$$P_{tube} = \frac{f \cdot D \cdot u^2 \cdot l}{2 \cdot d} \tag{5.2}$$

$$V_{tube} = \frac{\pi \cdot n \cdot d^2 \cdot l}{4} \tag{5.3}$$

where h is the convective heat transfer coefficient; a area of the surface of tubes; f friction factor; D the gas density ($4kg/m^3$); u mass flow rate; l the tube length; d the tube diameter; and n number of tubes.

In order to completely describe this problem, the expressions for h, a and relative equations, such as Nusselt number Nu, Reynold's number Re and mass flow rate, $\dot{m}$, are as follows:

$$h = \frac{N_u \cdot K}{d} \tag{5.4}$$

$$Nu = 0.022 \cdot P_r^{0.6} \cdot R_e^{0.8} \tag{5.5}$$

$$Re = \frac{u \cdot d}{V_e} \tag{5.6}$$

$$\dot{m} = \frac{\pi \cdot D \cdot d^2 \cdot u \cdot n}{4} \tag{5.7}$$

$$a = \pi \cdot d \cdot n \cdot l \tag{5.8}$$

$$f = 0.0791 \cdot Re^{-0.25} \tag{5.9}$$

where K the thermal conductivity of water ($0.169W/m^2\ ^\circ C$), P_r is the Prandtl number (2.22), V_e the viscosity of water (0.000023 m^2/s).

The plain heat exchanger can be treated as a cylinder, closed at one end. It only has two design parameters, the cylinder diameter, d, and cylinder height, l. The pressure drop of the heat exchanger is approximately equal to zero. PPEs for heat flow per unit degree temperature difference, Q, the internal volume, V, and related equations are:

$$Q_{plain} = h \cdot a \tag{5.10}$$

$$
\begin{aligned}
V_{plain} &= \frac{\pi \cdot d^2 \cdot l}{4} && (5.11)\\
a &= \pi \cdot d \cdot l && (5.12)\\
h &= \frac{N_u \cdot K}{d} && (5.13)\\
N_u &= 0.035 \cdot R_e^{0.8} && (5.14)\\
Re &= \frac{2 \cdot S \cdot D \cdot d}{C \cdot V_e} && (5.15)
\end{aligned}
$$

where C is the engine speed (50 cycles) and S the piston stroke (0.05 m).

The PPEs for the finned heat exchanger are identical to the plain heat exchanger, except that the surface area, a, is given by the expression:

$$a = 2 \cdot n \cdot l \cdot h \qquad (5.16)$$

where n is the number of fin, l their length and h their height. The rest of parameters in PPEs for the plain and finned heat exchangers are similar to the tube's. The subscript of *tube* is for the shell-and-tube heat exchanger and *plain* for both finned and plain heat exchangers.

Functional Requirements (FRs)

In this design, the heat flow per degree temperature difference, Q, the pressure drop across the exchanger, P, and the internal volume, V, must fall in certain intervals. The functional requirements for them are shown in Figure 5.3. The equation forms are:

$$Q_r = \begin{cases} 10 + 40\, y_{Q_r} \\ 500 - 450\, y_{Q_r} \end{cases}$$

$$P_r = \begin{cases} 1 \\ 100000 - 99999\, y_{P_r} \end{cases}$$

$$V_r = \begin{cases} 0.000001 \\ 0.0003 - 0.000299\, y_{V_r} \end{cases}$$

where the subscript r stands for its requirement, and the y_{Q_r}, y_{P_r}, y_{V_r} for each α-cut value.

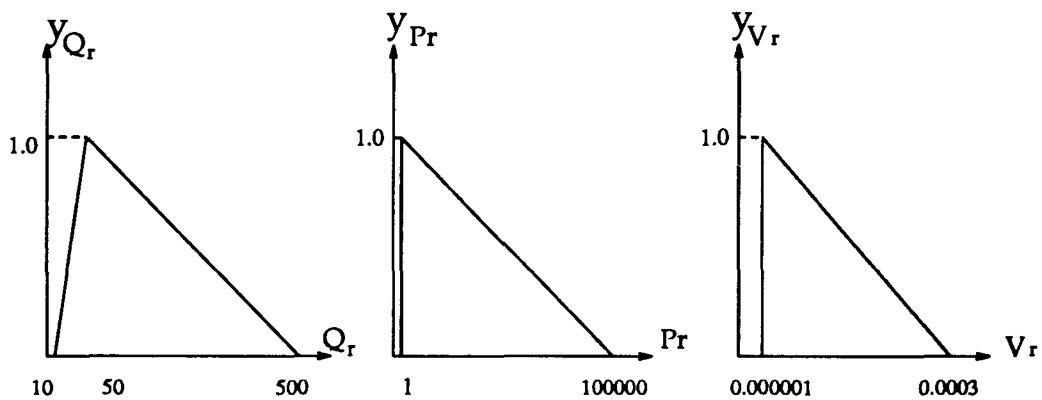

Figure 5.3: Requirements for Design

Design Parameters (DPs)

The design parameters can be specified by membership functions and represented by a trapezoidal or a triangular fuzzy number for all stages. These parameters can be selected from the knowledge base to match linguistic meaning of their terms. They are stored in the knowledge base, according to their different groups. At the first stage, wide ranges are used to assign the design parameters. These parameters in the shell-and-tube heat exchanger are the tube length, l; the tube diameter, d; the number of tubes, n; and the mass flow rate of water, $\dot{m}$. The first three variables are design parameters and the last one is a boundary condition. Note that the flow rate of water u in Equation (5.6) is not an independent boundary condition since it depends directly upon the mass flow rate of water, $\dot{m}$.

The values in the knowledge base are employed to get ranges of their fuzzy numbers representing each design parameter. Curves shown in Figure 5.4 represent preference functions for each design parameter for the shell-and-tube heat exchanger in preliminary stage. For instance, the input parameter l can take a range of possible values. Its value is constrained by a minimum and maximum length shown in Figure 5.4 (a) and a trapezoidal fuzzy number is selected. The interval [0.01, 1] corresponding to its peak value is the maximum degree of its membership function. For values to the right of its fuzzy number's peak or left, membership functions are specified to have lower values. The values under the minimum membership function are 0.001 m and 5 m, respectively. We can apply α-cuts to this parameter and get its intervals, for example, the interval [0.001, 5] corresponding to zero α-cut.

Figure 5.4 (b), (c) and (d) show the remaining design parameters: the tube diameter, d, the number of tubes, n and mass flow rate of water, $\dot{m}$, respectively. We choose trapezoidal fuzzy numbers to represent the tube diameter parameter and triangular fuzzy numbers for the last two parameters. So these parameters can be expressed by equations, according to Figure 5.4.

$$l = \begin{cases} 0.001 + 0.009\, y_l \\ 5 - 4\, y_l \end{cases}$$

$$d = \begin{cases} 0.00001 + 0.00099\, y_d \\ 5 - 4.5\, y_d \end{cases}$$

$$n = \begin{cases} 1 + 9\, y_n \\ 100 - 90\, y_n \end{cases}$$

$$\dot{m} = \begin{cases} 0.3\, y_{\dot{m}} + 1.5 \\ 2 - 0.2\, y_{\dot{m}} \end{cases}$$

where y_l, y_d, y_n and $y_{\dot{m}}$ are the degree of membership functions of l, d, n and $\dot{m}$, respectively.

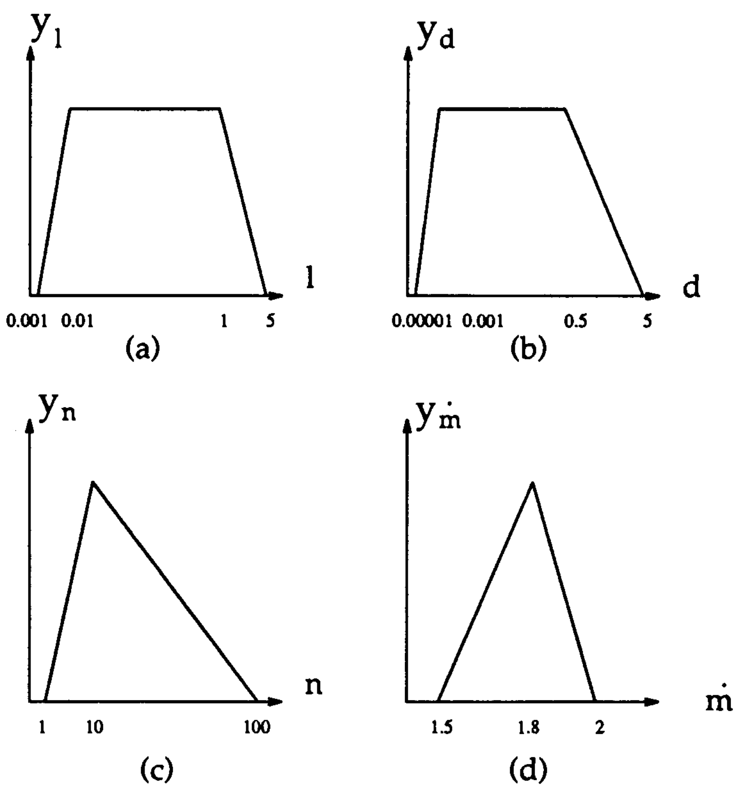

Figure 5.4: Design Parameters for a Shell-and-Tube Heat Exchanger

Since the plain heat exchanger is treated as a cylinder, closed at one end, it only has two design parameters, the cylinder diameter, d, and the cylinder height, l. The same values are taken in this example as Figure 5.5 (a) shows. Their formula follows:

$$d = \begin{cases} 0.0001 + 0.0099\, y_d \\ 0.5 - 0.45\, y_d \end{cases}$$

where y_d is the value of their membership function for the diameter and height of a cylinder.

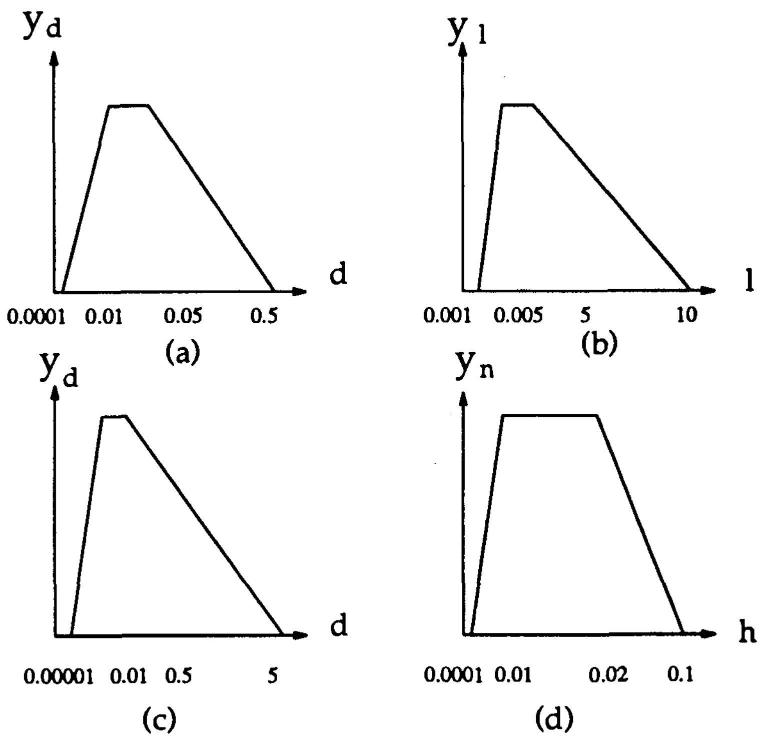

Figure 5.5: Design Parameters for Plain and Finned Heat Exchangers

The finned heat exchanger gives four parameters, number-of-fins, n; fin length, l; fin diameter, d and fin height, h. The number-of-fins has the same preference function as the number of tubes shown in Figure 5.4 (c). The rest of the parameters are in Figure 5.5 (b), (c) and (d), respectively. The forms are:

$$l = \begin{cases} 0.001 + 0.049\, y_l \\ 10 - 5\, y_l \end{cases}$$

$$d = \begin{cases} 0.00001 + 0.00999\, y_d \\ 5 - 4.5\, y_d \end{cases}$$

$$h = \begin{cases} 0.0001 + 0.0099\, y_h \\ 0.1 - 0.08\, y_h \end{cases}$$

where y_l, y_d and y_h are the degree of membership functions for the length, diameter and height of a finned heat exchanger.

Performance Parameters (PPs)

Given the input parameters, performance parameters can be determined by the PPEs and a pole computation algorithm. After the calculations have been performed, the fuzzy outputs for the heat flow per unit temperature difference, the pressure drop across the exchanger and the internal volume of each heat exchanger are obtained. Since they are very broad ranges, a logarithmic scale is used for drawing their graphs.

In Figure 5.6 (a), (b) and (c) diagrams are plotted for the performance parameters of the shell-and-tube heat exchanger. The heat flow rate per degree temperature difference of the exchanger is shown in Figure 5.6 (a). The performance value (Q_{tube}, μ) is drawn in a solid line and its requirement (Q_r, ν) in a dotted line. The requirement of the heat flow per unit degree temperature for the shell-and-tube heat exchanger is included in the heat flow per unit temperature difference output, as its graph shows.

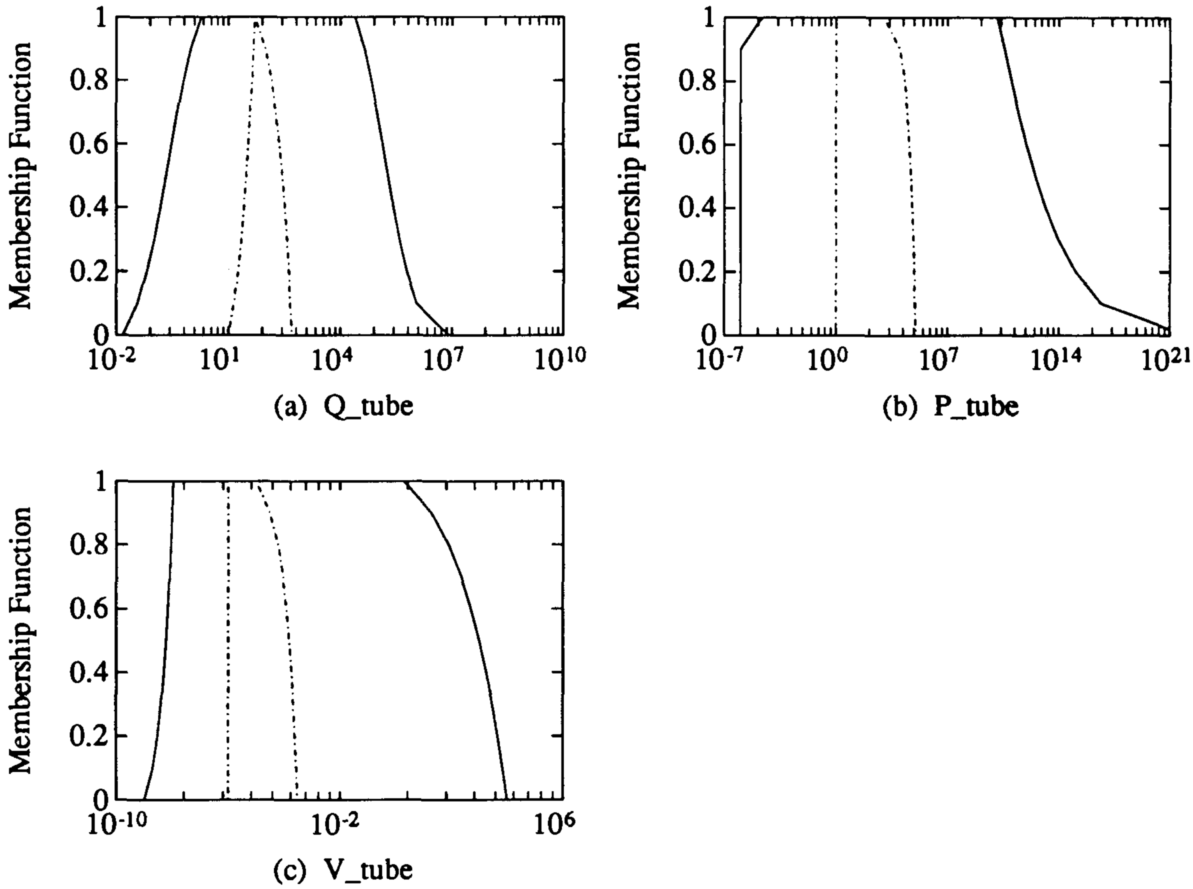

Figure 5.6: PPs and FRs for a Shell-and-Tube Heat Exchanger

Therefore, the design match f_d equals to one:

$$d_{Q_{tube}} = f_d(Q_{tube}, Q_r, \mu, \nu) = 1$$

Similarly we can obtain matches for the pressure drop across the exchanger and the internal volume. They are :

$$d_{P_{tube}} = f_d(P_{tube}, P_r, \mu, \nu) = 1$$

$$d_{V_{tube}} = f_d(V_{tube}, V_r, \mu, \nu) = 1$$

where (P_{tube}, μ) is for the performance parameter of the pressure drop across the exchanger, (V_{tube}, μ) for the internal volume of the exchanger, and (P_r, ν) and (V_r, ν) for their requirements, respectively.

We next combine all matches for the shell-and-tube heat exchanger to get an overall measure.

$$D = f_D(d_{Q_{tube}}, d_{P_{tube}}, d_{V_{tube}}) = min(d_{Q_{tube}}, d_{P_{tube}}, d_{V_{tube}}) = 1$$

Similarly, we can get the overall matches for the plain and finned heat exchangers. Since there is no intersection between the heat flow rate per unit degree temperature difference (a solid line in the diagram) and its requirement (a dotted line in the diagram) for the plain heat exchanger, shown in Figure 5.7 (a), the match for heat flow rate per unit degree temperature is zero. Therefore, it leads to zero overall measure, while the overall measure for the finned exchanger is close to 0.5, as the minimum of the match is 0.5 for the heat flow per degree of a finned heat exchanger, shown in Figure 5.7 (c). Figure 5.7 (b) shows the requirements and performance for the internal volume for the plain heat exchanger and (d) for the finned heat exchanger.

At this stage of design, we face a choice between a number of different mechanisms, such as shell-and-tube, plain, and finned heat exchanger to meet the given requirements. By using the match f_d and f_D to compare the imprecise performance for each alternative with the given requirements, the shell-and-tube heat exchanger is chosen with a maximum measure. The next stage allows us to refine this result.

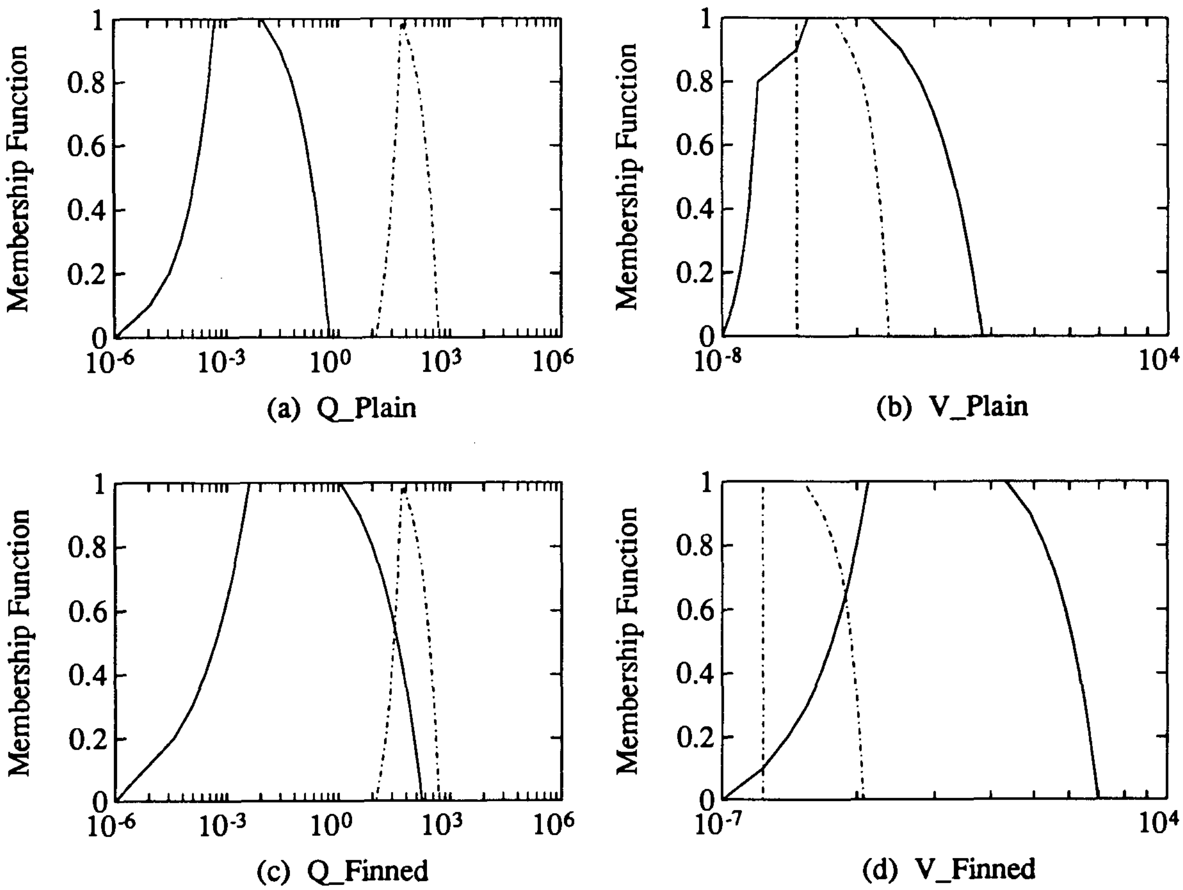

Figure 5.7: PPs and FRs for Plain and Finned Heat Exchangers

5.7.2 Refining Stage

Now the shell-and-tube heat exchanger is selected for the continuing design. Since there are wide plateaux for each parameter, it is necessary to refine each representation. The knowledge base supporting heat exchanger design contains fuzzy models of components for different classes of the shell-and-tube heat exchanger, for example, "VerySmall", "Small", and so forth. In the second level of the design, we should distinguish variants of each component by means of linguistic variables.

In the heat exchanger knowledge base, five classes can be selected to describe the shell-and-tube category, that is "VerySmall", "Medium", "Small", "Big", and "VeryBig", according to its size. Each class of the exchanger has its own diameters and lengths. There are represented by linguistic variables which allow natural specification of values for the imprecise concept. The "VerySmall" heat exchanger, for instance, diameter has the range [0.00001, 0.001] m. We can represent the possible ranges for the diameter of the shell-and-tube heat exchanger as follows:

$$\text{diameter} = \{\text{"VerySmall"}, \text{"Small"}, \text{"Medium"}, \text{"Big"}, \text{"VeryBig"}\}$$
$$\text{"VerySmall"} = [0.00001, 0.001]$$
$$\text{"Small"} = [0.0001, 0.01]$$
$$\text{"Medium"} = [0.001, 0.1]$$
$$\text{"Big"} = [0.001, 0.5]$$
$$\text{"VeryBig"} = [0.001, 5.0]$$

The design parameters for the shell-and-tube heat exchanger are the tube diameter d, tube length l, the number of tubes n and mass flow rate of gas $\dot{m}$. The number of tubes and mass flow rate of gas are the same as at the first stage, shown in Figure 5.4 (c) and (d), respectively. The diameter and length of different classes of the shell-and-tube heat exchanger are drawn in Figure 5.8, with trapezoidal fuzzy numbers. The formulae are below:

$$l_{VerySmall} = \begin{cases} 0.001 + 0.059\ y_l \\ 0.1 - 0.01\ y_l \end{cases}$$

$$l_{Small} = \begin{cases} 0.01 + 0.19\ y_l \\ 0.5 - 0.1\ y_l \end{cases}$$

$$l_{Medium} = \begin{cases} 0.01 + 0.59\ y_l \\ 1.0 - 0.1\ y_l \end{cases}$$

$$l_{Big} = \begin{cases} 0.01 + 0.59\ y_l \\ 1.0 - 0.1\ y_l \end{cases}$$

$$l_{VeryBig} = \begin{cases} 0.01 + 1.99\ y_l \\ 5 - y_l \end{cases}$$

$$d_{VerySmall} = \begin{cases} 0.00001 + 0.00059\ y_d \\ 0.001 - 0.0001\ y_d \end{cases}$$

$$d_{Small} = \begin{cases} 0.0001 + 0.0059\ y_d \\ 0.01 - 0.001\ y_d \end{cases}$$

$$d_{Medium} = \begin{cases} 0.001 + 0.059\ y_d \\ 0.1 - 0.01\ y_d \end{cases}$$

$$d_{Big} = \begin{cases} 0.001 + 0.199\ y_d \\ 0.5 - 0.1\ y_d \end{cases}$$

$$d_{VeryBig} = \begin{cases} 0.001 + 1.999\ y_d \\ 5.0 - y_d \end{cases}$$

where y_l and y_d stand for membership functions of the length and diameter, respectively.

With design parameters, Equations (5.1–5.9) and a pole computation, results can be obtained for the performance parameters. These performance parameters for the heat flow rate per degree, Q, pressure drop across the exchanger, P, and the internal

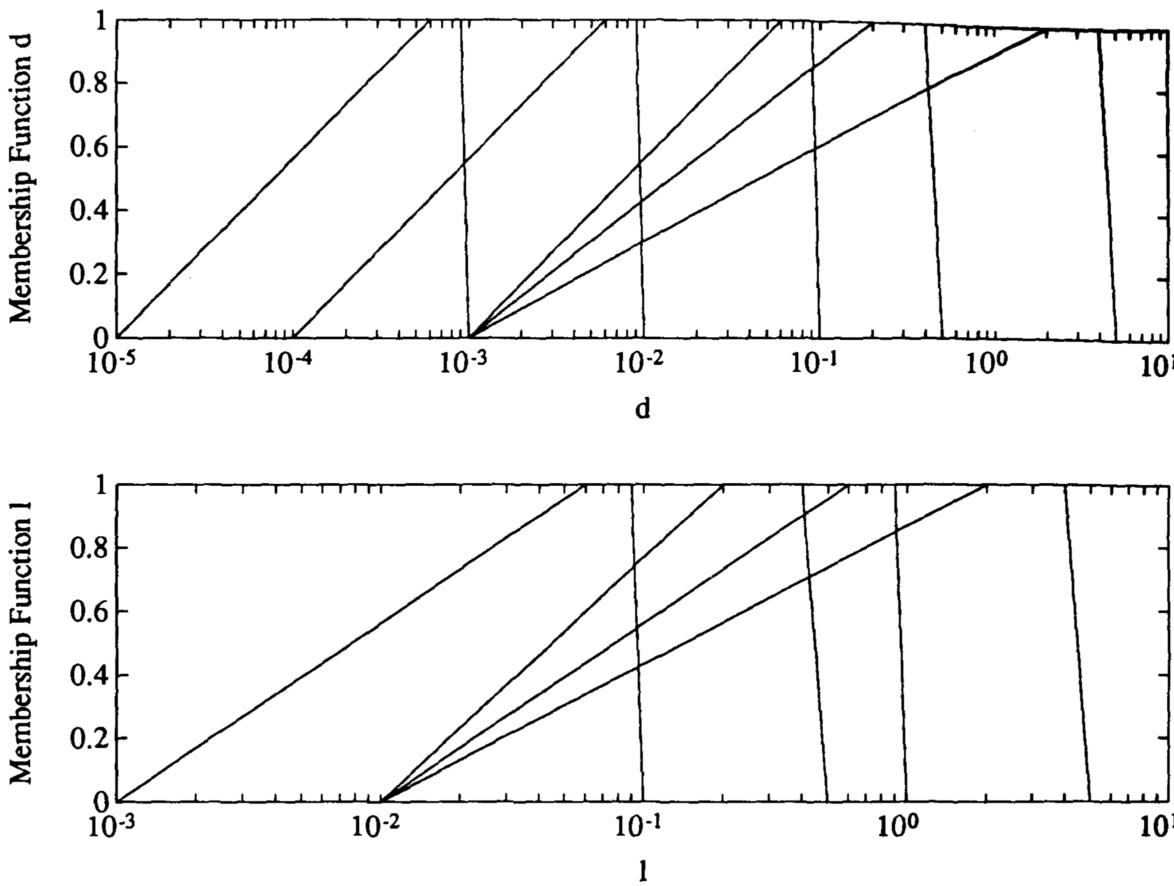

Figure 5.8: Design Parameters for a Shell-and-Tube Heat Exchanger

volume of the exchanger, V, are shown in Figure 5.9. From left to right diagrams are drawn for the "VerySmall", "Small", "Medium", "Big" and "VeryBig" shell-and-tube heat exchangers, respectively. Dotted lines are plotted for the functional requirements. Using the metric f_D, the overall measure for each class is obtained:

$$\begin{aligned} f_{VeryBig} &= 0 \\ f_{Big} &= 0.2 \\ f_{Medium} &= 0.3 \end{aligned}$$

$$f_{Small} = \; = \; 0.8$$
$$f_{VerySmall} = 0.2$$

The maximum measure is 0.8 for the "Small" size of the shell-and-tube heat exchanger.

5.7.3 Precise Stage

After first and second stages, design can be focused on a narrow range, such as "Small" shell-and-tube heat exchanger in this design. Then, using Antonsson and Wood back-track method as described in chapter 4, precise results can be obtained, which is called defuzzification. This operation is necessary to produce a nonfuzzy result. Using back-tracking strategy, each parameter is obtained for the physical system.

$$\mathrm{d} = 0.01\ m$$
$$\mathrm{l} = 0.01\ m$$
$$\mathrm{n} = 5$$
$$\dot{m} = 1.5\ kg/s$$

So choosing 0.01 m for the diameter, 0.01 m for the length and 5 tubes, we can get 34 $W/^{o}C$ of heat flow per degree, 7826 Pa pressure drop and 4 cm^3 internal volume, which are under requirements.

5.7.4 Algorithm Analysis

The algorithm used in the heat exchanger design is one example of a fuzzy calculation. From Equations (5.1-5.9), we can get the simplest expression for each performance parameter. For instance, we simplify the form for the heat flow per degree temperature difference, Q, as follows:

$$Q = 38.45\, l\, \dot{m}^{0.8}\, n^{0.2}\, d^{-0.8}$$

The intervals of design parameters $d \in A_d$, $l \in A_l$, $n \in A_n$ and $\dot{m} \in A_{\dot{m}}$ are positive. ϕ_l, ϕ_n, and $\phi_{\dot{m}}$ are positive, and ϕ_d is negative. Therefore, $A = A_d \times A_l \times A_n \times A_{\dot{m}}$ is a uniform solution space. Only two normal poles under each α-cut exist. From equations (3.1) and (3.2) these two poles can be obtained. These two poles are a global minimum value and a global maximum value. If the design parameters of a small shell-and-tube heat exchanger are

$$d = [0.0001,\ 0.01]\ m$$
$$l = [0.01,\ 0.5]\ m$$
$$n = [1,\ 100]$$
$$\dot{m} = [1.5,\ 2.0]\ kg/s$$

under $\alpha = 0$, we can obtain the interval for Q. The lower bound can be obtained by taking the minimum bound for design parameters with positive derivatives and the maximum bound for those parameters with negative derivatives, that is, $l = 0.01$, $n = 1$, $\dot{m} = 1.5$, and $d = 0.01$. Vice versa, the upper bound can be obtained using $l = 0.5$, $n = 100$, $\dot{m} = 2.0$ and $d = 0.0001$. The result for this case is $[21.17,\ 1.36 \times 10^5]$. Similarly, we can obtain output intervals for other values of α – we used $\alpha = 0.1, 0.2, \cdots, 0.9$ – and combine them to give the fuzzy number representing Q.

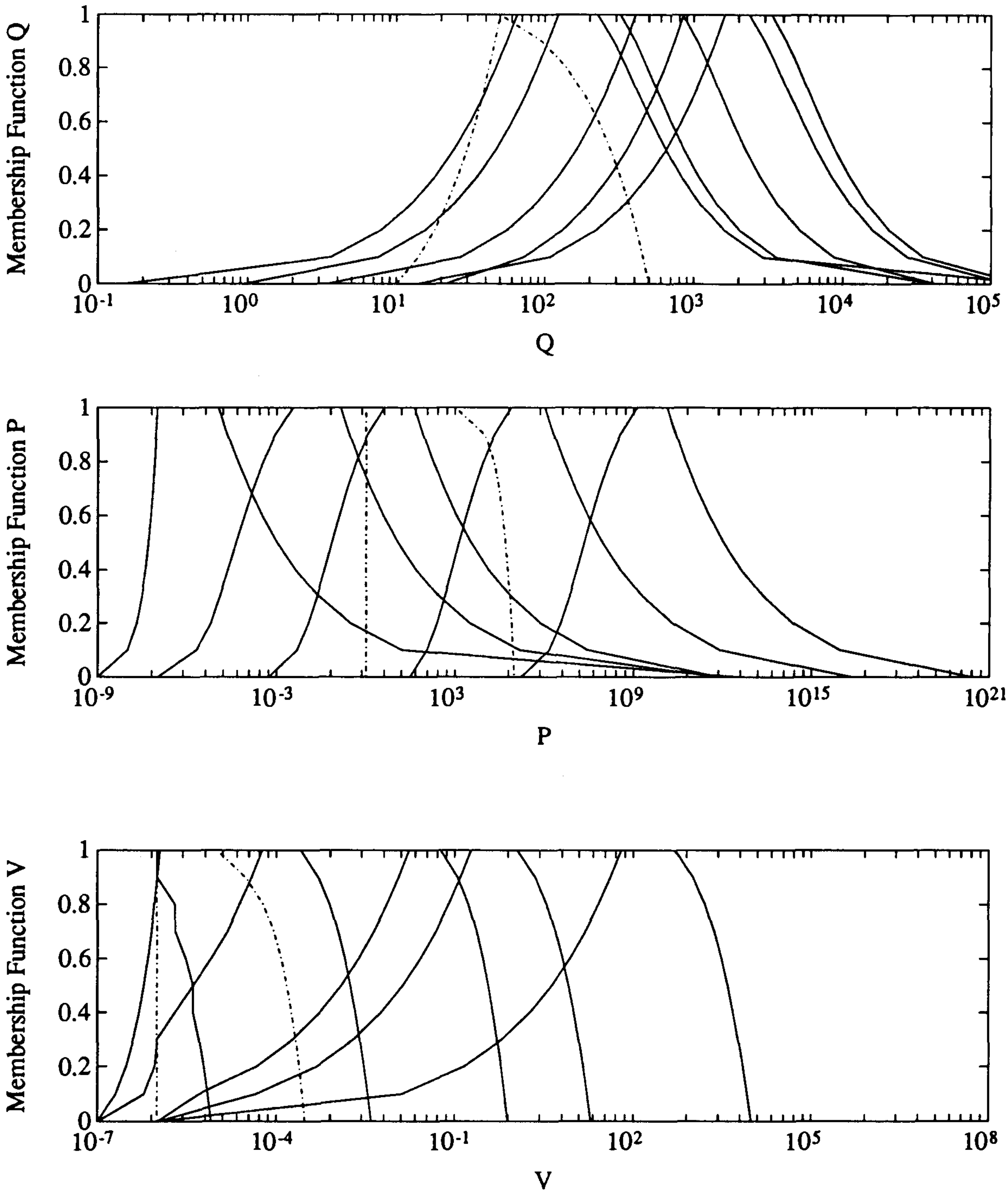

Figure 5.9: PPs and FRs for a Shell-and-Tube Heat Exchanger

5.8 Basic Architecture of Fuzzy Design

Different approaches for developing a fuzzy design system have been suggested in the last two chapters. Fuzzy design allows us to describe a social need using linguistic variables. The fuzzy mathematical models stored in the knowledge base provide a basis for decision making. Then defuzzification is needed to choose a crisp set of parameter values for the design system. Figure 5.10 illustrates a structure for heat exchanger design.

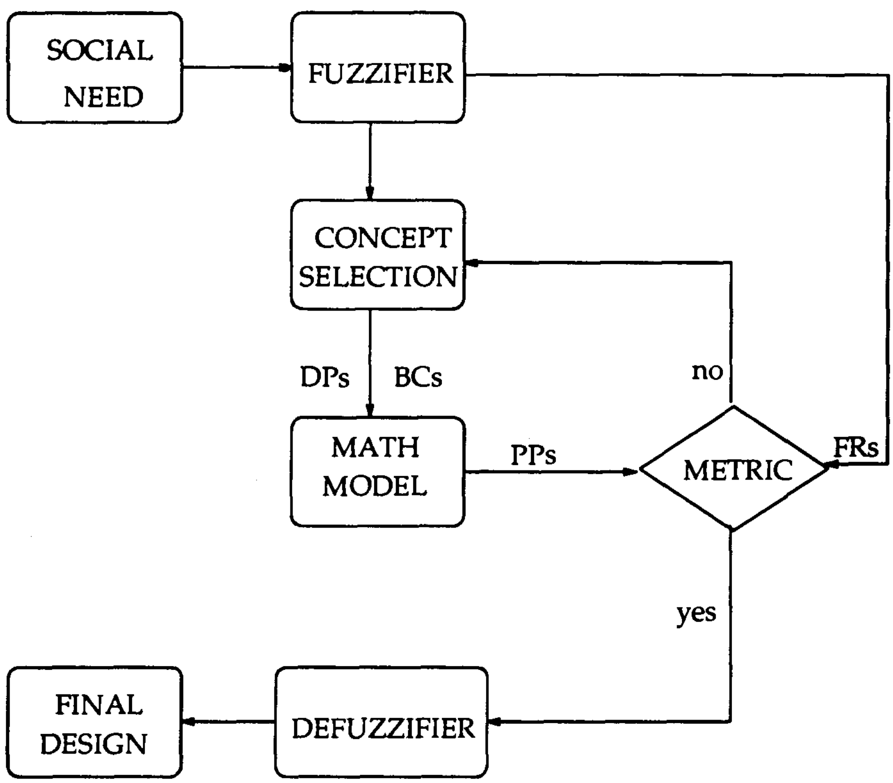

Figure 5.10: Architecture of Fuzzy Design

From Figure 5.10, boundary conditions and functional requirements are defined by the social need. These are linguistic variables and are fuzzified as either triangular or trapezoidal functions. The boundary conditions go to the fuzzy mathematical model, and the values calculated by this block are multiple fuzzy-valued performance parameters. So, we suggest a metric for trade-off between these various parameters and functional requirements in order to get the solutions. However, the solutions at this point are membership functions over ranges of values. Through the defuzzifier, we can get the exact solutions, that is, a single value for each parameter. In this thesis, we have suggested using the method of Antonsson and Wood, described in Chapter 4 for this purpose.

5.9 Conclusion

The design in previous sections cannot guide the designer through all phases of the design process. It can, however, be used for exploration to rule out certain possibilities at an early stage. The level of exploration is similar to that in a brainstorming session held early in the design process.

In the preliminary stage, the designer is not certain what value will be used for each design parameter, or at least, he can choose different values in a large range. At this stage, the imprecision is high. Fuzzy logic allows us to represent this range and a preference over the range. The approach described in chapter 5 is an example. It associates preference values with design parameters to reproduce the judgements of scale implicit in the vocabulary of a domain expert.

Fuzzy logic is one key to intelligent design. It can cope with the vagueness, approximation and uncertainty of the terms used by a human expert. Fuzzy logic is a natural way of representing human thinking which cannot practically be represented by conventional mathematical means. Fuzzy methodologies can be very useful in analysis and design, by allowing us to model vague and imprecise concepts and thus improve communication between the user and computer. It can narrow the gap between the precise and the cognitive.

The original contributions of this thesis are that we introduced the pole computation algorithm for calculating the result of applying any function to fuzzy number, presented a method for manipulating the engineering design, and used a vocabulary of component types to building a knowledge base.

5.10 Suggestions for Further Research

Further work will be continued on fuzzifier, that is, membership function selection. Although four types of fuzzy membership functions (shown in Figure 5.11) are most common, we only deal with (b) and (c) types in our design. The selection of membership functions affects the type of reasoning and the decision making to be performed by the knowledge base and the design system. Automating the process of Chapter 5 and extending the knowledge base to include more components is necessary. More research is needed in high-level inferencing of design. Fully developing the knowledge base, we can create a natural language interface between the human and the design system.

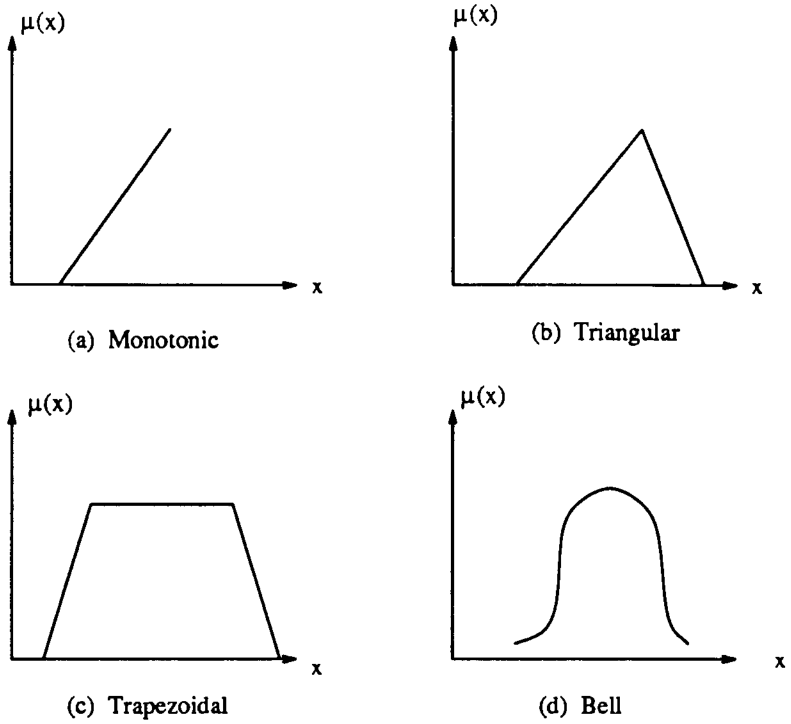

Figure 5.11: Membership Functions

BIBLIOGRAPHY

[1] G. Alefeld and J. Herzberger, *Introduction to Interval Computations*, Academic Press, Toronto, 1983.

[2] S. Baas and H. Kwakernaak, *Rating and Ranking of Multiple-Aspect Alternatives Using Fuzzy Sets*, Automatica, 13, pp. 47-48, 1977.

[3] D. C. Brown and B. Chandrasekaran, *Design Problem Solving*, Morgan Kauffmann, 1989.

[4] J. J. Buckley and D. M. Tucker, *Second Generation Fuzzy Expert Systems*, Fuzzy Sets and Systems (North-Holland), Vol. 31, pp. 271-284, 1989.

[5] J. J. Buckley, *A Generalized Extension Principle*, Fuzzy Sets and Systems (North-Holland), Vol. 33, pp. 241-241, 1989

[6] T. Calvert, J. Dickinson, J. Dill, W. Havens, J. Jones and L. Bartram, *An Intelligent Basis for Design*, IEEE Pacific Rim Conference on Communications, Computers and Signal Processing, May 9-10, 1991.

[7] W. M. Dong and F. S. Wong, *Fuzzy Weighted Averages and Implementation of the Extension Principle*, Fuzzy Sets and Systems 21, pp. 183-199, 1987.

[8] D. Dubois and H. Prade, *Fuzzy Sets and Systems*, Academic Press, New York, 1980.

[9] G. L. Glegg, *The Design of Design,* Cambridge University Press, 1971.

[10] M. M. Gupta, R. K. Ragade and R. R. Tager, *Advances in Fuzzy Set Theory and Applications,* North-Holland Publishing Company, 1979.

[11] B. V. Karlekar and R. M. Desmond, *Engineering Heat Transfer,* West Publishing Company, 1977

[12] A. Kaufmann and M. M. Gupta, *Introduction to Fuzzy Arithmetic,* Van Nostrand Reinhold Company, 1985.

[13] E. E. Kerre and A. V. Schooten, *A Deeper Look on Fuzzy Numbers from a Theoretical as well as from a Practical Point of View*, Fuzzy Logic in Knowledge-Based Systems Decision and Control (North-Holland), 1988.

[14] G. J. Klir and T. A. Folger, *Fuzzy Sets, Uncertainty and Information,* Prentice Hall, 1988.

[15] K. S. Leung and W. Lam, *Fuzzy Concepts in Expert Systems*, IEEE Computer, Sept., 1988

[16] M. Mizumoto and K. Tanaka, *Some Properties of Fuzzy Numbers*, Advances in Fuzzy Set Theory and Applications, North-Holland, Amsterdam, 1979.

[17] K. N. Otto and E. K. Antonsson, *Trade-Off Strategies in Engineering Design*, Research in Engineering Design, Springer-Verlag New York Inc., No. 2, pp. 87-103, 1991.

[18] K. J. Schmucker, *Fuzzy Sets, Natural Language, Computations and Risk Analysis,* Computer Science Press, 1984.

[19] N. P. Suh, *The Principles of Design,* Oxford University Press, 1990.

[20] T. Terano, K. Asai and M. Sugeno, *Fuzzy Systems Theory and its Application*, Academic Press Inc, 1992.

[21] D. E. Whitney, *Designing the Design Process,* Research in Engineering Design, Springer-Verlag New York Inc., 1990.

[22] J. S. Williams, A. Mathewman and D. Brown, *An Orthodontic Expert System*, Fuzzy Sets and Systems (North-Holland), Vol. 30, pp. 121-133, 1989.

[23] K. L. Wood and E. K. Antonsson, *Modeling Imprecision and Uncertainty in Preliminary Engineering Design*, Journal of Mechanism and Machine Theory 25, No. 3, pp. 305-324, 1990.

[24] K. L. Wood and E. K. Antonsson, *Computations with Imprecise Parameters in Engineering Design: Background and Theory*, ASME Journal of Mechanisms, Transmissions and Automation in Design 111, No. 4, pp. 616-625, 1989.

[25] K. L. Wood and E. K. Antonsson, *A Fuzzy Sets Approach to Computational Tools For Preliminary Engineering Design*, Advances in Design Automation, ASME, New York, pp. 263-271, Sept., 1987.

[26] K. L. Wood, E. K. Antonsson and J. L. Beck, *Representing Imprecision in Engineering Design: Comparing Fuzzy and Probability Calculus*, Research in Engineering Design, Springer-Verlag, New York, 1990.

[27] K. L. Wood and E. K. Antonsson, *Engineering Design Uncertainties*, NSF Engineering Design Research Conference, 1989

[28] R. R. Yager and L. A. Zadeh, *An Introduction to Fuzzy Logic Applications in Intelligent Systems*, Kluwer Academic Publishers, 1992.

[29] H. Q. Yang, H. Yao and J. D. Jones, *A General Method for Calculating Function of Fuzzy Numbers*, Applied Mathematics Letters, Vol. 5, No. 6, pp. 51-55, 1992.

[30] L. A. Zadeh, *Fuzzy Sets,* Information and Control, Vol. 8, pp. 338–353, 1965.

[31] L. A. Zadeh, *The Concept of a Linguistic Variable and its Application to Approximate Reasoning*, Information Science, 8, pp. 199-249, 1975.

[32] H. J. Zimmermann, *Fuzzy Set Theory and its Applications,* Second Edition, Kluwer Academic Publishers, 1991.