

1-12-2006

Web Shopping Expert Systems Using New Interval Type-2 Fuzzy Reasoning

Ling Gu

Follow this and additional works at: https://scholarworks.gsu.edu/cs_theses



Part of the [Computer Sciences Commons](#)

Recommended Citation

Gu, Ling, "Web Shopping Expert Systems Using New Interval Type-2 Fuzzy Reasoning." Thesis, Georgia State University, 2006.
https://scholarworks.gsu.edu/cs_theses/15

This Thesis is brought to you for free and open access by the Department of Computer Science at ScholarWorks @ Georgia State University. It has been accepted for inclusion in Computer Science Theses by an authorized administrator of ScholarWorks @ Georgia State University. For more information, please contact scholarworks@gsu.edu.

WEB SHOPPING EXPERT SYSTEMS USING NEW INTERVAL
TYPE-2 FUZZY REASONING

by

LING GU

Under the Direction of Yanqing Zhang

ABSTRACT

Finding a product with high quality and reasonable price online is a difficult task due to the fuzzy nature of data and queries. In order to handle the fuzzy problem, a new type-2 fuzzy reasoning based decision support system, the Web Shopping Expert for online users is proposed. In the Web Shopping Expert, an interval type-2 fuzzy logic system is used and a fuzzy output can be obtained using the up-low limit technique, which offers an opportunity to directly employ all the rules and methods of the type-1 fuzzy sets onto the type-2 fuzzy sets. To achieve the best performance the fuzzy inference system is optimized by the least square and numerical method. The key advantages of the least square method are the efficient use of samples and the simplicity of the implementation. The Web Shopping Expert based on the interval type-2 fuzzy inference system provides more reasonable conclusions for online users.

INDEX WORDS: Fuzzy Logic, Decision support system, Type-2 fuzzy logic sets, Type reduce, System optimizations, Least square method.

WEB SHOPPING EXPERT SYSTEMS USING NEW INTERVAL
TYPE-2 FUZZY REASONING

by

LING GU

A Thesis Submitted in Partial Fulfillment of the Requirements for the Degree of
Master of Science
in the College of Arts and Sciences
Georgia State University

2005

Copyright by

Ling Gu

2005

WEB SHOPPING EXPERT SYSTEMS USING NEW INTERVAL
TYPE-2 FUZZY REASONING

by

LING GU

Major Professor: Yanqing Zhang
Committee: Raj Sunderraman
Ying Zhu

Electronic Version Approved:

Office of Graduate Studies
College of Arts and Sciences
Georgia State University
December 2005

ACKNOWLEDGEMENTS

I would like to send special thanks to my thesis advisor Dr. Zhang for his help, guidance and support throughout the duration of the thesis. I would like to thank my thesis committee members.

I would also like to thank my family and friends for providing support and the occasional welcome distraction while I worked on this thesis.

TABLE OF CONTENTS

ACKNOWLEDEGBMENTS		iv
LIST OF TABLES		vii
LIST OF FIGURES		viii
CHAPTER		
1	INTRODUCTION	1
1.1	Motivation	1
1.2	Overview of related works	3
1.3	Contribution	5
1.4	Organization	7
2	DECISION SUPPORT SYSTEMS	7
2.1	Decision support systems	8
2.2	Fuzzy decision support systems	9
2.3	Web Shopping Expert	11
3	FUZZY LOGIC SYSTEMS	13
3.1	Fuzzy set theory and fuzzy logic systems	13
3.2	Type-1 and Type-2 fuzzy sets	16
3.3	Interval type-2 and type-reduce	17
3.4	Design of Web Shopping Expert	21
3.5	Structure and parameter optimizations	23
4	PARAMETER OPTIMIZATION USING LEAST SQUARE METHOD	25
4.1	Basic concepts of least square method	25

4.2	Tuning fuzzy membership function	27
4.3	Performance comparison	32
4.4	A real example	34
5	WEB SHOPPING EXPERT	36
5.1	Modeling	37
5.2	Practical issues	38
5.3	Implementation	42
5.4	Examples	46
5.5	Discussions	50
6	CONCLUSIONS	52
	REFERENCES	54

LIST OF TABLES

Table 1. Training data and results	31
Table 2. Performance before and after optimization	32
Table 3. Optimization and performance	34
Table 4. Fuzzy rules of the first group	41
Table 5. Fuzzy rules of the second group	41
Table 6. Fuzzy rules for combining the first and second group	42

LIST OF FIGURES

Figure 1. The general framework of the Web Shopping Expert.....	13
Figure 2. A typical type-1 membership functions with three linguistic values	14
Figure 3. The basic architecture of a type-1 FLS with crisp inputs and output	15
Figure 4. The primary (a) and secondary (b) membership functions of a type-2 fuzzy set ...	17
Figure 5. A schematic diagrams of the primary (a) and secondary (b) membership functions of an interval type-2 FLS	18
Figure 6: Integrated fuzzy output of R_{avg} and its uncertainties	21
Figure 7. The structure of interval type-2 fuzzy logic systems	22
Figure 8. The membership function of the k^{th} variable.	29
Figure 9. The membership functions of the input and output variables	30
Figure 10. The membership functions of the price and location before (dash line) and after (solid line) optimization	31
Figure 11. The errors of type-1 and type-2 after training	32
Figure 12. The membership functions of price, CUP, memory, display, and satisfaction for the laptops	35
Figure 13. Errors of before and after optimization	36
Figure 14. Membership functions and random values	40
Figure 15. The algorithm of the parameter optimization procedure	43
Figure 16. Web Housing Expert Home Page.....	46
Figure 17. Web Housing Expert System Page	47
Figure 18. Web Housing Expert Searching Result Page (1)	48
Figure 19. Web Housing Expert Searching Result Page (2)	48

Figure 20. Web Housing Expert Help Interface	49
Figure 21. Web Housing Expert Contact Interface	49
Figure 22. The expected and calculated values of satisfaction obtained from type-1 and type-2 FLSs	50

1. INTRODUCTION

With the rapid increased use of computers and concurrent enhancement of information technology, intelligent systems are being extensively developed to assist with decision making. A decision support system is a program that aids in making decisions and involves queries and constraints that need to be satisfied. Considering the imprecise and vague nature of the data and information in the real-world, it is obvious that the ability of managing uncertainty turns to be a crucial issue for decision support systems. Fuzzy logic theory provides a very useful solution for understanding, quantifying, and handling these vague, ambiguous, and uncertain characteristics of information. The fuzzy decision support systems, unlike classical logical systems, aim at modeling the imprecise modes of reasoning to make rational decisions in an environment of uncertainty and imprecision [1].

1.1 Motivation

A decision support system is an integrated, interactive computer system, consisting of analytical tools and information management capabilities, designed to aid decision makers in solving relatively large, unstructured problems [2]. This system could also be termed a multi-criteria analysis system which involves satisfying a set of criteria or constraints [3]. For example, one could be looking for buying a house and he/she might have various criteria or constraints such that the house is not too far from a highway exit, within 15-minute driving distance from his/her work place, which costs around \$300,000, the house is in a good location, a house has a big backyard etc. However, it is difficult to apply a conventional mathematical model, i.e., an expert system, to elicit these kinds of constraints due to the fact that the conventional mathematical model needs to precisely describe all the characteristics of the system and it lacks

flexibility. There are strong reasons for the lack of such facilities in conventional models. First, any decision process is extremely complex due to the large number of parameters under consideration. Second, one faces the problem which is how to "explain" to a machine the meaning of vague concepts usually used in situation characterization, such as the ones implicit in linguistic expressions like "not too far", "good location", or "big backyard". Another important problem is the uncertainty inherent to the information used by decision support systems based on natural languages, i.e., human languages. Even if this language appeals to some formalism, there still will be a question of how to interpret a constraint like "within 15-minute driving distance". Particularly, in the scenario of house purchase, there are multi-criteria, but where the "good location" is considered "slightly" more important than others. Furthermore, because of the uncertainty of words, "good location" has different meaning for different people. Some people would like downtown areas, some people would like a good school district, and some people would like quiet places. Conventional mathematical models present serious limitations to manipulate these kinds of logical uncertainties that arise due to human thinking, perception, vague concepts, unreliable measurements, and lexical imprecision because they assume that all variables in a problem are well-defined..

One possible approach to deal with the vague concepts is fuzzy logic systems, which are based on the fuzzy set theory, formulated and developed by Zadeh [4]. Fuzzy set theory is a generalization of classical set theory that provides a way to absorb the uncertainty inherent to phenomena whose information is vague and supplies a strict mathematical framework, which allows its study with some precision and accuracy. A fuzzy logic system (FLS) can deal with the vagueness and uncertainty residing in the knowledge possessed by human beings or implicated in the numerical data, and it allows us to represent the system parameters with linguistic terms

[5]. Fuzzy rules and membership functions have been used as a key tool to express knowledge. Moreover, the relative importance of the criteria that may not equally influence a decision can also be considered by the FLS [6].

This thesis aims at promoting the integration of fuzzy logic into a decision support system for the benefit of decision-makers. The focus of this thesis is to develop a fuzzy decision support system, which has the capability of handling multi-criteria and constraints and dealing with vague and imprecise data. The system will take into account the vague and imprecise character of data and information on the one hand, and represent the user or expert's preferences and knowledge precisely on the other hand. For this purpose, a generic model has been used to develop the decision support system. It is mainly based on type-2 fuzzy logic sets, which are suitable for handling not only vague and imprecise information but also uncertainty of words. Other techniques have been introduced into the system to solve the type-reduce, structure optimization, and parameter optimization problems.

1.2 Overview of related works

The main attraction of FLSs is their unique characteristics of the capability of handling complex, nonlinear, and sometimes mathematically intangible dynamic systems using simple solutions [7]. FLSs can provide better performance than conventional non-fuzzy approaches, with less development cost [1]. FLSs represent the human decision-making process with a collection of fuzzy *if-then* rules. The successful design of a FLS depends on several factors such as choice of the rule set, membership functions, inference scheme, and the defuzzification strategy. Of these factors, the selection of an appropriate rule set with proper membership functions is the most important one. However, obtaining an optimal set of fuzzy rules and

membership functions is not an easy task. It requires time, experience, and skills of the operator for the tedious fuzzy tuning exercise.

With an increase in the number of input variables, the possible set of fuzzy rules increases rapidly. For instance, if each variable (both input and output) has p fuzzy subsets, then for a FLS with q inputs and one output, the total number of the possible rules is $p^q - 1$. It is difficult to determine a small subset of rules from such a large “rule space” that would be suitable for controlling the process. In principle, there is no a general method for the fuzzy logic setup, although a heuristic and iterative procedure for altering the membership functions to improve performance has been proposed [8], even this is not optimal. Recently, many researchers have considered a number of intelligent schemes for the task of optimizing the fuzzy rules and membership functions. There have been several attempts both under supervised and self-organized paradigms for obtaining a good rule base. Some of these methods use neural networks [9] and others use genetic algorithms (GA) [10]. The rule base tuning has been attempted primarily in two ways: through tuning of membership functions of a given rule set or through selection of an “optimal” subset of rules from all possible rules.

Wang *et al.* used a table-lookup scheme to generate fuzzy rules directly from numerical examples and proved that this fuzzy inference system is a universal approximation by the Stone–Weierstrass theorem [11]. Nozaki *et al.* presented a heuristic method for generating Takagi–Sugeno (TS) fuzzy rules from numerical data, and then translated the consequent parts of TS fuzzy rules into linguistic representation [10]. Grauel *et al.* have investigated the connection between the shape of transfer functions and the shape of membership functions, where membership functions for multi-input of Sugeno controllers and designed rules were derived [12]. Klawonn has applied to construct a fuzzy controller from the training data [13]. Hong and Lee

have pointed out that the drawbacks of most fuzzy controllers and fuzzy expert systems are that they need to predefine membership functions and fuzzy rules to map numerical data into linguistic terms and to make fuzzy reasoning work [14]. They proposed a method based on the fuzzy clustering technique and the decision tables to derive membership functions and fuzzy rules from numerical data.

Decision support systems have been applied in various fields including medical diagnosis, business, military, industry, air traffic control and so on [7, 15]. Generally previous experience or expert knowledge is used to design decision support systems. It becomes interesting when no prior knowledge is available. The need for an intelligent mechanism for decision support comes from the limits of human knowledge processing. Artificial intelligence techniques have been explored to construct adaptive decision support systems. The use of the artificial intelligence opens a door to capture imprecision, uncertainty, learn from the data/information and continuously optimize the solution by providing interpretable decision rules would be the ideal technique. Several adaptive learning frameworks for constructing intelligent decision support systems have been proposed [10, 15-17]. Most of the existing fuzzy decision support systems are based on type-1 fuzzy sets. To handle the higher level uncertainties of the nature, Yager proposed the first decision support system using type-2 fuzzy sets [18].

1.3 Contributions

Considering the fuzzy nature of information in the real-world, it becomes obvious that the ability of managing uncertainty and imprecision turns to be a crucial issue for a decision support system. To deal with the uncertainty and imprecision, the approximate reasoning capability of fuzzy logic is the fundamental key [4]. Many approaches based on the fuzzy logic theory, which

provide flexible information processing capabilities to manage the imprecision, uncertainty, and approximate reasoning, have been proposed but the processing is far from satisfactory [15]. Fuzzy decision support systems usually employ type-1 fuzzy sets, which represent uncertainty by numbers in the range $[0, 1]$.

Type-1 FLSs are not able to completely capture the linguistic uncertainties in real-world, e.g. the uncertainty of words. Type-2 fuzzy set provides a powerful tool to overcome the limitation of type-1 FLSs. Type-2 fuzzy sets are an extension of type-1 fuzzy sets in which uncertainty is represented by an additional dimension. Type-2 FLSs offer better capabilities to handle linguistic uncertainties by modeling the uncertainties using type-2 membership functions. The extra dimension in type-2 FLS gives more degrees of freedom for better representation of uncertainty compared to type-1 fuzzy sets [19]. Using type-2 sets, FLSs provide the capability of handling a higher level of uncertainty and provide a number of missing components that have held back successful deployment of FLSs in human decision making.

There is no any online type-2 decision support system because of time cost due to the complex calculations of the type-2 fuzzy sets. This thesis aims to develop a Web based decision support system (Web Shopping Expert) using the interval type-2 FLS. My major contributions in this work are:

- Apply the interval type-2 fuzzy set into the decision support system to represent the high level uncertainty.
- Develop a method to handle the complex calculations in the interval type-2 FLS.
- Consider the linguistic variables separately to reduce the number of fuzzy rules.
- Introduce the least square method to optimize the parameters of membership function to obtain a better performance.

1.4 Organization

This thesis focuses on developing a decision support system (Housing Expert) based on the interval type-2 FLS, which is optimized by least square method. Detailed explanations for the concepts of decision support systems, fuzzy logic theory, and least square method are presented. I have demonstrated how to apply an interval type-2 FLS into decision support systems. The rest of the thesis is organized as follows. Section II describes the basic concepts and background of decision support system. Section III introduces the fuzzy logic theory and type-1 and type-2 FLS. Section IV illustrates parameter optimizations using the least square method and a real application is used to show how the least square method works. Section V presents a Web based decision support system using the interval type-2 FLS. Finally, Section VI gives conclusions.

2. DECISION SUPPORT SYSTEMS

Adelman has defined decision support systems as “interactive computer programs that utilize analytical methods, such as decision analysis, optimization algorithms, program scheduling routines, and so on, for developing models to help decision makers formulate alternatives, analyze their impacts, and interpret and select appropriate options for implementation” [20]. Andriole has also defined decision support systems as consisting of “any and all data, information, expertise or activities that contribute to option selection” [2]. Implicit in these definitions, a decision support system is an integrated computer program consisting of analytical tools and information management capabilities, designed to aid decision makers in solving relatively large, unstructured problems. A decision making process is based on scoring / ranking or measures / operations. One of the key objectives for any decision support system is to provide the required operators and measures for the decision making process.

2.1 Decision support systems

Since the late 1970s, a number of researchers and companies had developed interactive information systems that used data and models to help managers analyze semi-structured problems [2]. These diverse systems were all called decision support systems. From those early days, it was recognized that the decision support system could be designed to support decision-makers at any level in an organization [21]. Decision support systems could support operations, management and strategic decision-making. A variety of models were used to develop decision support systems including optimization and simulation [2]. Also, statistical packages were recognized as tools for building decision support systems. Artificial Intelligence researchers began work on management and business expert systems in the early 1980s [7].

The purpose of the conventional decision support systems (e.g. expert systems) is to emulate the reasoning process of human experts within a specific domain of knowledge. The knowledge engineering plays an important role in expert systems. It is concerned with the concepts and methods of symbolic inference, or reasoning, and how the knowledge used to make those inferences will be represented inside the computer. The knowledge engineering involves knowledge acquisition, knowledge representation, and human-machine interaction [2]. The knowledge acquisition is used to extract knowledge from the opinions of experts or a set of numerical data. There is a lot of work done on error analysis in expert systems and most of these deal with errors in the data which occur mainly due to the source of the data [20, 21]. While there has been a lot of emphasis on these types of errors, there is another kind of errors in decision making processes and these are mostly neglected. It is that even if we have highly accurate data, we still cannot achieve accurate results or make accurate decisions unless the analysis is done accurately. Analysis techniques in conventional decision support systems do not

consider for logical inaccuracies that arise due to human thinking, perception, vague concepts, unreliable measurements, imprecision, etc., and assume that all variables in a problem are well-defined. Decision making processes are traditionally handled by either the deterministic or probabilistic approaches [20]. The former provides an approximate solution, completely ignoring uncertainty, while the latter assumes that any uncertainty can be represented as a probability distribution. Unfortunately, both approaches only partially capture reality.

The challenge is how to deal with the very complicated and large information, which is usually full of vagueness, ambiguousness, and uncertainty as well as conflicts and contradictions. To handle the huge amount of information effectively and efficiently, a new type of autonomous and robust decision support system that can offer restrict the amount of information and provide the most relevant information is required.

2.2 Fuzzy decision support system

Today, technology advances in the computer industry have led to the rapid development of electronic-commerce applications and intelligent systems are being extensively developed to assist with online decision-making. One of the major problems for the conventional decision processes is the large quantity of information and its vague and sometimes conflicting nature [20]. The capability of handling imprecise concepts is essential due to the nature of information in the real world. In fact, much of human reasoning is approximate rather than precise in nature. A simple example would be in proximity analysis, wherein we would like to extract certain spatial features that exist within a particular distance, say 15 miles. A conventional decision support system, when instructed to do this proximity search, will extract features that are exactly within 15 miles. Even there is a feature of interest even as close to the boundary as 15.01 mile,

the system will not locate it. This is a serious logical error that cannot be handled by conventional methods. With the ever increasing demand for accurate results, there is a demand for making the analysis intelligent, rather than it being able to perform strictly just what was instructed. Hence, what is needed is a fuzzy analysis that can handle this situation intelligently.

In a decision making process, the problem can be treated as a query formulated by multiple criteria and constraints [2, 3]. In this case, the user often needs to select or rank alternatives that are associated with these criteria and constraints. Fuzzy logic in this context aims at representing the imprecise nature of the information. The use of fuzzy logic to represent and manipulate both quantitative and qualitative criteria of decision-making is an important alternative for building intelligent decision support systems [2]. Fuzzy querying and ranking is a very flexible tool in which linguistic concepts can be used in the queries and ranking in a very natural form. A fuzzy query is defined on the basis of attributes or variables that are represented by fuzzy measures. However, these variables can be characterized by different degrees of importance that can be represented by weights that correspond to the user preferences [22]. In ranking-based decision making processes, the values of attributes over the database are used for scoring each element of the database with respect to the query and the user preferences. The scores are calculated using similarity measures and aggregation operators. Then, data can be ranked according to these scores which correspond to their degrees of compatibility with the query. In addition, the selected objects do not need to match the decision criteria exactly, which gives the system a more human-like behavior. Fuzzy logic allows more choices in aggregation operators such as fuzzy Min and Max to express fuzzy queries.

The critical factor that affects the performance and reliability of a decision support system is the quantity and quality of the fuzzy rules that are stored in knowledge bases [15].

Consequently, knowledge acquisition becomes the major task in the intelligent system development. Domain facts and rules are the major knowledge sources that knowledge acquisition is concerned with. Essentially, knowledge acquisition starts from that derived when knowledge engineers interview the domain experts. The knowledge engineers then translate the obtained information into the form of knowledge bases without changing the initial meaning [21]. It is an interesting question: how does a decision support system handle the situation when the knowledge is incomplete or not available.

2.3 Web Shopping Expert

Searching database records and ranking the results based on multi-criteria queries are central for many database applications used within organizations in finance, business, industry and other fields. As an application of FLSs, an online decision support system (Web Shopping Expert) for house hunters to find their dream houses is developed. The basic idea of the Web Shopping Expert is to extract information from a web based database that matches user's queries, and filters out unmatched information. The match is measured by a ranking function.

Unlike most of the conventional decision support systems which are modeled using crisp logic and queries, which causes with imprecise and subjective processes results in rigid systems, the Web Shopping Expert employs fuzzy querying and ranking as a flexible tool allowing approximation where the selected objects do not need to exactly match the criteria for resembling natural human behavior. The key features of the Web Shopping Expert are:

- to assist decision-makers in assessing the consequences of decision made in an environment of imprecision, uncertainty, and partial truth and providing a systematic risk analysis;

- to help decision-makers answer “What if Questions”, examine numerous alternatives very quickly and find the value of the inputs to achieve a desired level of output;
- to be used with human interaction and feedback to achieve a capability to learn and adapt through time.

The higher level uncertainty should also be taken into consideration because of the vague, ambiguous, and uncertain criteria and constraints without that the decision-making process might lead to inaccurate decisions [19]. Hence, the use of interval type-2 FLS is proposed to handle uncertainties such as sharp class boundaries, measurement uncertainties, linguistic knowledge, and inherent uncertainty due to the nature of information.

The Web Shopping Expert consists of four major components: the FLS, the User Interface, the Database System, and the Training Processes, as shown in Figure 1. The FLS is the core module of the system. It has been developed to be generic so that it would fit other applications. The main FLS component is the query structure, which utilizes membership functions, similarity functions and aggregators. Note that location always is a very important factor for house hunters. However, the meaning of “good location” is different for different people. Some people like downtown while some people prefer suburb. Older people want their home close to a hospital. Young parents want their house in good school districts. To handle this uncertainty of words, type-2 fuzzy sets are introduced into the FLS. The FLS has 4 input linguistic variables (price, location, year, and distance) and an output (satisfaction). Through the user interface, a user can input criteria and constraints for a query, run different queries, and display results. The training component modifies parameters of membership functions based on the least square solution between the input-output relations. The database stores all data for training and queries.

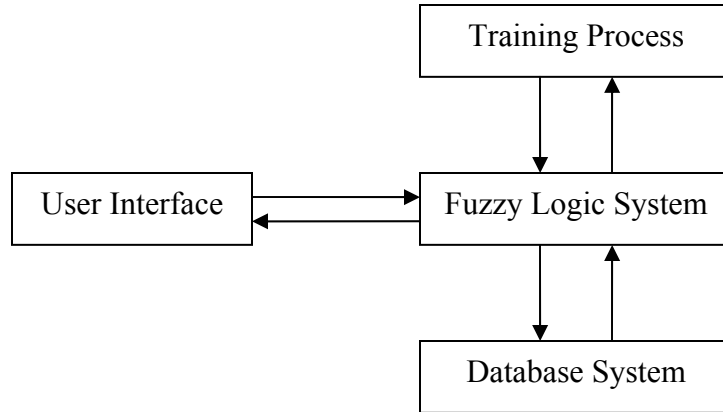


Figure 1. The general framework of the Web Shopping Expert

3. FUZZY LOGIC SYSTEMS

Since the introduction of the basic conceptions of the fuzzy set theory, FLS have been studied for more than 30 years. The success of their applications is in various fields. They can be very helpful to achieve classification tasks, offline process simulation and diagnosis, decision support tools, and process control [15].

3.1 Fuzzy set theory and fuzzy logic systems

FLSs are based on the fuzzy set theory, fuzzy if-then rules, and fuzzy reasoning. Fuzzy knowledge is expressed by the concepts of the fuzzy sets and linguistic variables which are characterized by membership functions [4, 7]. In the fuzzy set theory a fuzzy set A of a set X is defined as a set of ordered pairs, each with the first element from X and the second element from an interval of $[0, 1]$:

$$\mu_A: X \rightarrow [0, 1]. \quad (1)$$

This defines a mapping, μ_A , between elements of the set X and values in the interval $[0, 1]$. The value zero is used to represent complete non-membership, the value one is used to represent

complete membership, and values in between are used to represent intermediate degrees of membership. Usually, the mapping is described by a function, the membership function of A. There are three types of membership functions: Gaussian, Triangular and Trapezoidal. These functions have three main points, for the lower bound, upper bound and the point of maximum membership. Figure 2 presents a typical type-1 membership function in triangular shape. In this thesis all the membership function are defined in triangular shape. For other functions, optional extra points may be used to define the shape. There are many alternative methods to define the t-norm operator. The most frequently used one is the Maximum-Minimum operation.

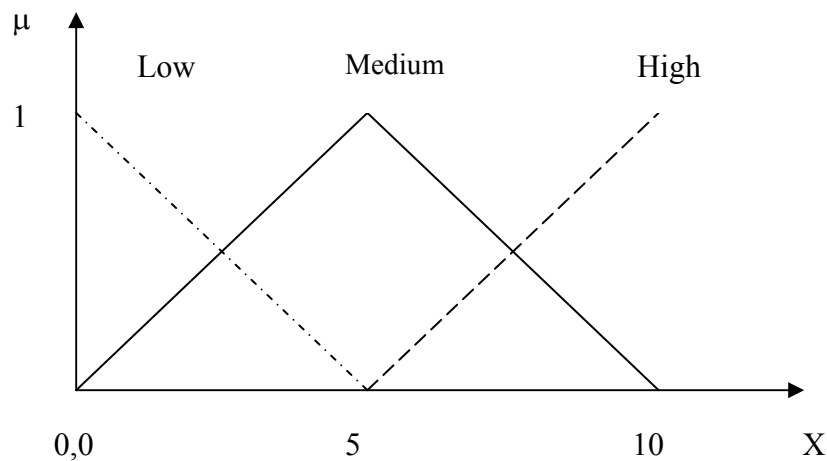


Figure 2. A typical type-1 membership functions with three linguistic values

The fuzzy rules are important for representing fuzzy knowledge. Most fuzzy systems (also in this thesis) employ the inference method proposed by Mamdani-Assilian in which the rule consequence is defined by fuzzy sets and has the following structure [23]:

$$R_i: \text{If } x_1 \text{ is } A_1, x_2 \text{ is } A_2, \dots, x_n \text{ is } A_n \text{ then } y = B_i . \quad (2)$$

Takagi and Sugeno proposed an inference scheme in which the conclusion of a fuzzy rule is constituted by a weighted linear combination of the crisp inputs rather than a fuzzy set, and which has the following structure [24]:

R_i : If x_1 is A_1 , x_2 is A_2 , ..., x_n is A_n

$$\text{then } y = p_0 + p_1A_1 + p_2A_2 + \dots + p_nA_n . \quad (3)$$

Takagi-sugeno FLS usually needs a smaller number of rules, because their output is already a linear function of the inputs rather than a constant fuzzy set.

The basic structure of a FLS consists of four components (fuzzifier, fuzzy rule base, fuzzy inference engine, and defuzzifier) with n-input and 1-output [7]. Figure 3 shows the basic architecture of a type-1 FLS with crisp inputs and output implementing a non-linear mapping from its input space to its output space [7]. The inference engine maps inputs into fuzzy logic sets. Using many different fuzzy logic inferential procedures, it handles the way in which rules are activated and combined. The fuzzifier converts crisp inputs into fuzzy terms, the fuzzy rule base contains a set of fuzzy rules, the fuzzy inference engine generates fuzzy outputs for inputs by making fuzzy reasoning based on the fuzzy rules, and the defuzzifier converts the fuzzy output into the crisp output.

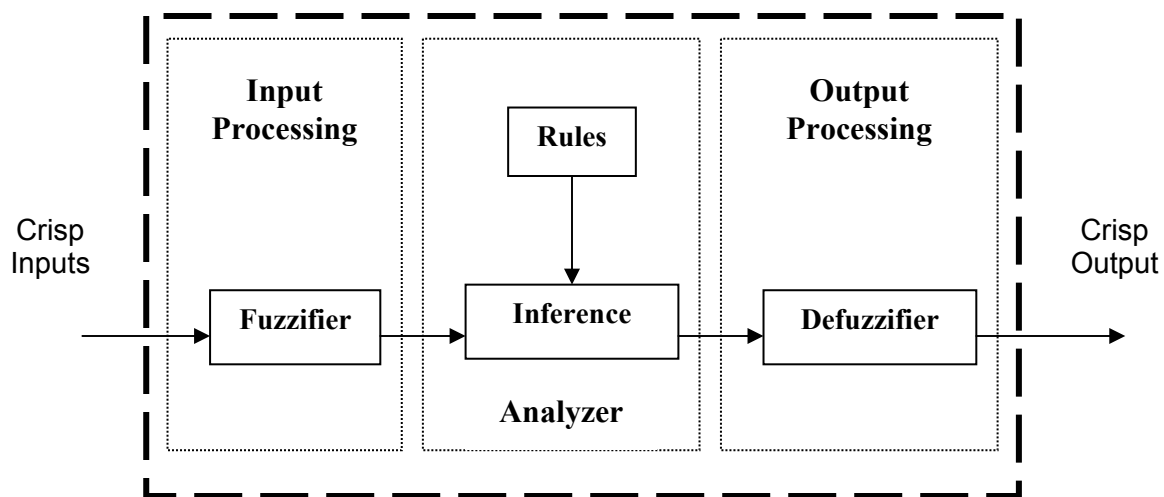


Figure 3. The basic architecture of a type-1 FLS with crisp inputs and output.

3.2 Type-1 and Type-2

Two kinds of fuzzy sets are used in FLSs: type-1 and type-2. Type-1 fuzzy sets are described by membership functions that are totally certain. The membership functions of type-1 fuzzy are two-dimension in which each element of the type-1 fuzzy set has a membership grade that is a crisp number in $[0, 1]$. On the other hand, type-2 fuzzy sets are described by membership functions that are themselves fuzzy [19]. The membership (primary membership) grade for each element of a type-2 fuzzy set is a fuzzy set in $[0, 1]$ and the possibility of the primary membership is defined by a secondary membership function, which is also be in a fuzzy set in $[0, 1]$. Type-2 fuzzy set provides a method to handle the uncertainty of linguistic variables. The membership functions of type-2 fuzzy are three dimensions. The extra third dimension in type-2 provides an additional degree of freedom to capture more information about the represented term [25]. This feature allows us quantify different kinds of uncertainties that can occur in a FLS. Type-2 fuzzy sets are useful in circumstances where it is difficult to determine the exact membership function for a fuzzy set. Type-1 fuzzy sets handle uncertainties by using precise membership functions and once membership functions have been chosen all the uncertainties disappear, because type-1 membership functions are totally precise. Type-2 fuzzy sets handle uncertainties about the meaning of the words (e.g. price and quality) that they represent by modeling the uncertainties using type-2 membership functions.

Figure 4a shows a typical primary membership function of type-2 fuzzy set. It is defined as triangular shape. For given value of x , μ can be any value between μ_L and μ_H defined by the primary membership function. Figure 4b presents its secondary membership function and it is also defined as triangular shape. The probability of μ to be a certain value between μ_L and μ_H is defined by the secondary membership function. The shaded region in Figure 4a is called

footprints of uncertainty and represents the collective domain of the respective type-2 fuzzy set [27]. Footprint of uncertainty enables people to graphically depict type-2 fuzzy sets in two-dimensions. Note that a type-1 FLS is a special case of a type-2 FLS. In Figure 4a if μ_L equals to μ_H , the type-2 FLS reduces to a type-1 FLS. If the secondary membership function becomes rectangular shape, in other word the probability of μ to be a certain value between μ_L and μ_H is same (equals to 1), this type-2 fuzzy set is called interval type-2 fuzzy set [25].

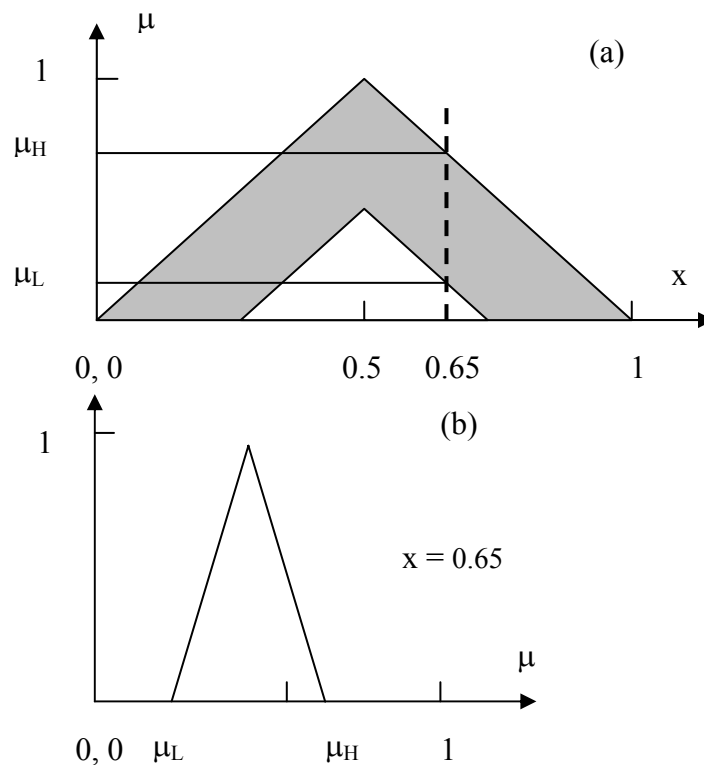


Figure 4. The primary (a) and secondary (b) membership functions of a type-2 fuzzy set.

3.3 Interval type-2 and type-reduce

Generally, computations of type-2 FLS are very complicate because type-reduction is awfully intensive [19, 25]. Mizumoto and Tanaka have studied the operations and properties of membership functions of type-2 fuzzy sets, and examined the operations of algebraic product and sum of them [26]. Kamik and Mendel have established a complete type-2 FLS theory to handle

uncertainties in FLS parameters based on their general formula of the extended sup-star composition [27].

If the secondary membership function is an interval set, the computations of type-2 FLS (interval type-2 FLS) become simpler [25]. Figure 5a and b present the primary and secondary membership functions of an interval type-2 fuzzy set. However, the analyses of the interval type-2 FLS still require extensive calculations. In an interval type-2 FLS, μ has the same probability (equals to 1) to be any values in its interval range, as shown in Figure 5a. Therefore, we can represent Figure 5a as Figure 2 in which

$$\mu_{\text{avg}} = (\mu_H + \mu_L) / 2, \quad (4)$$

and
$$\Delta\mu = \mu_H - \mu_{\text{avg}} = \mu_{\text{avg}} - \mu_L. \quad (5)$$

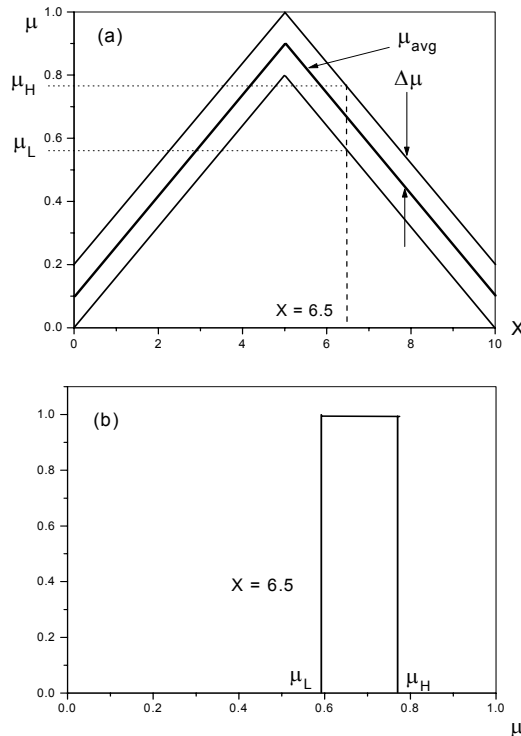


Figure 5. A schematic diagrams of the primary (a) and secondary (b) membership functions of an interval type-2 FLS.

If we can find a method to calculate the μ_{avg} and $\Delta\mu$ separately in the operations of type-2 FLS, the type-2 FLS is reduced to type-1 FLS. Then, all the methods and operations used in type-1 FLS can be applied and the computations will become much easier. Note that as the $\Delta\mu$ goes to zero, the type-2 FLS becomes type-1 FLS.

The standard error analysis provides a useful method to solve this problem. Assume that the relationship between the measured quantities x , y , and z in an experiment and result R is given by

$$R = f(x, y, z). \quad (6)$$

where function, $f(x, y, z)$ defines the mapping between x , y , z , and result R . The measurements will have associated errors given by δx , δy , and δz . By means of a Taylor expansion the change in R can be expressed in terms of changes associated with the independent variables.

$$\begin{aligned} R + \delta R &= f(x + \delta x, y + \delta y, z + \delta z) \\ &= f(x, y, z) + \frac{\partial f}{\partial x} \delta x + \frac{\partial f}{\partial y} \delta y + \frac{\partial f}{\partial z} \delta z + \dots \end{aligned} \quad (7)$$

$$\delta R = \frac{\partial f}{\partial x} \delta x + \frac{\partial f}{\partial y} \delta y + \frac{\partial f}{\partial z} \delta z + \dots \quad (8)$$

In these expansions the higher order terms have been neglected. As it stands, the above expressions takes into account the effect of the algebraic sign of the changes (δx , δy , and δz). Such a relation can be used to correct R for the effects of known systematic errors, but is not practical when the errors are random in nature [28]. This is because the standard deviation calculated in the case of random errors has no sign associated with it.

If we treat the uncertainty as a propagation error, the interval type-2 membership function becomes a type-1 membership function with an uncertainty of $\Delta\mu$ or two type-1 membership functions. Then, the standard error method can be directly applied to calculate the error

(uncertainty). Hence, we can use the type-1 FLS methods combining with the standard error method to calculate the interval type-2 fuzzy sets. This makes computations much easier by offering the opportunity to directly employ all the methods and operations of type-1 fuzzy sets to the interval type-2 fuzzy sets. The use of the error technique also provides more flexible and reasoning results to the decision support system.

Therefore, for an interval type-2 membership function because μ has the same probability to be any values in its interval range, we can denote the membership function using μ_{AVG} to present the mean value of μ and $\Delta\mu$ to present the uncertainty (or error as in error analysis).

The interval type-2 can be represented by

$$\mu = \mu_{AVG} \pm \Delta\mu . \quad (9)$$

Similarly, the μ_{AVG} gives the average value of the crisp output and $\Delta\mu$ represents the high level uncertainty due to the fuzzy nature.

Figure 6 presents the integrated fuzzy output of average value of μ_{AVG} (red line) and the additional uncertainty (black lines). To convert these fuzzy outputs to crisp outputs, a type-1 defuzzification process is used. The centroid of area of R for the low and up limits are calculated by the standard error method:

$$R_{AVG} = \frac{\int \mu(r, \Delta\mu) r dr}{\int \mu(r, \Delta\mu) dr} , \quad (10a)$$

$$\Delta R = 2 * \frac{\int \Delta\mu(r) r dr}{\int \mu(r) dr} . \quad (10b)$$

Note that type-2 gives a mean value R_{AVG} that is same as the output of type-1 because of the same membership functions but with an interval $2\Delta R$ to present the higher level uncertainty.

The final output can be represented by

$$R = R_{AVG} \pm \Delta R . \quad (11)$$

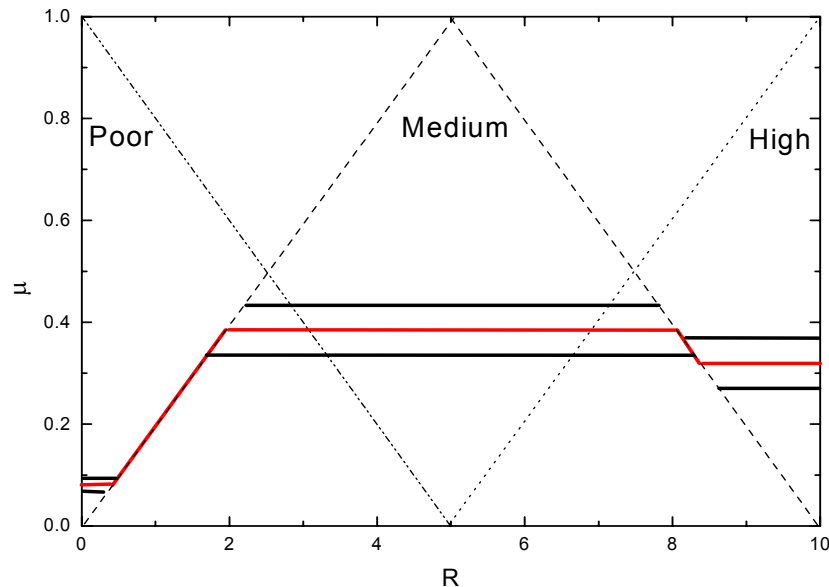


Figure 6: Integrated fuzzy output (bold lines) of R_{AVG} (red line) and its uncertainties (black lines).

3.4 Design of Web Shopping Expert

It is crucially important to represent constraints and options precisely for a decision support system. In the real world, a word can have different meaning for different people. Asking a same question to different people, the answers will be different. This difference leads to rules to be uncertain that type-1 fuzzy sets are unable to represent the users' constraints and options precisely. In order to handle these uncertainties of words, type-2 fuzzy sets are required.

A consumer decision support system (Web Shopping Expert) is designed and developed as an application of the interval type-2 fuzzy logic system. This decision support system is based on the interval type-2 fuzzy set to handle the higher level uncertainties of consumers' queries. The structure of this interval type-2 decision support system uses a similar structure of a type-2 FLS. It is shown in Figure 7 [19]. By separating computations of μ_{avg} and $\Delta\mu$, and using the error analysis method, this interval type-2 FLS can be reduced to type-1 FLS.

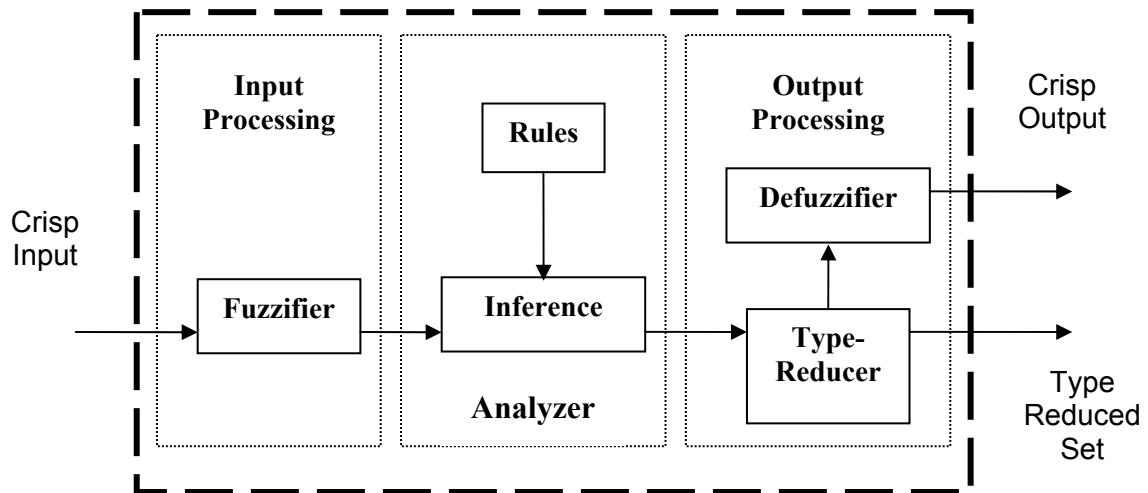


Figure 7. The structure of interval type-2 fuzzy logic systems

There are four input linguistic variables P (price), L (location), Y (years), and D (distance), and one output linguistic variable S (satisfaction) in the fuzzy inference system of the Web Shopping Expert. P has three linguistic values of low, fair, and high. L has three linguistic values of bad, fair, and good. Y has two linguistic values of old and new. D has two linguistic values of far and close. All of them are defined by their own membership functions in triangular shapes. The linguistic variable P has a type-2 membership functions and others have type-1 membership functions. This is because the standard of price can be different for different people (e.g. professional people and students). Also, it could be very difficult if you try to find an exactly \$200,000 house, but you may find many houses priced around \$200,000. Of course, location has different meaning to different people as discussed in the example of house hunting. The similar consideration can be applied on Y (years) and D (distance). For simplicity, however, the membership functions of L, Y, and D are defined as type-1 fuzzy sets here. The output variable S has three linguistic values of low, medium, and high defined by a type-1 triangular fuzzy set too.

3.5 Structure and parameter optimizations

Given a FLS, how can we define the membership functions of the linguistic variables and construct the fuzzy rule base that will result in the best performance? Since there is no general rule or method for the fuzzy logic setup, although some heuristic and iterative procedures for optimizing rules and altering the membership functions to improve performance have been proposed, several important problems in the development of fuzzy logic system still remain, such as the selection of the fuzzy rule base, the subjective definitions of the membership functions, and the selection of the variables of the system [15]. To obtain the best performance of a FLS, the different parts of the FLS have to be optimized.

The system optimizations can be divided into two categories: the structure optimization and the parameter optimization [9]. Structure optimization requires determining the fuzzy partition of input-output variables, and the set of rules to be used to generate mapping between input and output. Parameter optimization, which are tuned on the experimental data through optimization procedures, are associated with the membership functions of input-output variables or, in other words, with the locations of their fuzzy partition. Theoretically, both partitions and inference rules can be derived by the expert knowledge, but such information may be very poor, irregular, and unstructured [21]. In practice, it prevents from defining the optimal form of the mapping, where by optimal we mean that mapping is of minimal complexity, but able to capture all of the significant features of the system dynamics. For these reasons, learning methods that automatically generate the fuzzy systems from the data samples are introduced.

Structure optimization for fuzzy logic systems is very important for complexity reduction and performance enhancement. In a conventional fuzzy system, each rule contains all the input variables in its premise. It is found that such a system is hard to simplify and the system

performance is not satisfactory [2]. In addition, as the number of variables increases in the rule base, it becomes harder and harder for people to understand and interpret the rule. It has been shown that optimization of the rule structure cannot only reduce the rule complexity and improve the system performance, but also reveal the dependencies between the system inputs and the system output [10].

A common problem concerning adjustment of the membership parameters is that the shape of the membership functions is adjusted so drastically that either some of the fuzzy subsets lose their corresponding physical meanings, or the fuzzy subsets do not cover the whole space of the input variable [29]. In the latter case, the fuzzy partitioning is called incomplete, i.e., the fuzzy system takes no action if the value of the variable falls in the uncovered region. Besides, no sufficient research work has been carried out to keep the consistency of the fuzzy rules in generating fuzzy rules from data [14]. In most cases, only the rules that have the same antecedent but different consequent are considered to be inconsistent.

In this thesis I assume that the FLS uses correlation product inference, fit values are combined with the min operator, and the input and output membership functions are (possibly asymmetric) triangles. The initial rule base and some initial membership functions are given and constructed on the basis of experience. The generation of rule bases is a difficult and important task in the construction of fuzzy logic systems. By constraining the membership functions to a specific shape then each membership function can be parameterized by a few variables and the membership optimization problem can be reduced to a parameter optimization problem. The parameter optimization problem can then be formulated as a nonlinear filtering problem [17]. The parameter optimization of the fuzzy membership functions can be viewed as a weighted least-squares minimization problem, where the least square error is the difference between the

fuzzy system outputs and the target values for those outputs. The basic idea behind these learning techniques is very simple. These techniques provide a method for the fuzzy modeling procedure to learn information about a data set, in order to compute the membership function parameters that allow the associated fuzzy inference system to track the given input-output data [29].

4. PARAMETER OPTIMIZATION USING LEAST SQUARE METHOD

The least square method is a mathematical procedure for finding the best-fitting curve to a given set of points by minimizing the sum of the squares of the offsets (the residuals) of the points from the curve [30]. In optimizing the parameters of fuzzy logic systems, the least square technique is used to minimize the sum of the matching error between the expected results with the output of the fuzzy logic systems.

4.1 Basic concepts of least square method

The least square technique assumes that the best-fit curve of a given type data is the curve that has the minimal sum of the deviations squared from the given set of data [31].

Suppose that there is a set of N pairs of observations $\{Y_i, x_i\}$ where x_i is the independent variable and Y_i is the corresponding dependent variable. The fitting curve $f(x)$ has the deviation (Δ) from each data point, i.e., $\Delta_1 = Y_1 - f(x_1)$, $\Delta_2 = Y_2 - f(x_2)$, ..., $\Delta_N = Y_N - f(x_N)$. According to the linear least square method, the best fitting curve has the property that [30]:

$$\chi^2 = \Delta_1^2 + \Delta_2^2 + \dots + \Delta_N^2 = \sum_i^N \Delta_i^2 = \sum_i^N [Y_i - f(x_i)]^2 = \text{Minimum}. \quad (12)$$

In the simplest case, one variable and a linear function, the prediction is given by the following equation:

$$f(x) = a + bx. \quad (13)$$

This equation involves two free parameters which specify the intercept, a , and the slope, b , of the regression line. The least square method defines the estimate of these parameters as the values which minimize the sum of the squares. This amounts to minimizing the expression:

$$\chi^2 = \sum_i [Y_i - (a + bx_i)]^2. \quad (14)$$

This is achieved using standard techniques from calculus, namely the property that a quadratic (i.e., with a square) formula reaches its minimum value when its derivatives vanish. Taking the derivative of χ^2 with respect to a and b and setting them to zero give the following set of equations (called the normal equations):

$$\frac{\partial \chi^2}{\partial a} = 2Na + 2b \sum_i x_i - 2 \sum_i Y_i = 0, \quad (15)$$

and

$$\frac{\partial \chi^2}{\partial b} = 2b \sum_i x_i^2 + 2a \sum_i x_i - 2 \sum_i x_i Y_i = 0. \quad (16)$$

Solving these two equations gives the least square estimates of a and b as:

$$a = \bar{Y} - b\bar{x}, \quad (17)$$

with \bar{x} and \bar{Y} denoting the means of x and Y , respectively, and

$$b = \frac{\sum_i (Y_i - \bar{Y})(x_i - \bar{x})}{\sum_i (x_i - \bar{x})^2}. \quad (18)$$

The general form of this kind of model is [31]

$$y(x) = \sum_i^n a_i f_i(x) \quad (19)$$

where $f_1(x), \dots, f_n(x)$ are arbitrary fixed functions of x , called the basis functions. Note that the functions $f_i(x)$ can be nonlinear functions of x . In this discussion “linear” refers only to the model’s dependence on its parameters, a_i . The least square error is defined as

$$\chi^2 = \sum_i^n \left[\frac{Y_i - \sum_k a_k f_k(x_i)}{\sigma_i} \right]^2, \quad (21)$$

where σ_i is the measurement error (standard deviation) of the i th data point. If the measurement errors are not known, they can all be set to a constant value $\sigma_i = 1$ [31].

When the data come from different sub-populations for which an independent estimate of the error variance is available, a better estimate than the linear least square can be obtained using weighted least squares [31]. The idea is to assign to each observation a weight that reflects the uncertainty of the measurement. In general, the weight w_i , assigned to the i th observation, will be a function of the variance of this observation. A straightforward weighting schema is to define $w_i = \sigma_i^{-2}$. Then,

$$\chi^2 = \sum_i^n w_i \left[Y_i - \sum_k a_k f_k(x_i) \right]^2. \quad (22)$$

If the dependent variable Y is a function of more than one variable, a *vector* of variables \mathbf{x} , then the basis functions will be functions of a vector, $f_1(\mathbf{x}), \dots, f_n(\mathbf{x})$. The χ^2 merit function is now [30]

$$\chi^2 = \sum_i^n \left[\frac{Y_i - \sum_k a_k f_k(\bar{x}_i)}{\sigma_i} \right]^2. \quad (23)$$

All of the preceding discussion goes through unchanged, with x replaced by \mathbf{x} [30].

4.2 Tuning fuzzy membership function

In this thesis, equation (23) is used due to the multi-independent variables. Because the σ_i is not known, they are all set to a constant value of 1. In the Web Shopping Expert, the rule consequence is defined by the fuzzy sets and has the following structure:

$$R_i: \text{If } x_1 \text{ is } A_1, x_2 \text{ is } A_2, \dots, x_n \text{ is } A_n \text{ then } y = B_i . \quad (24)$$

For a set of input, the corresponding output of the system is defined by

$$y^* = \frac{\sum_{i=1}^n \mu_i B_i}{\sum_{i=1}^n \mu_i} = \sum \bar{\mu}_i , \quad (25)$$

where R_i is the i th fuzzy rule, x_i is input variable, A_i is a membership function of fuzzy sets, B_i is a constant, n is the number of the fuzzy rules, y^* is the inferred value, μ_i is the premise fitness of R_i and $\bar{\mu}_i$ is the normalized premise fitness of μ_i .

Hence, the optimal consequence parameters of the membership functions that minimize χ^2 , can be determined in consequence parameter identification, using the least-square method. χ^2 is the criterion that uses the mean squared differences between the output data of an original system (expected) and the output data obtained from the FLS. It can be defined by equation (26):

$$\chi^2 = \sum_i^n \left[\frac{y_i^* - y_i^0}{\sigma_i} \right]^2 , \quad (26)$$

where y^0 is the expected value of the output.

Assume a FLS has n input variables and one output variable. The i^{th} ($i = 1, 2, \dots, n$) variable has m_i linguistic values. All membership functions of these variables are defined as either type-1 or interval type-2 with triangular shape. Hence, the membership function of the i^{th} variable can be determined by the low and high bounds, L_i and H_i , inner points $P_{i,j}$ ($j = 1, 2, \dots, m_i - 2$) and interval range $\Delta\mu_i$ ($\Delta\mu_i = 0$ for type-1) of the i^{th} variable, where L_i and H_i can be any values but fixed, $P_{i,j}$ and $\Delta\mu_i$ are adjustable variables in training processes. Figure 8 shows the membership function of the k^{th} variable with $m_k = 3$. With well defined Mamdani-Assilian fuzzy rules by Equation (24), the general training algorithm using the least square will be:

Begin

1. Read training data
2. Type-reduce for type-2 membership functions
3. Select tuning parameters, $P_{1,1}, P_{1,2}, \dots, P_{i,j}, \dots, P_{n,(m-2)}, \Delta\mu_1, \Delta\mu_2, \dots, \Delta\mu_i, \dots, \Delta\mu_n$
4. Initialize tuning parameters and χ^2
5. Determine appropriate rules to fire using Equation (24)
6. Calculate μ_{AVG} using prod and max operators defined in type-1 fuzzy sets
7. Calculate average output using the method of the centroid of area using Equation (25)
8. Compute squared deviation
9. Calculate $\chi^2(P_{1,1}, P_{1,2}, \dots, \Delta\mu_1, \Delta\mu_2, \dots)$ using Equation (26)
10. Adjust the tuning parameters ($P_{i,j}, \Delta\mu_i$) to find minimum $\chi^2(P_{1,1}, P_{1,2}, \dots, \Delta\mu_1, \Delta\mu_2, \dots)$
11. Output tuning results

End

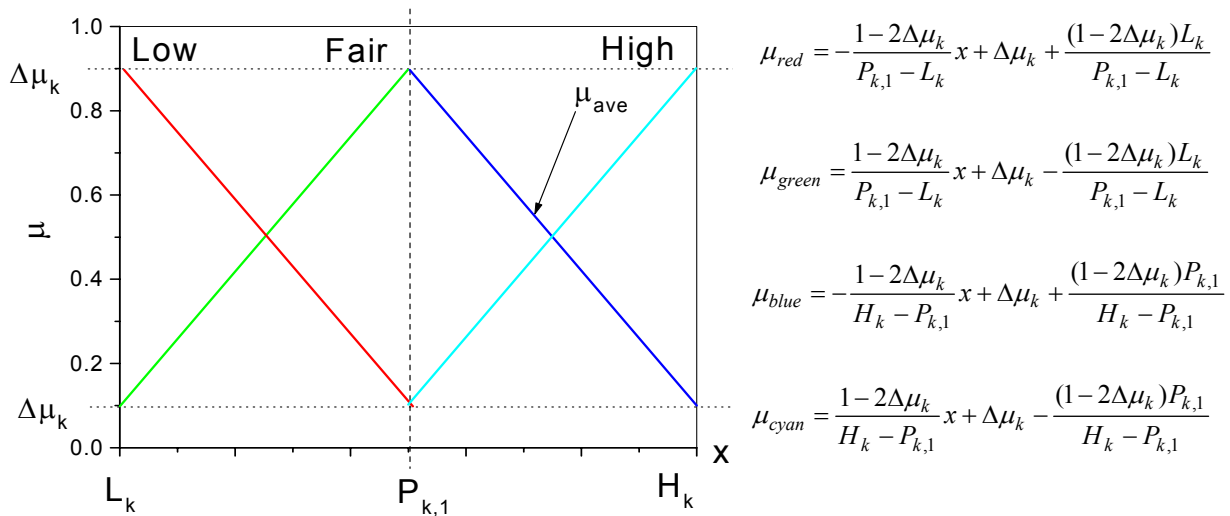


Figure 8. The membership function of the k^{th} variable.

Since the input and output variables and parameters of their membership functions of the Web Shopping Expert are given and initialized, as shown in Figure 9, the relationship between inputs and output can be mapped through the fuzzy rules and membership functions. The

location, year, and distance have the type-1 membership functions while the price has the interval type-2 membership function but is reduced to type-1 by the type reduce method discussed in section 3. The output of the system is also defined by the fuzzy rules and membership functions of the fuzzy logic system.

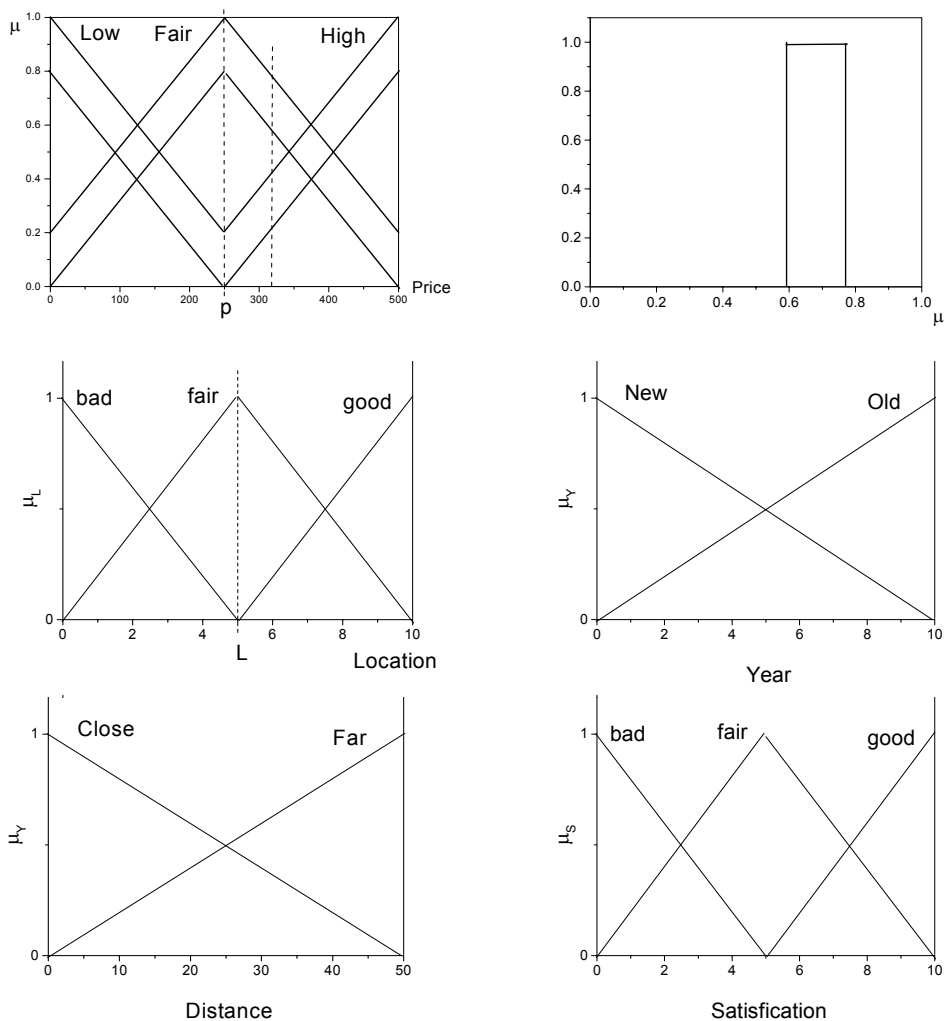


Figure 9. The type-1 membership functions of the input and output variables

For simplification, the membership function of years and distance are fixed. Three parameters ($\Delta\mu_p$, P_p , and P_l) which define the inner points of the price and location can be adjusted. To avoid the local minimum problem, a numerical method is applied to search whole space and to find the true minimum. In the optimized process ten data sets are used as the

training data as presented in Table 1. The result of optimization shows that the parameter of membership function of price changes from 250 to 180, the price interval $\Delta\mu_p$ varies from 0.05 to 0.08, and the location shifts from 5 to 5.1. Figure 10 presents the new (solid red line) and old (dash black line) membership functions of price (a) and location (b). Note that the type-2 membership functions of the average price are different from those of type-1 because of the price

Table 1. Training data and results

No	Address	Price	Location	Year	Distance	S(Exp.)	S(type-1)	S(type-2)
1	101 1 st Ave.	230	7	5	15	7	7.2	7.3
2	123 Lakeside Rd.	450	9	2	30	6	4.3	5.4
3	204 7 th St.	310	8	1	15	8	7.5	7.7
4	226 Wood St.	400	8	3	20	7	6.9	6.9
5	228 Wood St.	350	8	6	10	8	8.2	8.3
6	402 North Rd.	150	5	6	25	6	5.8	5.8
7	533 3rd Ave.	180	6	8	15	6	6.9	6.3
8	666 Main St.	290	7	1	10	7	7.4	7.0
9	705 Main St.	260	7	7	10	8	7.8	8.1
10	717 South Rd.	200	5	5	5	7	6.4	6.8

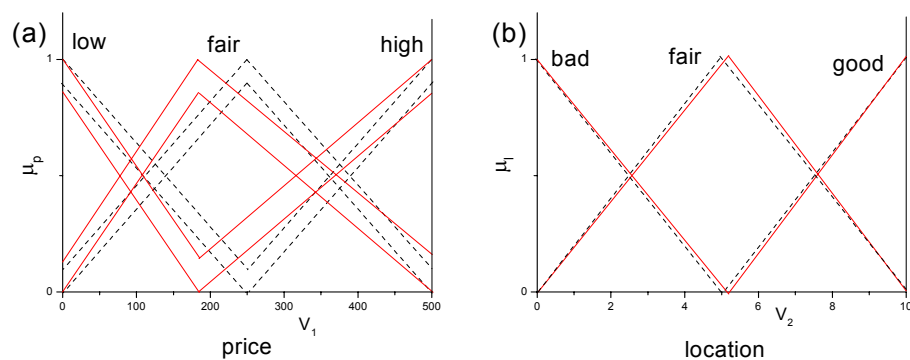


Figure 10. The membership functions of the price and location before (dash line) and after (solid line) optimization.

interval. When the price interval equals to zero, they are same and type-2 becomes type-1. The training results show that type-2 generally gives better calculated values than type-1 does, as shown in Figure 11.

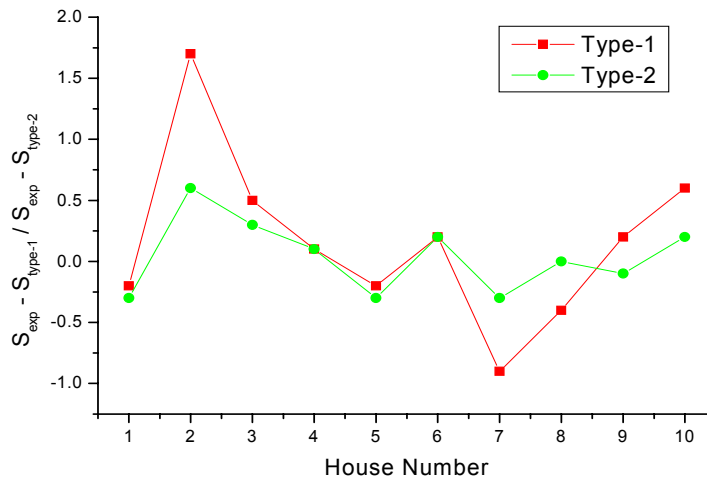


Figure 11. The errors of type-1 and type-2 after training.

4.3 Performance Comparison

One of the most important considerations in designing any fuzzy system is the generation of the fuzzy rules as well as the membership functions for each fuzzy set. In most existing applications, the fuzzy rules are generated by experts in the area. With increasing number of variables, the possible number of rules for the system increases exponentially. This makes it difficult for experts to define a complete rule set for good system performance. Least square methods have been successfully used for prediction problems in the context of reinforcement learning [16, 31]. They are easier to implement and debug, to understand why a method succeeds or fails, to quantify the importance of each basis feature, and to engineer these features for better performance. The key advantages of least square methods are the efficient use of samples and the simplicity of the implementation.

The performances of the FLS before and after optimizing membership functions are investigated. The errors of before (ΔE_{before}) and after (ΔE_{after}) optimizing membership functions are calculated based on the differences between the expected values and the estimated values. Clearly, the errors of before optimization are much larger than those after optimization as shown in Table 2. This large error can cause a serious problem for the system output. For example, in Table 2, the expected satisfactions of house-3 and house-6 are 8 and 6, respectively, but the calculated values are 4.4 and 7.3 due to the unsuitable membership functions created by an arbitrary generation. Assume house 3 is ranked number 5 (last one on a top 5 list) on the system output list. Because of the large error caused by the unsuitable membership functions another house becomes the number 5 and house 3 is no longer in the list based on the estimated values.

Table 2. Performance before and after optimization

House #	S (Exp)	S _{before} (calc)	S _{after} (calc)	S_Range	ΔE_{before}	ΔE_{after}
1	7	5.8	7.3	0.547	1.2	0.3
2	6	4.6	5.4	0.482	1.4	0.6
3	8	4.4	7.7	0.368	3.6	0.3
4	7	5.5	6.9	0.258	1.5	0.1
5	8	7.3	8.3	0.347	0.7	0.3
6	6	7.3	5.8	0.295	1.3	0.2
7	6	7.2	6.3	0.496	1.2	0.3
8	7	5.9	7.0	0.331	1.1	0.0
9	8	6.6	8.1	0.415	1.4	0.1
10	7	7.6	6.8	0.346	0.6	0.2

4.4 A Real Case

A set of real data (laptop rank) is collected from Computer shopper magazines (issues July – October, 2005) to examine the optimization processes and performances of the FLS. Four parameters (price, CPU, memory, and display) of the laptops are used in the test, as listing in Table 3. The membership function of price is defined as an interval type-2 fuzzy set while CPU, memory, display, and satisfaction are assigned to type-1 membership functions. Figure 12 shows the membership functions before (solid lines) and after (dash lines) optimization. Table 3 also presents the calculated satisfactions before (S_{before}) and after (S_{after}) optimization for each laptop. The rank in the Table 3 is the editors' rating (0 – 10) from the Computer Shopper.

Table 3. Optimization and performance

No.	Company Model	Price (\$)	CPU (GHz)	Memory (MB)	Display (inch)	Rank	S_{before}	S_{after}	S_{interval}
1	ABS Mayhem G4	2199	2.1	1024	15.4	6.2	7.8	6.5	0.31
2	Acer Aspire As2003Lmi	1599	1.6	512	15.4	5.5	4.4	5.8	0.34
3	Acer Aspire 5000	1099	1.6	512	15.4	6.8	4.9	6.6	0.36
4	Averatec 4200	1199	1.6	512	15.4	5.6	5.3	5.8	0.24
5	Dell Inspiron 9300	2842	2	1024	17	7.8	7.0	7.4	0.41
6	Gateway M460S	1359	1.86	512	15	6.4	4.9	6.6	0.28
7	HP NC6000	1448	1.6	512	14.1	6.2	4.6	6.4	0.30
8	HP Pavillon ZD 8000	2023	2.1	1024	17	7.3	6.7	7.6	0.23
9	IBM ThinkPad T43	1999	1.86	512	14.1	6.5	6.2	6.2	0.33
10	MPC T2300	2150	1.86	512	15	6.6	5.6	6.1	0.25
11	Toshiba Satellite M45	1499	1.73	512	15.4	6.8	4.4	6.3	0.34
12	Toshiba Tecra M3	1699	1.73	512	14.1	5.9	5.1	6.5	0.32

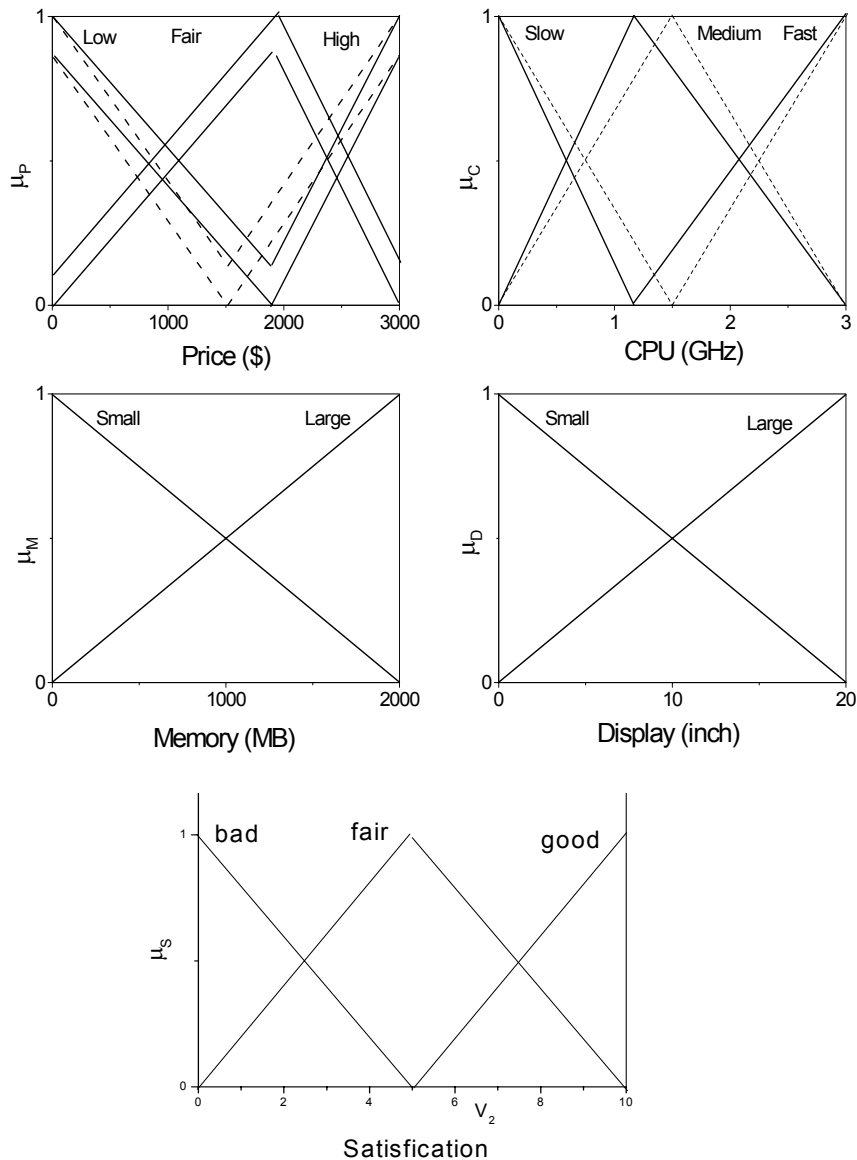


Figure 12. The membership functions of price, CUP, memory, display, and satisfaction for the laptops.

Comparison to the editors' rating (Rank), S_{after} gives much better results than S_{before} does although there are big errors for some laptops as shown in Figure 13. This is because only four parameters are used in the FLS. Actually, the performance of laptops also depends on other components such as hard drive, graphic memory, weight, and so on. In this study, those

components are ignored by assuming they are same (but actually they are not), even that the reasonable S_{after} still can be obtained. It indicates that a FLS must be optimized to achieve a better performance. The least square method provides an efficient and simple way to optimize system parameters. The satisfaction range S_{interval} offers a useful method to describe a laptop to different people. For instance, Dell Inspiron 9300 is a very good laptop to programmers but it maybe too heavy (8.2 pounds) for a traveler or too expensive (\$2,858) for a student. This kind uncertainty cannot be handled by the type-1 FLS which only gives a crisp satisfaction, S_{after} .

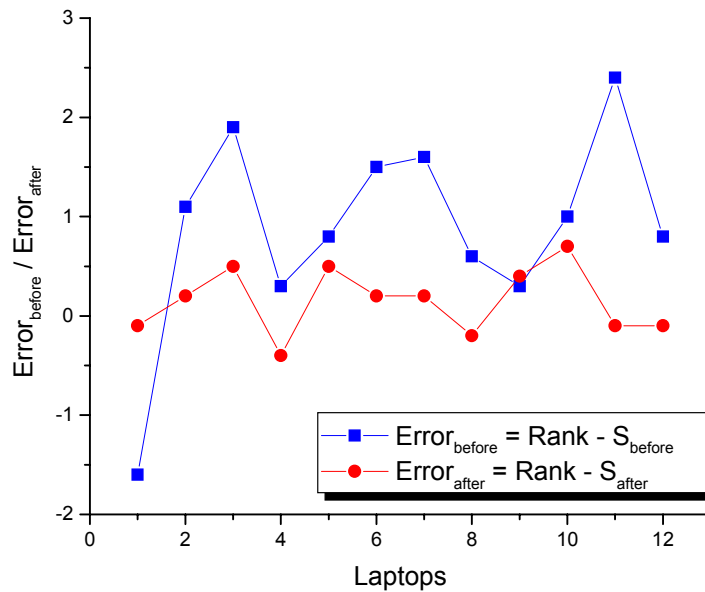


Figure 13. Errors of before and after optimization.

5. WEB SHOPPING EXPERT

Decisions are based on information. In the real world, information falls into two categories: (a) measurement-based, e.g., numerical information about distance, speed, direction and count; and (b) perception-based, e.g., information about intention, likelihood and causal dependencies. For a decision making process, it can be treated as a query formulated by multiple

criteria and constraints [32, 33]. The decision support system is a menu-oriented dialogue system for solving multi-criteria problems with crisp and fuzzy restrictions [2].

5.1 Modeling

The Web Shopping Expert is a decision support system, which helps people find their requirements (e.g. house) based on their queries and constraints. The Web Shopping Expert should have following distinguishing capabilities and functions: (1) identifies decision-relevant information; (2) translates user's queries and constraints expressed in a natural language into the generalized constraint language; (3) generates answers to queries through generalized constraint propagation; and (4) ranks decision alternatives. Therefore, the functionality of the Web Shopping Expert should primarily consist of a database system, an intelligent inference component, a training component, and a graphical user interface, as shown in Figure 1.

Database system: Database systems store all available resources and data. The database system of the Web Shopping Expert supports queries and constraints, and displays of the ranked results in as they exist in the data base form. The Web Shopping Expert supports multiple queries, analytical modeling and display of results geared towards the decision making issues.

Fuzzy inference system: To achieve the capability of multi-criteria of the Web Shopping Expert, the inference system has four inputs: price, location, year, and distance. The price has the interval type-2 fuzzy membership function to handle the higher level uncertainties of words and the others have type-1 membership functions.

User Interface: User interface of a decision support system has to comply with the requirement of interactive, iterative and participative involvement of a decision-maker. The system should provide an interface which

- Is easy to use in order to be effective.
- Should facilitate selection of parameters.
- Should be transparent to facilitate visualization of process.

5.2 Practical issues

One of the major problems of rationalizing decision processes is the large quantity of information that is vague and sometimes conflict. The capability of handling imprecise concepts is essential in decision support systems. The use of fuzzy sets to represent and manipulate both quantitative and qualitative criteria of decision-making is an important alternative for building intelligent decision support systems.

The most important considerations in designing any fuzzy system are the generation of fuzzy rules as well as membership functions for each fuzzy set. Generally, the fuzzy rules are generated by experts. With an increasing number of variables, the possible number of rules for the system increases exponentially, which makes it difficult to define a complete rule set for good system performance.

For the fuzzy logic system of the Web Shopping Expert, four practical issues are considered: (1) linguistic variables and their linguistic values, (2) membership functions, (3) rule base, and (4) system optimizations. To show the design procedures, as an example, assume that the Web Shopping Expert is used to help people find their dream houses although it can be used for other proposes or multiple proposes.

Linguistic variables and their linguistic values

There are many factors which affect a house hunter to buy a house. The most important thing for buying a house, of course, is the price of the house. House location also is a very

important factor. Old people would like their houses close to a hospital, young parents would like their houses in a good school district, some people want to live in downtown area for convenience, and some people want to live in countryside for quiet. Generally, a house condition depends on how many years of the house. For working people the distance between work place and their house is another factor, which is often considered. In this application, only these four factors are selected as the input variables of the Web Housing Expert. Other factors such as house size, structure, and yards are ignored for reducing the input variables because the possible number of fuzzy rules for a fuzzy logic system increases exponentially by increasing a number of input variables. One output variable of the Web Housing Expert is the satisfaction factor.

Basically, for a fuzzy logic system the total number of rules depends on the number of input variables and their linguistic values. The number of rules equals n^m , where n is the number of linguistic values for each input and m is the number of input variables. If a system has two input variables and each input has two linguistic values, the total number of rules is 4. When each input has three linguistic values, the total number of rules becomes to 9. To reduce the number of rules, three linguistic values (low, fair, and high) are used to present the linguistic variable price. Location also has three linguistic values (good, fair, and bad). Each of year and distance has two linguistic values (new, old) and (far, close), respectively. The output also has three linguistic values (good, fair, and poor).

Membership functions

All the membership functions used in this example are defined in the triangular shape for simplification. For the variable which has two linguistic values, its membership function can be easily defined by start and end points. For the variable which has three linguistic values, its membership function is defined by start, inner, and end points. Figure 14 shows the defined

membership functions of the input variables and their initialized values. Note that the membership functions of variables with two linguistic values are not adjustable while those with three linguistic values are adjustable. The inner point is the adjustable parameter.

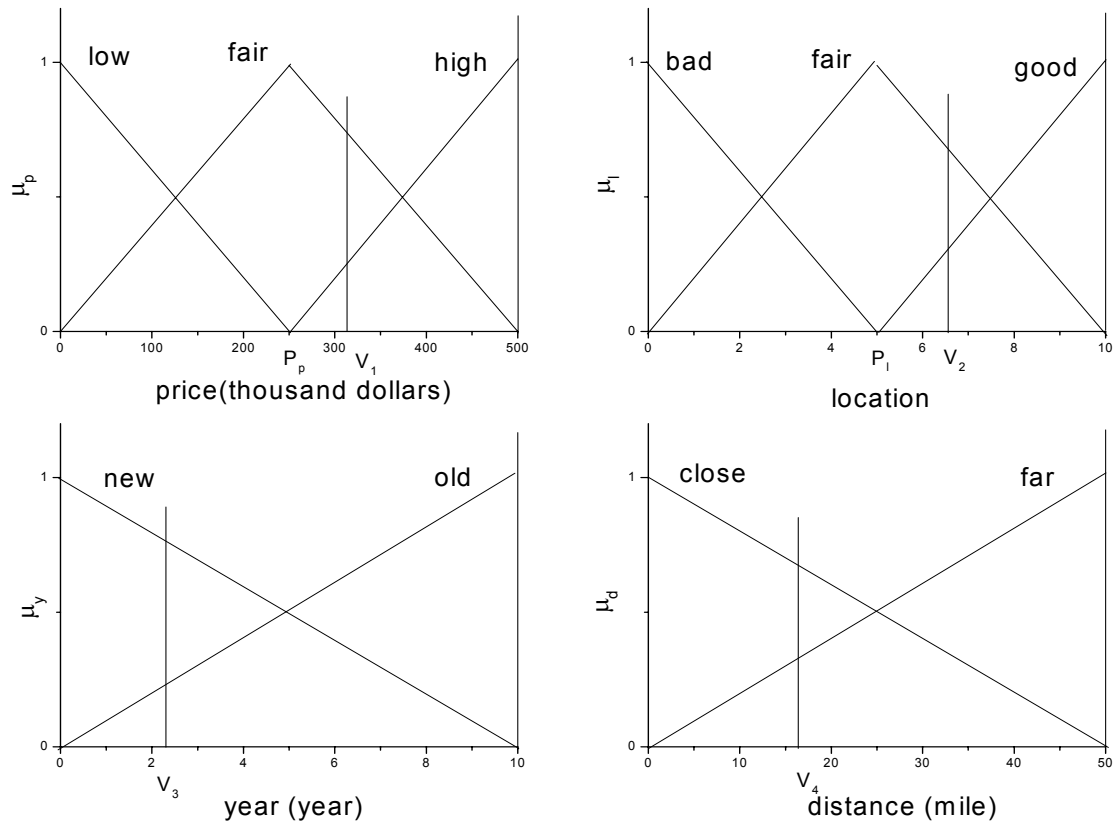


Figure 14. Membership functions and random values

Rule base

There are four input variables in the Web Housing Expert. Two of them have three linguistic values and the other two have two linguistic values. The total number of possible rules for the system is $3^2 * 2^2 = 36$. The fuzzy domain associated with each linguistic variable is shown in Figure 14. $V_1, V_2, V_3,$ and V_4 are the randomly selected examples from the database. In this case, the antecedent of the fuzzy rules is:

$$P \text{ is } \{\text{fair, high}\} \text{ and } L \text{ is } \{\text{fair, good}\} \text{ and } Y \text{ is } \{\text{new, old}\} \text{ and } D \text{ is } \{\text{close, far}\}. \quad (27)$$

Since Y and D take every one of the elements from their domains, the antecedent above in Equation (27) is equivalent to

$$P \text{ is } \{\text{fair, high}\} \text{ and } L \text{ is } \{\text{fair, good}\}. \quad (28)$$

Therefore, to reduce the number of rules, the inputs can be divided into two groups and considered separately. The first group includes price and location, and the second group contains year and distance. Tables 4 and 5 present the rules of the first and second groups, respectively. The final output is obtained by analyzing the results of these two groups and defined by another set of fuzzy rules in Table 6. Follow the analysis procedures of type-1 FLS, employ the nine fuzzy IF-THEN rules defined in Table 6, and apply operators of *prod* and *max* used in type-1 FLS for the fuzzy operation. Table 4 says, for example, if P is low and L is good then μ_1 is good. Table 5 says if Y is old and D is close then μ_2 is fair. From Table 6 we have if μ_1 is good and μ_2 is fair then μ_C is good.

Table 4. Fuzzy rules of the first group

L \ P	Low	Fair	High
Good	Good	Good	Fair
Fair	Good	Fair	Poor
Bad	Fair	Poor	Poor

Table 5. Fuzzy rules of the second group

D \ Y	New	Old
Close	Good	Fair
Far	Fair	Poor

Table 6. Fuzzy rules for combining the first and second group

$\mu_1 \backslash \mu_2$	Good	Fair	Poor
Good	Good	Good	Fair
Fair	Good	Fair	Poor
Poor	Fair	Poor	Poor

System optimizations

The system optimizations can be divided into two categories: the structure optimization and the parameter optimization. Structure optimization determines the fuzzy partition of input-output variables, and the set of rules to be used to generate mapping between input and output, as described above. Parameter optimization, which are tuned on the experimental data through optimization procedures, are associated with the membership functions of input-output variables or, in other words, with the locations of their fuzzy partition. The least square method is used to optimize the system performance by adjusting the parameters of the membership functions. Figure 15 present the algorithm of the parameter optimization procedure. The *prod* and *max* operators are applied in the calculations for satisfactions: e.g. $\mu_1 = \mu_p * \mu_i$, $\mu_2 = \mu_y * \mu_d$, $\mu_C = \mu_1 * \mu_2$, and $\mu_S = \max(\mu_{C1}, \mu_{C2}, \dots)$.

5.3 Implementation

The functionality of the Web Housing Expert consists primarily of a database system, a graphical user interface, and an intelligent inference component. A training component is also attached to the system for optimizing and updating processes.

Algorithm for tuning fuzzy membership functions

```

1. Begin
2. Define variables (P, L, Y, D, and S)
3. Define and initialize membership functions (Figure 8 and 13)
4. Define fuzzy rules (Table 4 - 6)
5. Read training data
6. Initialize  $\chi^2$ 
7. Search minimum  $\chi^2$ 
   For  $\Delta\mu = 0$  to 0.2
     For p = 100 to 400
       For l = 3 to 7
          $\chi^2(\Delta\mu, p, l) = 0$ 
         For each individual training data
           Calculate S using prod and max operators
           Update  $\chi^2(\Delta\mu, p, l)$ 
         done
         If  $\chi^2(\Delta\mu, p, l)$  smaller than  $\chi^2$  Then  $\chi^2 \leftarrow \chi^2(\Delta\mu, p, l)$ 
       done (l)
     done (p)
   done ( $\Delta\mu$ )
8. Output  $\Delta\mu, p, l, \chi^2$ 
End

```

Figure 15. The algorithm of the parameter optimization procedure.

Database: A database is a data store. It helps in managing large volumes of data with fast and easy retrieval mechanisms. The amount of information that can be stored in a database is only limited by the available physical disk space. The database system of the Web Housing Expert is developed using Microsoft Access, which is a popular relational database management system for creating desktop and client/server database applications that run under the Windows operating system. Access stores an entire database application within a single file.

Interface: A java servlet program is written to implement the user interface of the Web Housing Expert. Servlets are usually executed as part of a web server. Most major web servers, including the Netscape Web servers and Microsoft's Internet Information Server, support them. The servlets demonstrate communication between clients and servers via the HTTP protocol of the

World Wide Web. The Web Housing Expert is a web-based application. The consumer will query the system, and then the system does data processing and returns results to the consumer through the user interface. Therefore, the servlets provide a useful tool to present the communication feature between the consumer and server of this system. The interface lets users present their queries and constraints then passes them through Structured Query Language (SQL) to the program that manages the database. It returns the top five query results if there is, or reminds the user to widen the query constraints.

Fuzzy inference system: The fuzzy inference system of the Web Housing Expert is developed using interval type-2 fuzzy sets. The fuzzy inference system is based on the Mamdani model. To achieve the capability of multi-criteria, the inference system has four inputs: price, location, year, and distance. The price has the interval type-2 fuzzy membership function to handle the higher level uncertainties and the others have type-1 membership functions.

The Mamdani fuzzy logic system is a modeling strategy that can be designed by formulating a qualitative knowledge about the system behaviors. Due to the incomplete knowledge the rules and its prediction need to be trained in order to obtain the best performance. In practice the fuzzy rules are analyzed by dividing the input variables into two groups. The total number of rules is reduced from 36 to 22. The membership functions of price and location are optimized using the least square technique to achieve the best performance.

Because of the interval type-2 membership function of price, the upper-lower limit technique is used to generate a fuzzy output instead of a crisp output from the defuzzifier. It provides more flexible and reasoning results for the consumer decision support system. The use of the upper-lower limit technique also offers the opportunity to directly employ all the rules and methods of type-1 fuzzy sets that are much easier than those of type-2 fuzzy sets. The upper-

lower limit technique allows the system to have more inputs whose membership functions can be defined as either type-1 or interval type-2. For instance, we have three variables A, B, and C with errors of ΔA , ΔB , and ΔC , respectively. By the standard error theory [30]:

If $x = ABC$,

then $\Delta x = \Delta ABC + A\Delta B + AB\Delta C$;

If $y = AB + C$,

then $\Delta y = \Delta AB + A\Delta B + \Delta C$;

If $z = AB / C$,

then $\Delta z = \Delta AB / C + A\Delta B / C + AB\Delta C / C^2$.

Obviously, if A is a type-1 variable, ΔA equals zero. Accordingly, all the terms consist of ΔA equal zero too. If all the variables A, B, and C are type-1 variables, Δx , Δy , and Δz will be equal to zero. Then, the interval type-2 system is automatically converted (type-reduced) to the type-1 system. This characteristic makes system implementations much easier because we do not need to consider the types of inputs and just simply set all of them to type-2. Additionally, this characteristic also indicates that interval type-2 system is suitable for type-1 inputs and works as a type-1 system without any modification.

The Web Housing Expert works as follows:

1. Obtain user's queries through the user interface.
2. Search the database to find matched data.
3. Determine the fuzzy membership values activated by the inputs through the fuzzy inference system.
4. Decide which rules are fired in the rule set.
5. Combine the membership values for each activated rule using the AND operator.

6. Trace rule activation membership values back through the appropriate output fuzzy membership functions.
7. Utilize defuzzification to calculate the value for the output variable.
8. Rank decisions according to the output values using database system.
9. Display the output values on the user interface.

5.4 Interfaces

The Web Housing Expert is a web-based application of a new type-2 fuzzy reasoning based decision support system. It is made of four parts: home page, system page, result page, and help and contact page. The interface is very user-friendly. From the link bar located at the top of all pages, users can get anywhere at any point in the system. Figure 16 displays the home page interface, which introduces the Web Housing Expert system.



Figure 16. Web Housing Expert Home Page.

Figure 17 shows the system page, which mainly consists of a query form. There are four input variables including house price, distance from home to work place, year of house, and house location. The house location is ranged from 0 to 10. The number 10 represents the best location, and accordingly number 0 represents the worst location. For house price, the number 100 stands for \$100,000. The users can use linguistic value such as less than \$20k or between 10~15 miles and so on to fill the query form. The query form is initialized (defaulted) to show some examples. Meanwhile, the users can get help from the help link anytime. To make a request, users simply fill the form and click the “QUERY” button to submit their requests. Then the query results will be displayed in the result page as shown in Figure 18, which gives the top five ranked houses to the users. If no house matches a query, system will ask the user to widen query constraints as shown in Figure 19. Figures 20 and 21 show help page and contact page, respectively, which help the consumers use the system effectively.

The screenshot displays the 'Housing Expert' web application. The browser window title is 'Housing Expert - Microsoft Internet Explorer'. The address bar shows 'http://localhost:8080/houseexpert/Expertsystem'. The page features a blue header with the 'Housing Expert' logo and the tagline 'Help You Find Your Dream House Online'. Below the header is a navigation menu with buttons for 'Home', 'System', 'Help', and 'Contact Us'. The main content area contains a query form with the instruction 'Please select or enter the parameters appropriately as follows:'. The form includes four input fields with pre-filled values: 'Enter House Price You Wish to Buy: about \$250k', 'Enter Distance From Work to Home: less than 15 miles', 'Enter House Year You Wish to Buy: between 2~5 years', and 'Enter District Area You Like to Live: above 5'. At the bottom of the form are 'Submit' and 'Reset' buttons.

Figure 17. Web Housing Expert System Page.

Housing Expert - Microsoft Internet Explorer

Address: http://localhost:8080/Houseexpert/Houserresults

Links: Merriam-Webster OnLine

Housing Expert
Help You Find Your Dream House Online

Thursday 9/8/2005

[Home](#) [System](#) [Help](#) [Contact Us](#)

The following table shows the top 5 properties matching your search criteria:

No.	Address	Price	L	Y	D	S	R
1	120 E. Market St	260	6	2	5	7.0	0.15
2	1113 E. Lincoln St	250	6	2	8	6.8	0.15
3	88 2nd St	260	6	4	10	6.7	0.17
4	1101 main st	260	6	4	12	6.6	0.18
5	121 Chase Rd	270	6	4	10	6.4	0.18

Note:
L --- Location (1 ~ 10)
Y --- Year (yrs)

Figure 18. Web Housing Expert Searching Result Page (1).

Housing Expert - Microsoft Internet Explorer

Address: http://localhost:8080/Houseexpert/Houserresults

Links: Merriam-Webster OnLine

Housing Expert
Help You Find Your Dream House Online

Friday 9/9/2005

[Home](#) [System](#) [Help](#) [Contact Us](#)

Sorry, ☹ there are no properties on our database which match your search criteria at present. Please try widening your search !

Copyright © 2005 Housing Expert, Inc. All rights reserved.

Figure 19. Web Housing Expert Searching Result Page (2).



Figure 20. Web Housing Expert Help Interface.

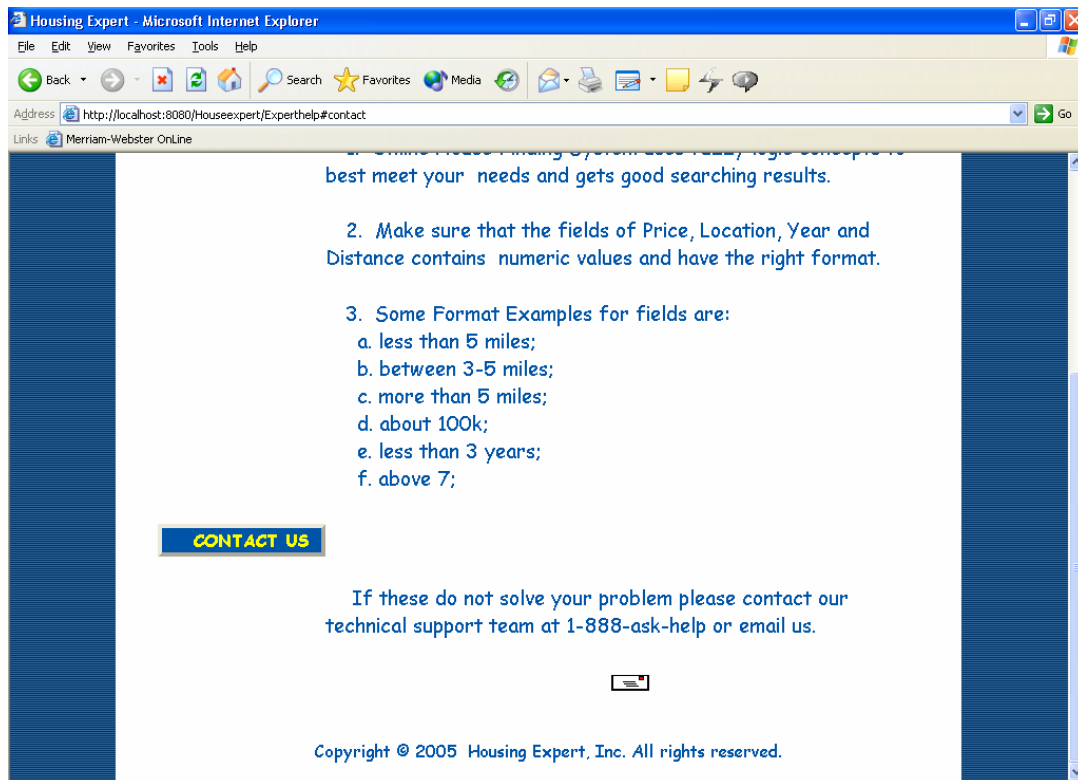


Figure 21. Web Housing Expert Contact Interface.

5.5 Discussions

A fuzzy reasoning based decision support system, the Web Housing Expert is developed. To handle the high level uncertainties due to the fuzzy nature of human languages a type-2 FLS has been employed. For reducing the complex calculations of the type-2 FLS the upper-lower limit method has been applied. A fuzzy output can be obtained using the upper-lower limit technique. Comparing with type-1 systems the Web Housing Expert provides more reasonable results based on the type-2 FLS for the different users. Figure 22 presents the calculated values of satisfaction obtained from type-1 and type-2 FLSs, see Table 2. The same membership functions except that of price, which is type-1 in the type-1 FLS and interval type-2 for the type-2 FLS are used in the calculation.

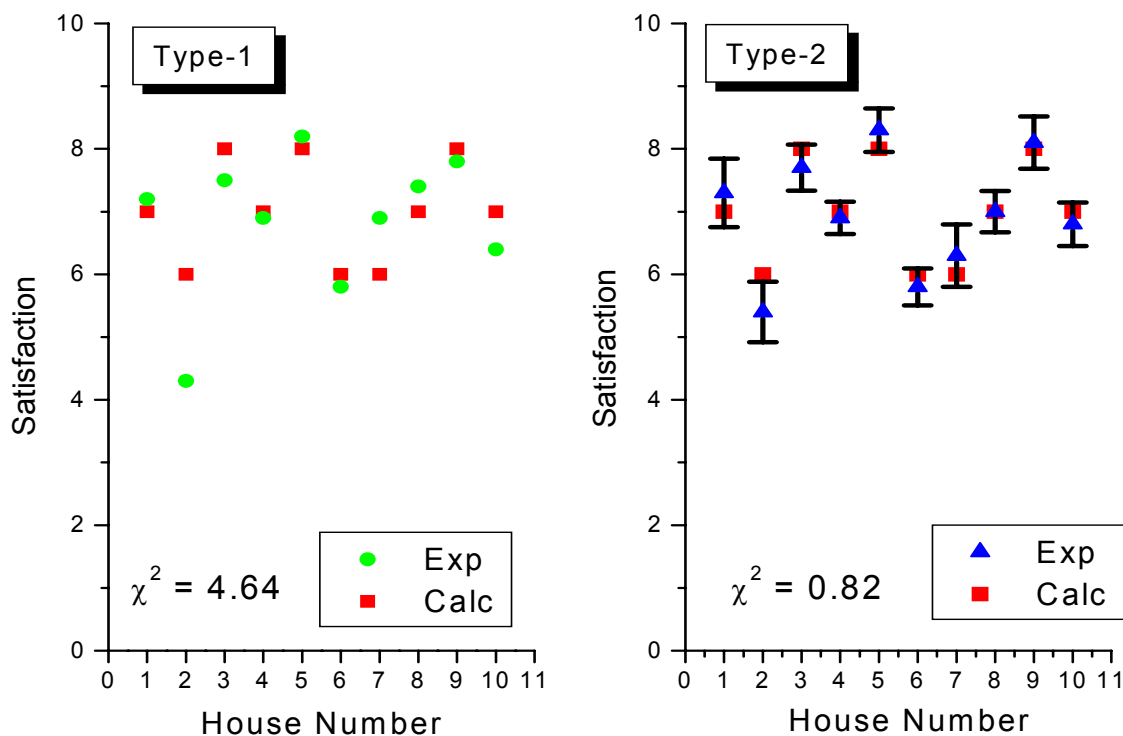


Figure 22. The expected and calculated values of satisfaction obtained from type-1 (left) and type-2 FLSs (right).

As shown in Figure 22, type-2 provides an interval satisfaction instead of the crisp output from type-1. The mean value of the interval satisfaction S_{AVG} is better (smaller χ^2) than the crisp output of the type-1 FLS because type-1 and S_{AVG} of type-2 have the different membership functions. The interval value of the type-2 membership function, $\Delta\mu$ not only affects the interval range but also change the mean value of the interval. All the expected values are covered by the interval output of type-2 FLS. Therefore, interval type-2 FLS provide a more accurate and reasonable result than type-1 FLS dose. The interval also gives more freedom to describe search results for a same query from different people because people will have different opinion to a house. For example, type-1 gives a satisfaction value of 7.5 for house 3 with the satisfaction of 8 while type-2 gives a satisfaction range of 7.33 – 8.07 (or 7.7 ± 0.37). Obviously, people could not have the same view due to the different requirements to house 1. Some people might think the price of house 1 is too high and some people might think the price of house 1 is reasonable based on their own view about the house. Therefore, they will have different satisfactions to house 1. Type-2 FLS uses the satisfaction range to represent these different opinions from different people but type-1 FLS cannot deal with this situation because it only gives a crisp satisfaction value.

The introduction of interval type-2 FLS reduces the calculation complexity of type-2 fuzzy sets. The use of the upper-lower limit method provides additional simplification of calculations by inheriting all the type-1 methods into type-2 FLS as discussed in previous chapters. Therefore, the interval type-2 FLS based Web Shopping Expert offers a flexible and effective way to represent different opinions from different users.

The system optimization is very important to achieve the best performance. In practice the fuzzy rules are analyzed by dividing the input variables into two groups and the total number

of rules is reduced from 36 to 22. The least square technique is used to optimize the membership functions of price and location. The numerical method, which scans all the parameter space, is introduced in the parameter optimization to avoid the local minimum problem. The key advantages of least square methods are the efficient use of samples and the simplicity of the implementation.

6 CONCLUSIONS

In this thesis, a new interval type-2 fuzzy reasoning based decision support system, the Web Shopping Expert for online users is proposed. This system provides a guideline for users to gain the best profit. An upper-lower limit method, which offers an opportunity to directly employ all the rules and methods of the type-1 fuzzy sets onto the interval type-2 fuzzy sets, has been introduced to handle the complex calculations of the type-2 FLS. It has been shown that a fuzzy output of an interval type-2 fuzzy sets based decision support system can be obtained using the upper-lower limit technique. Note that the interval type-2 FLS not only provides a more reasonable result but also a more accurate result than type-1 FLS does. The interval output from type-2 FLS gives more freedom to describe higher level uncertainties in human languages and the different views to a same question from different people. The type-1 FLS cannot deal with these situations because it only gives a crisp satisfaction value. The Web Shopping Expert can be used for single propose (as shown in the examples of house hunting and laptop finder) or multiple proposes.

To obtain the best performance the fuzzy inference system of the Web Shopping Expert is optimized. The system optimizations can be divided into two parts: the structure optimization and the parameter optimization. Structure optimization determines the fuzzy partition of input-

output variables, and the set of rules to be used to generate mapping between input and output. Parameter optimization, which is tuned on the training data through optimization procedures, is associated with the membership functions of input-output variables. In practice the fuzzy rules are analyzed by dividing the input variables into two groups. The total number of rules is reduced from 36 to 22. The membership functions of price and location are optimized using the least square technique to achieve the best performance. The numerical method is introduced in the parameter optimization to avoid the local minimum problem. The key advantages of least square methods are the efficient use of samples and the simplicity of the implementation. They are much easier to implement and debug. It is also easier to understand why a linear method succeeds or fails, to quantify the importance of each basis feature, and to engineer these features for better performance.

Because of the time limitation, this interval type-2 fuzzy sets based decision support system is not perfect. Some improvements can be achieved such as the use of more input variables to describe objectives more precisely, use of more type-2 variables to handle the uncertainties of words effectively, weight of different input variables to present the importance of different variables, and interval inputs to solve the problem that the precise search criteria may not work well sometimes due to the fuzzy nature of information.

REFERENCES

- [1] H.T. Mowrer, "Uncertainty in natural resource decision support systems: sources, interpretation, and importance", *Computers and Electronics in Agriculture*, **27**, 139-154 (2000).
- [2] S.J. Andriole, *Handbook of Decision Support Systems*, Tab Publishers, Blue Ridge Summit, Penn., 1989.
- [3] T.D. Phad, "Fuzzy queries, search, and decision support system," *Soft computing*, **6**, pp. 373-399 (2002).
- [4] Zadeh, L.A., "The Concept of a Linguistic Variable and its Application to Approximate Reasoning - I," *Information Sciences*, **8**, pp. 199-249 (1975).
- [5] L.A. Zadeh, "Fuzzy Logic = Computing with Words," *IEEE Trans. on Fuzzy Systems*, **4**, 103-111 (1996).
- [6] E.U. Choo, B. Schoner, and W.C. Wedley, "Interpretation of criteria weights in multi-criteria decision making". *Computers & Industrial Engineering*, **37**, 527-541 (1999).
- [7] Zhang, Y-Q., "Fuzzy Logic", *The Internet Encyclopedia*, (Editor-in-Chief: H. Bidgoli, and 14 other editors), John Wiley & Sons, Inc. Hoboken, NJ, 2003.
- [8] K. Nozaki, H. Ishibuchi, and H. Tanaka, "A simple but powerful heuristic method for generating fuzzy rules from numerical data", *Fuzzy Sets and Systems*, **86**, 251-270, 1997.
- [9] M. Figueiredo and F. Gomide, "Design of fuzzy systems using neurofuzzy networks", *IEEE Trans. Neural Networks*, **10**, 815-827, 1999.
- [10] F. Herrera and J. L. Verdegay, Eds., "*Genetic Algorithms and Soft Computing*", Berlin, Physica-Verlag, 1996.
- [11] L. X. Wang, J. M. Mendel, "Generating Fuzzy Rules By Learning From Examples," *IEEE Trans. On Systems Man and Cybernetics*, **22**, 1414-1427 (1992).
- [12] A. Grauel, I. Renners and L. A. Ludwig. "Optimizing Fuzzy Classifiers by Evolutionary Algorithms". In *Proceedings of the IEEE 4th International Conference on Knowledge-Based Intelligent Engineering Systems and Allied Technologies*, 2000.

- [13] F. Klawonn, "Fuzzy Sets and Vague Environments", *Fuzzy Sets and Systems*, **66**, 207-221, 1994.
- [14] T-P Hong and C-Y Lee, "Learning fuzzy knowledge from training examples", *Proceedings of the seventh international conference on Information and knowledge management*, 161-166, Bethesda, Maryland, 1998.
- [15] Jang, J-S. R., C-T. Sun, and E. Mizutani, "*Neuro-Fuzzy and Soft Computing*," Prentice Hall, 1997.
- [16] L.X. Wang and J.M. Mendel, "Fuzzy basis functions, universal approximation, and orthogonal least-squares learning", *IEEE Trans. on Neural Networks*, **5**, 807-814, 1992.
- [17] M. Setnes and H. Hellendoom, "Orthogonal Transforms for Ordering and Reduction of Fuzzy Rules", *Proceedings of the 9th IEEE International Conference on Fuzzy Systems*, San Antonio, USA. 700-705, 2000
- [18] R.R. Yager, "Fuzzy subsets of type II in decision," *J. Cybern.*, **10**, pp. 137-159, (1980).
- [19] Kamik, N.N., J.M. Mendel, and Q.L. Liang, "Type-2 Fuzzy Logic Systems," *IEEE Trans. on Fuzzy System*, **7**, 643-658 (1999).
- [20] L. Adelman, "*Evaluating Decision Support and Expert Systems*", John Wiley and Sons, New York, 1992.
- [21] R.T. Clemen, *Making hard decisions: An introduction to decision analysis*, Duxbury Press, Pacific Grove, CA, 1996.
- [22] R.R. Yager, "On ordered weighted averaging aggregation operators in multicriteria decision making," *IEEE Transactions on systems, man and cybernetics: Part A*, **18**, pp. 183-190 (1988).
- [23] E.H. Mamdani and S. Assilian, "An Experiment in Linguistic Synthesis with a Fuzzy Logic Controller", *International Journal of Man-Machines Studies*, **7**, 1 -13 (1975).
- [24] M. Sugeno and K. Tanaka, "Successive Identification of a Fuzzy Model and Its Applications to Prediction of a Complex System", *Fuzzy Sets and Systems*, **42**, 315-334 (1991).
- [25] Q.L. Liang and J.M. Mendel, "Interval Type-2 Fuzzy Logic System: Theory and Design," *IEEE Transaction on Fuzzy System*, **8**, 535-550 (2000).

- [26] M. Mizumoto and K. Tanaka, "Some properties of fuzzy sets of type-2," *Inform. Contr.*, **31**, 312-340 (1976).
- [27] N.N. Kamik, and J.M. Mendel, "Introduction to Type-2 Fuzzy Logic Systems," *Proc. IEEE Fuzzy Conf.*, Anchorage, AK, 915-920 (1998).
- [28] J.R. Taylor, *Introduction to Error Analysis*, 2nd Ed., University Science Books, 1997.
- [29] D.E. Goldberg, *Genetic Algorithms in search, optimization, and machine learning*, Addison-Wesley, Reading, MA, 1989.
- [30] P.R. Bevington, *Data Reduction and Error Analysis for the Physical Sciences*, McGraw-Hill, New York, 1969
- [31] C.L. Lawson and R.J. Hanson, *Solving Least Squares Problems*, SIAM, Phil., PA, 1995.
- [32] J. Casillas, O. Cordon, F. Herrera, and L. Magdalena, "Interpretability improvements to find the balance interpretability-accuracy in fuzzy modeling: an overview", in *Interpretability Issues in Fuzzy Modeling*, Springer, 2003.
- [33] L.A. Zadeh and M. Nikravesh, Perception-Based Intelligent Decision Systems, AINS; ONR Summer 2002 Program Review, UCLA.