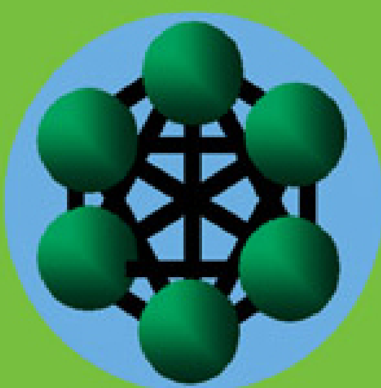
Jie Lu • Guangquan Zhang
Da Ruan • Fengjie Wu

VOL. 6

SERIES IN ELECTRICAL AND COMPUTER ENGINEERING



Multi-Objective Group Decision Making

Methods, Software and Applications with Fuzzy Set Techniques

Imperial College Press

Multi-Objective Group Decision Making

Methods, Software and Applications with Fuzzy Set Techniques

SERIES IN ELECTRICAL AND COMPUTER ENGINEERING

Editor: Wai-Kai Chen *(University of Illinois, Chicago, USA)*

Published:

Vol. 1: Net Theory and Its Applications
Flows in Networks
by W. K. Chen

Vol. 2: A Mathematical Introduction to Control Theory
by S. Engelberg

Vol. 5: Security Modeling and Analysis of Mobile Agent Systems
by Lu Ma and Jeffrey J. P. Tsai

SERIES IN ELECTRICAL AND COMPUTER ENGINEERING VOL. 6

Multi-Objective Group Decision Making

Methods, Software and Applications with Fuzzy Set Techniques

Jie Lu • Guangquan Zhang

University of Technology, Sydney, Australia

Da Ruan

Belgian Nuclear Research Centre (SCK•CEN) & Ghent University, Belgium

Fengjie Wu

University of Technology, Sydney, Australia

ICP

Imperial College Press

Published by

Imperial College Press
57 Shelton Street
Covent Garden
London WC2H 9HE

Distributed by

World Scientific Publishing Co. Pte. Ltd.
5 Toh Tuck Link, Singapore 596224
USA office: 27 Warren Street, Suite 401-402, Hackensack, NJ 07601
UK office: 57 Shelton Street, Covent Garden, London WC2H 9HE

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

Series in Electrical and Computer Engineering — Vol. 6
MULTI-OBJECTIVE GROUP DECISION MAKING
Methods, Software and Applications with Fuzzy Set Techniques

ISBN-13 978-1-86094-793-3
ISBN-10 1-86094-793-X

Printed in Singapore.

Foreword

Human activities are certainly very diverse, but one of the most important and most frequent activities is decision making. Decision making includes information gathering, data mining, modelling, and analysis. It includes formal calculus as well as subjective attitudes and it has different appearances in different situations and under different circumstances. It is, therefore, not surprising, that several scientific disciplines are concerned with this topic. Logic and Psychology, Management and Computer Sciences, Artificial Intelligence and Operations Research study this phenomenon. Since these disciplines often work independent of each other and very often without any intercommunication it is not surprising that the term '*decision*' is semantically defined differently in different disciplines and that misunderstandings occur whenever scientists from different areas discuss matters of decision making with each other. For logicians, for instance, and mathematicians a decision is the (timeless) act of selection between different alternatives of actions executed by one (abstract) person and generally guided by *one* criterion. For a sociologist or empirical decision theoretician a decision is a special, time consuming, goal-oriented information processing act, which may include one person, one organisation, or group of persons and which may be influenced by many explicit and hidden criteria and objectives.

This book focuses on one of the most complex decision making structures, in which several persons are involved in the decision making process, of which each has not only one objective function, different from the objective functions of other decision makers, but several. In addition these criteria and objectives are not dichotomous (crisp) but fuzzy, which is usually the case in reality. This represents the combination of three classical areas of decision theory: classical formal and empirical-cognitive decision theory, the theory of multi-criteria and/or multi-objectives decision making and the theory of group decision making. Part I of this book gives an introduction to all three areas. In addition this part of the book also offers an introduction to Fuzzy Set Theory and to Decision Support Systems, i.e., computer based systems, which support human decision makers in their activities.

Part II of this book combines two components of Part I, namely multi-objective decision making and fuzzy set theory and considers in more detail different models and methods in this area. In analogy to Part I these methods are then moulded in appropriate decision support systems. Building on this Part III turns to fuzzy group decision making. It first describes the methods used to solve this type of decision problems and then describes a web-based decision support system, which is especially designed for group decisions. This is certainly the most advanced type of decision technology that can be found today. This is extended in Part IV to the last stage of sophistication of decision making modelling, namely fuzzy multi-objective group decision making.

Of particular interest, not only to practitioners but also to researchers is Part V of this book: applications. A very strong motivation of decision theory has always been, not only to develop theories but to help to improve decision making in practice. The application of theories to real problems is by far not trivial but can often be one of the hardest parts of decision making or problems solving. It is, therefore, particularly valuable for the use of this book, that real applications from very different areas are described in detail. That may not only make other applications easier, but it might also facilitate the understanding of the theories and methods which are the contents of the first four parts of this book. This can only be topped by the enclosed CD, which allows readers to apply the methods themselves and solve problems that they might have or get a deeper understanding of the quite demanding theory which is described in this book.

The authors of this book can be congratulated to this exceptional work and it can only be hoped, that many researchers, students and practitioners make use of the material that is offered in this book.

Aachen, December 2006

H.-J. Zimmermann

Preface

This book presents what a multi-objective group decision-making problem is and how a decision support system can support reaching a solution in practice.

In this book, both fuzzy set theory and optimisation method are the key techniques to solve a multi-objective group decision-making problem under an uncertain environment. We offer several advantages here:

- It combines decision making theories, tools and applications effectively. For each issue of fuzzy multi-objective decision making, fuzzy multi-criteria decision making, fuzzy group decision making, multi-objective group decision-making, fuzzy multi-objective group decision-making presented in this book, we discuss their models and methods in great details with the related software systems and cases studies.
- It is designed as a unified whole in which each chapter relates its content to what went before and is, in turn, related to what will follow. Some case based examples such as product planning are discussed in different chapters for different decision situations, individual and group decision makers, and the use of different decision support systems to get desired solutions.
- It doesn't attempt to provide exhaustive coverage of every fact or research result that exists. It mainly reflects our last ten years research results in this field and what is more related, and also assumes about what the readers have already studied.
- As the technology is up-to-date throughout some results come from ours and other authors' recent publications.

Our potential readers could be organisational managers and practicing professionals, who can use the provided methods and software to solve their real decision problems; researchers in the areas of multi-objective decision making, multi-criteria decision making, group decision making, fuzzy set applications and decision support systems; students at the advanced undergraduate or master's level in management or business administration programs; and students at the advanced undergraduate or master's level in information systems and application of computer science programs.

This book is organised as follows. The first part, from Chapters 1 to 5, covers concepts and frameworks of decision making, multi-objective and multi-attribute decision making, group decision making, decision support systems, and fuzzy systems in general. Readers will learn how to model a decision problem and go through all phases of decision making process as well as the characteristics of multi-objective decision making and the components of a decision support system. The second part of the book, from Chapters 6 to 8, presents fuzzy multi-objectives decision making, including its model, several methods, and an implemented decision support system. The third part, from Chapters 9 to 11, is about group decision making within an uncertain environment. The fourth part, from Chapters 12 to 13, covers the framework, methods and systems of fuzzy multi-objective group decision making, which applies the results developed in the first two parts of the book. The last part, from Chapters 14 to 16, focuses on applications of the decision methods and systems presented in previous chapters. These applications include power market, team situation awareness and logistic management.

Most of the chapters, from Part 2 to Part 5, have real case based examples and illustrate how to use the provided decision support techniques. Within five decision support systems presented in this book, a CD-ROM included in this book has two of them, called *fuzzy multi-objective decision support system* (FMODSS) and *fuzzy group decision support system* (FGDSS). Examples illustrated in the book are mainly screenshots from using those two systems. Readers are encouraged to practice with the two systems with real world decision problems.

We wish to thank Australian Research Council (ARC) as the work presented in this book was partially supported under ARC discovery grants DP0211701, DP0557154 and DP0559213; co-workers who have advised and conducted some research results of this book with us; many researchers who have worked in multi-objective decision making, group decision making, fuzzy set application, decision support systems and

related areas over the past several decades, for which we have added their significant insight in the book and well-known publications in the reference list; the researchers and students at University of Technology Sydney (UTS) who suffered through several versions of the decision support systems shown in this book and whose comments improved it substantially; and Steven Patt, Editor at World Scientific, who helped us to ensure the book was as good as we were capable of making it.

Jie Lu, UTS, Sydney
Guangquan Zhang, UTS, Sydney
Da Ruan, SCK•CEN, Mol and UGent, Gent
Fengjie Wu, UTS, Sydney

December 2006

This page intentionally left blank

Contents

This page intentionally left blank

Part I

Decision Making, Decision Support Systems, and Fuzzy Sets

This page intentionally left blank

Chapter 1

Decision Making

This chapter presents basic concepts and methodologies of decision making, which will be used in describing fuzzy multi-objective group decision-making models, methods, systems, and applications presented in this book. We will briefly explain what the word *decision* means, what the particular characteristics of decision making are, how to model a decision problem, and what is involved in applying computerised support systems for a decision problem.

1.1 Decision and Decision Makers

Each organisation has its goals and achieves these goals through the use of resources such as people, material, money, and the performance of managerial functions such as planning, organising, directing, and controlling. To carry out these functions, managers are engaged in a continuous process of making decisions. Each decision is a reasoned choice among alternatives. The manager is thus a decision maker. However, decision makers can be managers at various levels, from a software development project manager to a CEO of a large company, and their decision problems can be various. Simple examples include deciding what to buy, when to visit a place, how to arrive there, who to employ, which grant to apply for, and deciding whom or what to vote for in an election. These problems can be in various logistics management, customer relationship management, marketing, and production planning.

Decisions can be made by individuals or groups. Individual decisions are often made at lower managerial levels and in small organisations, and group decisions are usually made at high managerial levels and large organisations. There may be conflicting preferences for a group of decision makers, and may be conflicting objectives even for a sole decision maker. For example, in a product planning decision, an individual planner may consider profit, cost, and labour satisfaction as objectives. Obviously, the three objectives here are conflict with each other. When this problem is put in a group, except the confliction among the three objectives, some members may have more concern on profit and others may be on labour satisfaction. The decision making becomes more complicated as each individual preference needs to be considered in achieving an aggregated group decision.

The decision making is more complicated and difficult because the number of available alternatives is much larger today than ever before. Due to the availability of information technology and communication systems, especially the availability of the Internet and its search engines, we can find more information quickly and therefore generating more alternatives. Second, the cost of making errors can be very large because of the complexity of operations, automation, and the chain reaction that an error can cause in many parts, in both vertical and horizontal ways, of the organisation. Third, there are continuous changes in the fluctuating environment and more uncertainties in impacting elements, including information sources and information itself. More importantly, the rapid change of the decision environment requires decisions to be made quickly. These reasons cause organisational decision makers to require increasing technical support to help making high quality decisions. A high quality decision is expected to bring, such as in bank management, greater profitability, lower costs, shortening distribution times, increasing shareholder value, attracting more new customers, or having a certain percentage of customers respond positively to a direct mail campaign.

Many standard methods can be used to classify decision problems. One of the important classifications is based on a given problem structure: structured, semi-structured, or unstructured, the latter two are also called *ill-structured*. Different classes of decision problems may require different modelling and solution methods.

In a *structured problem*, the procedures for obtaining the best or the most satisfactory solution are known by standard solution methods. In general, such problems can be described by existing classic mathematical models. For example, statistics is used to compare several products and to select one with the lowest cost.

An *unstructured problem* is fuzzy, for which there is no standard solution method. Human intuition is often the basis for decision making in an unstructured problem. Typical unstructured problems include planning new services, hiring an executive, or choosing a set of research and development projects for the next year.

Semi-structured problems fall between structured and unstructured problems, having both structured and unstructured elements. Solving them involves a combination of both standard solution procedures and human judgment.

1.2 Decision Making Process

Decision making is the cognitive process leading to the selection of a course of action among alternatives. Every decision-making process produces a final choice (sometimes called a *solution*). In general, a decision process begins when we need to find a solution but we do not know what and when a solution is accepted by decision makers. Decision making can be also seen as a reasoning process, which can be rational or irrational, and can be based on explicit assumptions or tacit assumptions.

A systematic decision-making process proposed by Simon (1977) involves three phases: *Intelligence*, *Design*, and *Choice*. A fourth phase, *Implementation*, was added later. Fig. 1.1 shows a conceptual picture of the four-phase decision-making process.

The decision making process starts with the *intelligence phase*, where the reality is examined, the problem is identified, and the problem statement is defined. In the *design phase*, a model that represents the system is constructed. This is done by making assumptions that simplify reality and by writing down the relationships among all variables. The model is then validated, and criteria are set for evaluation of the alternative courses of action that are identified. Often the process of model construction identifies potential alternative solutions, and vice versa. The *choice phase* includes selection of a proposed solution to the model. This solution is tested to determine its viability. Once the proposed solution seems to be reasonable, we are ready for the last phase: *implementation*. Successful implementation results in solving the real problem. Failure leads to a return to an earlier phase of the process.

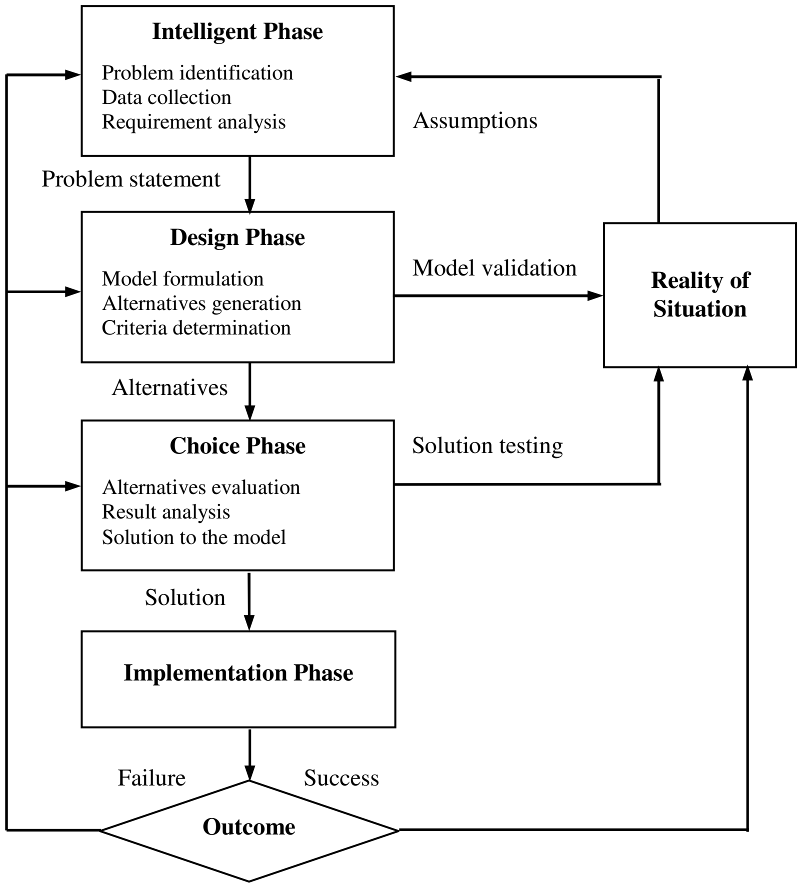


Fig. 1.1: Decision making process framework

Under the general decision process framework, different decision makers may emphasise one phase or another. Different decision-making problems may also require more details or sub-phases and support techniques in one or more phases. Literature on this subject shows many theories and results about how a decision is made, with some detailed and specific analysis and suggestions. To efficiently help decision makers understand and easily follow a decision-making process, we list nine steps as an extension of the framework in Fig. 1.1.

Step 1: Identify decision problems

To identify a decision problem includes good understanding on managerial assumptions, organisational boundaries, and any related initial and desired conditions. It aims to express the decision problem in a clear way and prepare a clear *problem statement*. This step, with Step 2 together, corresponds to the *intelligent phase* of the decision-making process framework. One example used here is to select an IT company for the development of an online consumer service (OCS) system for a business.

Step 2: Analyse requirements

Requirements are conditions in which any acceptable solution to the problem must meet. In a mathematical form, these requirements are the constraints describing the set of the feasible solutions of the decision problem. Requirements can be obtained by collecting data and analysing the decision situation. The requirements for this example include the cost and the deadline of the OCS system development, and the connection with the current business information system.

Step 3: Establish objectives and goals

The *design phase* of decision-making process starts from here and continues through to Step 6. This step identifies the important objectives of the decision problem and their goals. The objectives may be conflict but this is a natural concomitant of practical decision situations. The goals are the statements of intent and desirable programmatic values. In the mathematical form, the goals are objectives contrary to the requirements that are constraints. Not all objectives are of equal importance. Some are essential; whereas others are not absolutely necessary. For this example, the objective is to attract more customers through developing the OCS system.

Step 4: Generate alternatives

Objectives obtained will be used to help generating alternatives. But any alternative must meet the requirements. If the number of the possible alternatives is finite, we can check one by one if it meets the requirements. The infeasible ones must be deleted from the further consideration, and we obtain the explicit list of the alternatives. If the number of the possible alternatives is infinite, the set of alternatives is considered as that of the solutions fulfilling the constraints in the mathematical form of the requirements. In our example, three IT companies' responses are interested in the OCS system development and all can meet the cost and the deadline requirements, they are all as alternatives.

Step 5: Determine criteria if needed

To choose the best alternative, we need to evaluate all alternatives against objectives (Step 7). We may need some criteria to compare alternatives and to discriminate among alternatives, based on the objectives and goals. It is necessary to define discriminating criteria as objective measures of the goals to measure how well each alternative achieves the goals. In our example, to achieve the objective, to attract more customers, the OCS system developed should be user friendly, security, and easy to maintain, *etc*. This list of features can be used as criteria.

Step 6: Select a decision-making method or tool

In general, there are always several methods or tools available for solving a decision problem. The selection of an appropriate method or tool depends on the concrete decision problem and the preference of decision makers. Some methods are more suitable than others for a particular decision problem by a particular decision maker. Expertise and experience will help this selection. However, a principle is the simpler the method, the better. But complex decision problems may require more complex methods. In our example, as the decision is made in a group and linguistic terms may be used to express individual preference, a fuzzy Analytic Hierarchy Process (AHP) method (see Chapter 9) may be more suitable.

Step 7: Evaluate alternatives against criteria

The *choice phase* of decision making begins with this step. A tentative decision will be made in this step through the evaluation of the alternatives against the objectives by using the determined criteria supported by the selected method or tool. With respect to some commonly shared and understood scale of measurement and the subjective assessment of the evaluation, the selected decision-making tool can be applied to rank the alternatives or to choose a subset of the most promising alternatives. In our example, by applying the selected method, one IT company is chosen to take the development of the OCS system.

Step 8: Validate solutions against problem statements

If the tentatively chosen alternative has no significant adverse consequences, the choice is made. However, the alternatives selected by the applied decision-making method or tool have always to be validated against the requirements and goals of the decision problem. It may happen that the decision-making tool was misapplied. In complex problems the selected alternatives may also call the attention of decision makers that further goals or requirements should be added to the decision model.

Step 9: Implement the problem

This step is to use the obtained solution to the decision problem.

From the process, we can see that the decision is a choice among various alternatives. Each decision can be characterised by a problem statement, a set of alternatives, and decision criteria. Decision makers go through all these phases in the process of reaching a decision. There is no any unified description of decision-making process. But a systematic decision-making process can help ensure that all aspects of decision making receive proper consideration and lends to computerised support.

1.3 Problem Modelling and Optimisation

From the decision-making process we have found that the core of the decision process is *design phase*, which is to formulate a model for an identified decision problem. In general, different types of models will require different kinds of decision-making methods. We list here some popular decision-making models, which will be used for one phase or the whole decision-making process.

Analytic Hierarchy Process (AHP) is a decision modelling technique that allows consideration of both qualitative and quantitative aspects of decisions. It reduces complex decisions to a series of one-on-one comparisons, and then synthesises the results. To use it, a detailed description of a hierarchy diagram will be given in Section 2.5.

Paired Comparison Analysis is used for working out the importance of a number of options related to each other. This makes it easy to choose the most important problem to solve, determine more important criteria to use, or select the solution that will give the greatest advantage. It also helps decision makers set priorities where there are conflicting demands on the resources.

Grid Analysis, also known as *decision matrix analysis* or *multi-attribute utility theory*, is a technique for supporting decision making. Decision matrices are most effective in which we have many alternatives and factors (criteria) to take into account. The first step is to list decision makers' alternatives and factors (criteria). Then it will work out the relative importance (weight) of factors in the decision. The weights will be used to decision makers' preferences by the importance of the factor.

Decision Tree is a graph of decisions and their possible consequences, used to create a plan to reach a goal. A decision tree, as a special form of tree structure, is a predictive model to map observations about an item with conclusions about the item's target value. Each interior node corresponds to a variable; an arc to a child represents a possible value of that variable. A leaf represents the predicted value of the target variable given the values of the variables represented by the path from the root.

Optimisation model is a more sophisticated approach to solving decision problem. Optimisation, also called *mathematical programming*, refers to the study of decision problems in which one seeks to minimise or maximise a function by systematically choosing the values of variables from within an allowed set. An optimisation model includes three sets of elements: *decision variables*, *objective function*(*s*), and *constraint*(*s*). Many real-world decision problems can be modelled by an optimisation framework. There are many types of optimisation models such as linear programming, non-linear programming, multi-objective programming, multi-attribute programming, and multi-level programming.

Linear Programming is an important type of optimisation in which the objective function and constraints are all linear. Linear programming problems include specialised algorithms for their solution and for other types of optimisation problems by solving linear programming problems as sub-problems. Linear programming is heavily used in various management activities, either to maximise the profit or minimise the cost.

To model a decision problem by optimisation, we, in general, need three basic components: *decision variables*, *uncontrollable variables* (and/or *parameters*), and *result variables*.

Decision Variables describe alternative courses of action. The levels of these variables are determined by decision makers. For example, for a product planning problem, the amount to products produced is a decision variable.

Uncontrollable Variables or *Parameters* are the factors that affect the result variables but *are not under the control* of decision makers. Either of these factors can be fixed, in which they are called *parameters*, or they can vary, *variables*. These factors are uncontrollable because they are determined by elements of the system environment. Some of these variables limit decision makers and therefore form what are called the *constraints* of the problem. Examples are each product's produce cost, each product's marketing requirement and so on in a product planning problem.

Result Variables are outputs, reflecting the level of effectiveness of the system. The results of decisions are determined by decision makers (value of the decision variables), the factors that cannot be controlled by decision makers, and the relationships among the variables. They can be the total profit and risk, rate of return in a product planning problem.

Now we use a linear programming model to explain how to build a model for a practical decision problem. A company produces two kinds of products: A_1 and A_2. Each A_1 can yield a profit of 4000 dollars per unit, and each A_2 6000 dollars per unit. The decision problem is how many A_1 and A_2 should be produced in the first season of 2007. The objective is to obtain the maximised profit from producing the two products. However, the company has limitations in its labour, material, and marketing requirements. It needs 100 hours to produce one unit of A_1, and 200 hours to one unit of A_2, but it has only 100,000 hours labour available. The material costs of one unit of A_1 and A_2 are \$2000 and \$3000 respectively, and the total material budget is \$4,000,000. Also, it needs to produce at least 100 units of A_1 and 200 units of A_2 as marketing requirements. Within this product statement, we can determine the following:

Decision variables:

x_1 = units of A_1 to be produced;
x_2 = units of A_2 to be produced.

Result variable (*objective function*):

Maximise total profit: $z = 4{,}000\, x_1 + 6{,}000\, x_2$.

Uncontrollable variables (*constraints*):

Labour constraint: $100\, x_1 + 200\, x_2 \le 100{,}000$ (hours);
Material constraint: $2{,}000\, x_1 + 3{,}000\, x_2 \le 4{,}000{,}000$ (dollars);
Marketing requirement for A_1: $x_1 \ge 100$ (units);
Marketing requirement for A_2: $x_2 \ge 200$ (units).

This is a linear programming problem. Its formal model can be described as

$$\text{Max} \quad z = 4000x_1 + 6000x_2$$

$$\text{s.t.} \quad \begin{cases} 100x_1 + 200x_2 \le 100{,}000 \\ 2{,}000x_1 + 3{,}000x_2 \le 4{,}000{,}000 \\ x_1 \ge 100 \\ x_2 \ge 200 \end{cases}$$

By using a linear programming function of FMODSS in the attached CD, we can have the following result:

$$x_1 = 600\,(units)$$
$$x_2 = 200\,(units)$$
$$z = 3{,}600{,}000\,(dollars)$$

We can learn from this example on how to model a real-world problem. The existing decision models can help us find a way to model it and the existing decision support tools can support to generate a solution quickly.

We can find that optimisation is an ideal model for decision making. The only limitation is that it works only if the problem is structured and, for the most part, deterministic. An optimisation model defines the required input data, the desired output, and the mathematical relationships in a precise manner. Obviously, if the reality differs significantly from the assumptions used in developing the model, such a classic optimisation model cannot be used. However, a non-classical optimisation model (such as a fuzzy optimisation model) can be used.

As already discussed, many decisions are semi-structured or unstructured problems. This does not preclude using optimisation because many times a problem can be decomposed into sub-problems, some of which are structured enough for applying optimisation models. Also, optimisation can be combined with simulation and intelligent techniques, such as fuzzy logic and machine learning, for the solution of complex problems.

1.4 Computerised Decision Support

Due to the large number of elements including variables, functions, and parameters involved in many decisions, computerised decision support has become a basic requirement to assist decision makers in considering and coducting the implications of various courses of decision making. In the meantime, the impact of computer technology, particularly Internet in recent years, on organisational management is increasing. Interaction and cooperation between users and computers are rapidly growing to cover more and more aspects of organisational decision activities. Internet or Intranet-based computerised information systems have now become vital to all kinds of organisations, including businesses and governments.

Therefore computer applications in organisations are moving from transactions processing and monitoring activities to problem analysis and solution finding. Web-based services are changing from online information presentation and data access to intelligent and personalised information delivery and product customization and recommendation. Internet or intranet-based online analytical processing and real-time decision support are becoming the cornerstones of modern management within the development of e-commerce, e-business and e-government. There is a trend toward providing managers with information systems that can assist them directly in their most important task: making decisions.

Computerised decision support technologies (models, methods, and systems) can help decision making in several aspects. First, computerised system allows decision makers to perform large numbers of computations, such as complex optimisation models, very quickly. It therefore makes many complex models be used in real decision problem solving, including some emergency situations, which needs to be responded in a very short time. Second, many decision problems involve data, which is stored in different databases, data warehouses, and at websites possibly outside the organisation. Also data may have different types, such as sound and graphics, and with complex relationships. Computerised technology can search, store, and transmit needed data quickly and economically for helping decision making. Third, computerised technology can help reduce the risk of human errors and improve decision results' reliability. Fourth, computerised support technique can improve the quality of the decisions made. Using computerised support, decision makers can understand the nature of the problem better, can perform complex simulations, check many possible alternatives, and assess diverse impacts. For example, for a complex multi-objective programming problem, more alternatives can be obtained and evaluated, more uncertain data can be dealt with, more times of complex situations can be analysed and knowledge can be applied through linking with expert systems. All these capabilities lead to better decisions. Finally, computerised support can reduce decision cost. A good example is with the support of web-based systems, group members can be at different locations to have decision meetings.

The important issue is that with computerised technology, many complex decision-making tasks can be handled effectively now. However, computer based decision support techniques can be better useful in a structured decision problem than semi-structured and unstructured decision problems. In an unstructured problem only part of the problem can by supported by advanced decision support tools such as intelligent decision support systems. For semi-structured decision problems, the computerised support technology can improve the quality of the information on which the decision is based by providing not only a single solution but a range of alternative solutions. These capabilities will be further described in Chapter 4.

This page intentionally left blank

Chapter 2

Multi-Objective and Multi-Attribute Decision Making

Decisions in the real world contexts are often made in the presence of multiple, conflicting, and incommensurate criteria. Particularly, many decision problems at tactical and strategic levels, such as strategic planning problems, have to consider explicitly the models that involve multiple conflicting objectives or attributes. In this chapter, we introduce models and methods of multi-objective and multi-attribute decision making. We first present basic concepts related to criteria, objectives, and attributes used in decision making, and then introduce multi-objective decision-making models, features, and relevant methods. Following these, we will further introduce multi-attribute decision-making models and methods respectively.

2.1 Criteria, Objectives, and Attributes

Managerial problems are seldom evaluated with a single or simple goal like profit maximisation. Today's management systems are much more complex, and managers want to attain simultaneous goals, in which some of them conflict. Therefore, it is often necessary to analyse each alternative in light of its determination of each of several goals. For a profit-making company, in addition to earning money, it also wants to develop new products, provide job security to its employees, and serve the community. Managers want to satisfy the shareholders and, at the same time, enjoy high salaries and expense accounts; employees want to increase their take-home pay and benefits. When a decision is to be made, say, about an investment project, some of these goals complement each other while others conflict.

Multi-criteria decision making (MCDM) refers to making decision in the presence of multiple and conflicting criteria. Problems for MCDM may range from our daily life, such as the purchase of a car, to those affecting entire nations, as in the judicious use of money for the preservation of national security. However, even with the diversity, all the MCDM problems share the following common characteristics (Hwang and Yoon, 1981):

- *Multiple criteria*: each problem has multiple criteria, which can be objectives or attributes.
- *Conflicting among criteria*: multiple criteria conflict with each other.
- *Incommensurable unit*: criteria may have different units of measurement.
- *Design/selection*: solutions to an MCDM problem are either to design the best alternative(s) or to select the best one among previously specified finite alternatives.

There are two types of *criteria*: *objectives* and *attributes*. Therefore, the MCDM problems can be broadly classified into two categories:

- Multi-objective decision making (MODM)
- Multi-attribute decision making (MADM)

The main difference between MODM and MADM is that the former concentrates on continuous decision spaces, primarily on mathematical programming with several objective functions, the latter focuses on problems with discrete decision spaces.

For the further discussion about MODM and MADM, some basic solution concepts and terminologies are supplied by Hwang and Masud (1979) and Hwang and Yoon (1981).

Criteria are the standard of judgment or rules to test acceptability. In the MCDM literature, it indicates attributes and/or objectives. In this sense, any MCDM problem means either MODM or MADM, but is more used for MADM.

Objectives are the reflections of the desire of decision makers and indicate the direction in which decision makers want to work. An MODM problem, as a result, involves the design of alternatives that optimises or most satisfies the objectives of decision makers.

Goals are things desired by decision makers expressed in terms of a specific state in space and time. Thus, while objectives give the desired direction, goals give a desired (or target) level to achieve.

Attributes are the characteristics, qualities, or performance parameters of alternatives. An MADM problem involves the selection of the '*best*' alternative from a pool of pre-selected alternatives described in terms of their attributes.

We also need to discuss the term *alternatives* in detail. How to generate alternatives is a significant part of the process of MODM and MADM model building. In almost MODM models, the alternatives can be generated automatically by the models. In most MADM situations, however, it is necessary to generate alternatives manually. Issues on how and when to stop generating alternatives can be very important. Generating alternatives is heavily dependent on the availability and the cost of information, and requires expertise in the problem area. Alternatives can be generated with heuristics as well, and be from either individuals or groups. The generation of alternatives usually comes before the criteria for evaluating the alternatives are determined, but the selection of alternatives comes after that.

2.2 MODM Models

Multi-objective decision making is known as the continuous type of the MCDM. The main characteristics of MODM problems are that decision makers need to achieve multiple objectives while these multiple objectives are non-commensurable and conflict with each other.

An MODM model considers a vector of decision variables, objective functions, and constraints. Decision makers attempt to maximise (or minimise) the objective functions. Since this problem has rarely a unique solution, decision makers are expected to choose a solution from among the set of efficient solutions (as alternatives), which will be explained later on in this section. Generally, the MODM problem can be formulated as follows:

$$\text{(MODM)} \begin{cases} \max & f(x) \\ \text{s.t.} & x \in X = \{x \in R^n \mid g(x) \le b, x \ge 0\} \end{cases} \tag{2.2.1}$$

where $f(x)$ represents n conflicting objective functions, $g(x) \le b$ represents m constraints, and x is an n-vector of decision variables, $x \in R^n$.

Multi-objective linear programming (MOLP) is one of the most important forms to describe MODM problems, which are specified by linear objective functions that are to be maximised (or minimised) subject to a set of linear constraints. The standard form of an MOLP problem can be written as follows:

$$\text{(MOLP)} \begin{cases} \max & f(x) = Cx \\ \text{s.t.} & x \in X = \{x \in R^n \mid Ax \le b, x \ge 0\} \end{cases} \tag{2.2.2}$$

where C is a $k \times n$ objective function matrix, A is an $m \times n$ constraint matrix, b is an m-vector of right hand side, and x is an n-vector of decision variables.

We have the following notion for a *complete optimal solution*.

Definition 2.1 (Sakawa, 1993) x^* is said to be a *complete optimal solution*, if and only if there exists an $x^* \in X$ such that $f_i(x^*) \ge f_i(x)$, $i = 1, \ldots, k$, for all $x \in X$.

Also, *ideal solution*, *superior solution*, or *utopia point* are equivalent terms indicating a complete optimal solution.

In general, such a complete optimal solution that simultaneously maximises (or minimises) all objective functions does not always exist when the objective functions conflict with each other. Thus, a concept of *Pareto-optimal solution* is introduced into MOLP.

Definition 2.2 (Sakawa, 1993) x^* is said to be a *Pareto optimal solution*, if and only if there does not exist another $x \in X$ such that $f_i(x) \ge f_i(x^*)$ for all i and $f_j(x) \ne f_j(x^*)$ for at least one j.

The Pareto optimal solution is also named differently by different disciplines: *non-dominated solution*, *non-inferior solution*, *efficient solution*, and *non-dominate solution*.

In addition to the Pareto optimal solution, the following *weak Pareto optimal solution* is defined as a slight weak solution concept than the Pareto optimality.

Definition 2.3 (Sakawa, 1993) x^* is said to be a *weak Pareto optimal solution*, if and only if there does not exist another $x \in X$ such that $f_i(x) > f_i(x^*)$, $i = 1, \ldots, k$.

Here, let X^{CO}, X^P or X^{WP} denote complete optimal, Pareto optimal, or weak Pareto optimal solution sets, respectively. Then from above definitions, we can easily get the following relations:

$$X^{CO} \subseteq X^P \subseteq X^{WP} \tag{2.2.3}$$

A *satisfactory solution* is a reduced subset of the feasible set that exceeds all of the aspiration levels of each attribute. A set of satisfactory solutions is composed of acceptable alternatives. Satisfactory solutions do not need to be non-dominated. And a preferred solution is a non-dominated solution selected as the final choice through decision makers' involvement in the information processing.

This book mainly focuses on MOLP, the linear form to describe MODM. Therefore, MODM means its linear form here.

2.3 MODM Methods

2.3.1 *Classifications*

During the process of decision making, some preference information articulation from decision makers may be required, and what type of information and when it is given play a critical role in the actual decision-making method. Under this consideration, the methods for solving MODM problems have been systematically classified into four classes by Hwang and Masud (1979) and Lai and Hwang (1994) in Table 2.1.

Table 2.1: A classification of MODM methods

	Stage at which information is needed	*Type of information*	*Typical methods*
1	*No articulation of preference information*		• Global Criteria Method (Hwang and Masud, 1979, Salukvadze, 1974)
2	*A priori articulation of preference information*	Cardinal	• Weighting Method (Hwang and Masud, 1979) (Sakawa, 1993)
		Ordinal & cardinal	• Goal Programming (GP) (Ignizio, 1976)
3	*Progressive articulation of preference information (interactive method)*	Explicit trade-off	• Efficient Solution via Goal Programming (ESGP) (Ignizio, 1981) • Interactive Multiple Objective Linear Program (IMOLP) (Quaddus and Holzman, 1986) • Interactive Sequential Goal Programming (ISGP) (Hwang and Masud, 1979) • Zionts and Wallenius (ZW) (1975)
		Implicit trade-off	• STEP Method (STEM) (Benayoun *et al.*, 1971) • STEUER (1977)
4	*A posterior articulation of preference information (non-dominated solutions generation method)*	Implicit/ explicit trade-off	• Parametric method (Hwang and Masud, 1979) • Constraint method (Hwang and Masud, 1979) (Sakawa, 1993)

As shown in Table 2.1, basically, the first class of methods does not require any information from decision makers once the objective functions and constraints have been defined. The solution to an MODM problem is presented on the assumptions about decision makers' preference.

The second class of methods assumes decision makers have a set of goals to achieve and these goals will be given before formulation of a mathematical model. The goal programming (GP) assumes that decision makers can specify goals for the objective functions. The key idea behind GP is to minimise the deviations from goals or aspiration levels set by decision makers. GP therefore, in most cases, seems to yield a satisfactory solution rather than an optimisation one. By introducing the auxiliary variables, the Linear GP (LGP) problem can be converted to an equivalent linear programming problem.

The third class, interactive methods, requires more decision makers involvement in the solution process. The interaction takes place through decision makers-computer interface at the each iteration. Trade-off or preference information from decision makers at each of iterations is used for determining a new solution. Therefore, decision makers actually gain insights into the problem. The interactive programming was first initiated by Geoffrion *et al.* (1972) and further developed by many researchers. Specially, the STEP method seems to be known as one of the first interactive MOLP techniques, and there have been some modifications and extensions. The interactive GP method was also proposed (Dyer, 1972). It attempts to provide a link between GP and interactive approaches. Since then, several GP-based interactive methods that combine the attractive features from both GP and interactive approaches have been supplied.

Finally, the fourth class is just for determining a subset of the complete set of non-dominated solutions to MODM problem. It deals strictly with constraints and does not consider the preference of decision makers. The desired outcome, however, is to narrow the possible courses of actions and select the preferred course of action easier.

Interaction is one of the most important features for MODM. There are three types of interaction in the MODM process: *pre-interaction*, *pro-interaction*, and *post-interaction*. The seven MODM methods selected from Table 2.1, ESGP, IMOLP, ISGP, LGP, STEM, STEUER, and ZW, have obvious differences in interaction processes with decision makers. Table 2.2 shows the situation of the seven methods taking the three types of interaction. For example, LGP takes a pre-interaction with users before the solution process starts through collecting the weights, goals, and priorities of objectives. The IMOLP and ISGP also take a pre-interaction respectively. The method STEM takes a pro-interaction during the solution process. The principle of it is to require decision makers to give the amounts to be sacrificed of some satisfactory objectives until all objectives become satisfactory. It first displays a solution and the ideal value of each objective. It then asks decision makers to accept or reject this solution. If accepted, this solution is taken as the final satisfactory solution. However, decision makers may have different choices. They often like to further search so that more alternatives solutions can be generated. If the current solution is rejected, a relaxation process starts. Decision makers will accept a certain amount of relaxation of a satisfactory objective to allow an improvement of the unsatisfactory ones. When the relaxation fails, the system enables decision makers to continue re-entering a set of relaxation values. The second solution is then found. If decision makers accept it, it is the final satisfactory solution. Otherwise the system repeats the above process. Post-interaction is used in all these seven methods. After a set of candidate solutions has been generated, decision makers are required to choose the most satisfactory one.

Table 2.2: Types of interaction of MODM methods

Type	*ESGP*	*IMOLP*	*ISGP*	*LGP*	*STEM*	*STEUER*	*ZW*
Pre-interaction		*	*	*			
Pro-interaction	*	*	*		*		*
Post-interaction	*	*	*	*	*	*	*

'*' means '*yes*'

Decision makers have different preferences in interactive types and some decision-making problems may require a particular type of interaction. These MODM methods may be suitable for different decision makers and applications.

In the following two sub-sections, we will give more details on two typical MOLP methods: Weighting method and GP method from Table 2.1.

2.3.2 *Weighting method*

The key idea of the weighting method is to transform the multiple objectives in the MOLP (2.2.2) problem into a weighted single objective functions, which are described as follows (Kuhn and Tucker, 1951, Zadeh, 1963):

$$\begin{cases} \max \ \mathrm{w}f(x) = \sum_{i=1}^{k} w_i f_i(x) \\ \text{s.t.} \quad x \in X \end{cases} \tag{2.3.1}$$

where $w = (w_1, w_2, \ldots, w_k) \geq 0$ is a vector of weighting coefficients assigned to the objective functions.

Let us consider the following example of MOLP problem.

$$\max f(x) = \max \begin{pmatrix} f_1(x) \\ f_2(x) \end{pmatrix} = \max \begin{pmatrix} 2x_1 + x_2 \\ -x_1 + 2x_2 \end{pmatrix} \tag{2.3.2}$$

$$\text{s.t.} \quad \begin{cases} -x_1 + 3x_2 \leq 21 \\ x_1 + 3x_2 \leq 27 \\ 4x_1 + 3x_2 \leq 45 \\ 3x_1 + x_2 \leq 30 \end{cases}$$

When $w_1 = 0.5, w_2 = 0.5$, the weighting problem is formulated as

$$\begin{cases} \max \ \mathrm{w}f(x) = x_1 + 3x_2 \\ \text{s.t.} \quad x \in X \end{cases} \tag{2.3.3}$$

The optimal solution is $(x_1^*, x_2^*) = (3, 8)$, and the optimal objective function values are $f^*(x) = (f_1^*(x), f_2^*(x))^T = (14,13)^T$.

When $w_1 = 1, w_2 = 0$, the optimal solution is $(x_1^*, x_2^*) = (9, 3)$, and the optimal objective function values are $f^*(x) = (f_1^*(x), f_2^*(x))^T = (21,-3)^T$.

When $w_1 = 0, w_2 = 1$, the optimal solution is $(x_1^*, x_2^*) = (0, 7)$, and the optimal objective function values are $f^*(x) = (f_1^*(x), f_2^*(x))^T = (7,14)^T$.

2.3.3 *Goal programming*

Goal programming was originally proposed by Charnes and Cooper (1961) and has been further developed by Lee (1972), Ignizio (1976 and

1983), and Charnes and Cooper (1977). The method requests decision makers to set goals for each objective that they wish to attain. A preferred solution is then defined as the one that minimises the deviations from the goals.

Based on the MOLP model (2.2.2), some goals $g = (g_1, g_2, \ldots, g_k)^T$ are specified for objective functions $f = (f_1(x), f_2(x), \ldots, f_k(x))^T$ by decision makers, and a decision variable $x^* \in X$ in the MOLP problem is sought so that the objective functions $f^*(x) = (f_1^*(x), f_2^*(x), \ldots, f_k^*(x))^T$ are as close as possible to the goals $g = (g_1, g_2, \ldots, g_k)^T$.

The difference between $f^*(x) = (f_1^*(x), f_2^*(x), \ldots, f_k^*(x))^T$ and $g = (g_1, g_2, \ldots, g_k)^T$ is usually defined as a deviation function $D(f(x), g)$. Then the GP may be defined as an optimisation problem:

$$\begin{cases} \min D(f(x), g) \\ \text{s.t. } x \in X = \{x \in R^n \mid Ax \leq b, x \geq 0\} \end{cases} \tag{2.3.4}$$

that is, find an $x^* \in X$, which minimises $D(f(x), g)$ or

$$x^* = \arg\min_{x \in X} D(f(x), g). \tag{2.3.5}$$

Normally, the deviation function $D(f(x), g)$ is a maximum of deviation of individual goals,

$$D(f(x), g) = \max\{D_1(f_1(x), g_1), \ldots, D_k(f_k(x), g_k)\} \tag{2.3.6}$$

From (2.3.4) and (2.3.6), the minimax approach is applied to the GP problem:

$$\begin{cases} \min \max\{D_1(f_1(x), g_1), \ldots, D_k(f_k(x), g_k)\} \\ \text{s.t. } \quad x \in X = \{x \in R^n \mid Ax \leq b, x \geq 0\} \end{cases} \tag{2.3.7}$$

By introducing the auxiliary variable γ, (2.3.7) can then be transferred to the following linear programming problem:

$$\min \quad \gamma$$
$$\text{s.t.} \quad \begin{cases} D_1(f_1(x), g_1) \le \gamma \\ D_2(f_2(x), g_2) \le \gamma \\ \cdots \\ D_m(f_m(x), g_m) \le \gamma \\ Ax \le b \\ x \ge 0 \end{cases} \tag{2.3.8}$$

Let us look at the following example of MOLP problem again:

$$\max f(x) = \max \begin{pmatrix} f_1(x) \\ f_2(x) \end{pmatrix} = \max \begin{pmatrix} 2x_1 + x_2 \\ -x_1 + 2x_2 \end{pmatrix} \tag{2.3.9}$$
$$\text{s.t.} \quad \begin{cases} -x_1 + 3x_2 \le 21 \\ x_1 + 3x_2 \le 27 \\ 4x_1 + 3x_2 \le 45 \\ 3x_1 + x_2 \le 30 \end{cases}$$

Suppose the goals are specified as $g = (10, 10)^T$. The original MOLP problem can be converted as the following LP problem with the auxiliary variable γ:

$$\min \quad \gamma$$
$$\text{s.t.} \quad \begin{cases} 2x_1 + x_2 - 10 \le \gamma \\ -x_1 + 2x_2 - 10 \le \gamma \\ -x_1 + 3x_2 \le 21 \\ x_1 + 3x_2 \le 27 \\ 4x_1 + 3x_2 \le 45 \\ 3x_1 + x_2 \le 30 \\ x_1, x_2 \ge 0 \end{cases} \tag{2.3.10}$$

Then, the optimal solution is $(x_1^*, x_2^*) = (2, 6)$, and the optimal objective function values are $f^*(x) = (f_1^*(x), f_2^*(x))^T = (10, 10)^T$.

When the goals are specified as $g = (15, 15)^T$, the optimal solution is $(x_1^*, x_2^*) = (1.865, 7.622)$, and the optimal objective function values are $f^*(x) = (f_1^*(x), f_2^*(x))^T = (11.351, 13.378)^T$. From this optimal objective function value, we can find that it does not attain the goals. The reason is that the goals specified are beyond the feasible constraint area. The point of $\left(x_1^*, x_2^*\right) = (1.865, 7.622)$ is on the boundary of the feasible constraint area.

2.3.4 *A case-based example*

A manufacturing company has six machine types - milling machine, lathe, grinder, jig saw, drill press, and band saw - whose capacities are to be devoted to produce three products x_1, x_2, and x_3. Decision makers have three objectives of maximising profits, quality, and worker satisfaction. It is assumed that the parameters and the goals of the MOLP problem are defined precisely in this example. For instance, to produce one unit of x_1 needs 12 hours of milling machine, as listed in Table 2.3 (Lai, 1995).

Table 2.3: Production planning data

Machine	*Product x_1 (unit)*	*Product x_2 (unit)*	*Product x_3 (unit)*	*Machine (available hours)*
Milling machine	12	17	0	1400
Lathe	3	9	8	1000
Grinder	10	13	15	1750
Jig saw	6	0	16	1325
Drill press	0	12	7	900
Band saw	9.5	9.5	4	1075
Profits	50	100	17.5	
Quality	92	75	50	
Worker Satisfaction	25	100	75	

This problem can be described by an MOLP model as follows:

$$\max \begin{pmatrix} f_1(x) \\ f_2(x) \\ f_3(x) \end{pmatrix} = \max \begin{pmatrix} 50x_1 + 100x_2 + 17.5x_3 \\ 92x_1 + 75x_2 + 50x_3 \\ 25x_1 + 100x_2 + 75x_3 \end{pmatrix} \tag{2.3.11}$$

$$\text{s.t.} \begin{cases} g_1(x) = 12x_1 + 17x_2 \le 1400 \\ g_2(x) = 3x_1 + 9x_2 + 8x_3 \le 1000 \\ g_3(x) = 10x_1 + 13x_2 + 15x_3 \le 1750 \\ g_4(x) = 6x_1 + 16x_3 \le 1325 \\ g_5(x) = 12x_2 + 7x_3 \le 900 \\ g_6(x) = 9.5x_1 + 9.5x_2 + 4x_3 \le 1075 \\ x_1, x_2, x_3 \ge 0 \end{cases}$$

We can see that this is a typical MODM problem.

2.4 MADM Models

Multi-attribute decision making refers to making preference decision (*e.g.*, evaluation, prioritisation, and selection) over the available alternatives that are characterised by multiple, usually conflicting, attributes. The main feature of MADM is that there are usually a limited number of predetermined alternatives, which are associated with a level of the achievement of the attributes. Based on the attributes, the final decision is to be made. Also, the final selection of the alternative is made with the help of inter- and intra-attribute comparisons. The comparison may involve explicit or implicit trade-off.

Mathematically, a typical MADM (or called MCDM) problem can be modelled as follows:

$$\text{(MADM)} \begin{cases} \text{Select}: A_1, A_2, \ldots, A_m \\ \text{s.t.}: C_1, \mathrm{C}_2, \ldots, \mathrm{C}_n \end{cases} \tag{2.4.1}$$

where $A = (A_1, A_2, \ldots, A_m)$ denotes m alternatives, $C = (C_1, C_2, \ldots, C_n)$ represents n attributes (often called criteria) for characterising a decision situation. The *select* here is normally based on maximising a multi-attribute value (or utility) function elicited from the stakeholders. The basic information involved in this model can be expressed by the matrix:

$$D = \begin{array}{c} \\ A_1 \\ A_2 \\ \vdots \\ A_m \end{array} \begin{array}{c} \begin{array}{cccc} C_1 & C_2 & \cdots & C_n \end{array} \\ \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{12} & x_{22} & \cdots & x_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mn} \end{bmatrix} \end{array} \tag{2.4.2}$$

$$W = [w_1 \ w_2 \ \dots w_n]$$

where $A_1, A_2, \dots, A_m$ are alternatives from which decision makers choose; $C_1, C_2, \dots, C_n$ are attributes with which alternative performances are measured; x_{ij}, $i = 1, \dots, m$, $j = 1, \dots, n$, is the rating of alternative A_i with respective to attribute C_j; and w_j is the weight of attribute C_j.

Some critical issues of MADM are explained as follows for the later MADM method discussion.

- Quantification of qualitative ratings

An alternative in an MADM problem is usually described by some qualitative attributes. For the comparison between any two of this kind of attributes, assigning numerical values to qualitative data by scaling is the preferred approach. The Likert-type scale (Spector, 1992), which is probably the most suitable for the purposes, is described as follows.

A set of statements covering qualitative attributes is constructed. For example, the performance of an IT company for developing an E-business system can be described on a five-point scale as '*very low*,' '*low*,' '*medium*,' '*high*,' and '*very high*.' To score the scale, a five-point scale with 1, 2, 3, 4, or 5 is credited, which is corresponding from '*very low*' to '*very high*.' Sometimes, a more detailed scale such as seven-point or nine-point scale might be applied depending on the decision problem context. Since the Likert-type scale is an interval scale, the intervals between statements are meaningful but scale scores have no meaning. For example, a scale system of (3, 5, 7, 9 and 11) can be utilised instead of (1, 2, 3, 4, and 5). More examples to use this scale system are shown in Chapters 9 and 10.

- Normalisation of attribute ratings

Attribute ratings are usually normalised to eliminate computational problems caused by different measurement units in a decision matrix. It is however not always necessary but essential for many compensatory MADM methods. The procedure of normalisation aims at obtaining comparable scales, which allows inter-attribute as well as intra-attribute comparisons. Consequently, normalised ratings have dimensionless units, and the larger the rating becomes, the more preference it has. There are two popular normalisation methods used in the MADM methods:

(1) Linear normalisation

This procedure is a simple procedure that divides the ratings of a certain attribute by its maximum value. The normalised value of x_{ij} is given as

$$r_{ij} = x_{ij} \Big/ x_j^* \qquad i = 1, \ldots, m;\ j = 1, \ldots, n$$

where x_j^* is the maximum value of the jth attribute. Clearly, the attribute is more satisfactory as r_{ij} approaches 1, ($0 \le r_{ij} \le 1$).

(2) Vector normalisation

This procedure divides the ratings of each attribute by its norm, so that each normalised rating of x_{ij} can be calculated as

$$r_{ij} = \frac{x_{ij}}{\sqrt{\sum_{i=1}^{m} x_{ij}^2}} \qquad i = 1, \ldots, m;\ j = 1, \ldots, n$$

2.5 MADM Methods

Multi-attribute decision-making methods have been developed for mainly evaluating completing alternatives defined by multiple attributes. Hwang and Yoon (1981) classified 17 typical MADM methods according to the type and salient features of information received from decision makers. Furthermore, Yoon and Hwang (1995) supplied a modified taxonomy of 13 MADM methods. In this classification, methods are firstly categorised by the type of information received by decision makers. If no information is given, the *dominance method* is applicable. If information on the environment is as either pessimistic or optimistic, the *Maximin* or *Maximax* method is applicable. If information on attributes is given, a subcategory is used to further group the methods. The information given could be a standard level of each attribute, which involves *conjunctive* and *disjunctive* methods, or may be attribute weights assessed by ordinal or cardinal scales, which include *Simple Additive Weighting* method (Farmer, 1987), *TOPSIS* method (Hwang and Yoon, 1981), *ELECTRE* method (Roy, 1971), and *AHP* method (Saaty, 1980), *etc*. This section will particularly introduce two popular MADM methods, TOPSIS and AHP, in detail.

2.5.1 *TOPSIS*

Hwang and Yoon (1981) developed the *Technique for Order Preference by Similarity to Ideal Solution* (TOPSIS) method based on the concept that the chosen alternatives should have the shortest distance from the positive-ideal solution and the longest distance from the negative-ideal solution. Formally, for an MADM problem with *m* alternatives that are evaluated by *n* attributes (or called criteria), the positive-ideal solution is denoted as

$$A^* = \left(x_1^*, \ldots, x_j^*, \ldots, x_n^*\right)$$

where x_j^* is the best value for the *j*th attribute among all available alternatives. Then the negative-level solution is given as

$$A^- = \left(x_1^-, \ldots, x_j^-, \ldots, x_n^-\right)$$

where x_j^- is the worst value for the *j*th attribute among all available alternatives.

The method is presented as the following steps:

Step 1: Calculate normalised ratings

The vector normalisation is used for computing r_{ij} as

$$r_{ij} = \frac{x_{ij}}{\sqrt{\sum_{i=1}^{m} x_{ij}^2}}, \ i = 1, \ldots, m; \ j = 1, \ldots, n$$

Step 2: Calculate weighted normalised ratings

The weighted normalised value is calculated as

$$v_{ij} = w_j r_{ij}, \ i = 1, \ldots, m; \ j = 1, \ldots, n$$

where w_j is the weight of *j*th attribute.

Step 3: Identify positive-ideal and negative-ideal solutions

A^* and A^- are defined in terms of weighted normalised values:

$$A^* = \{v_1^*, \ldots, v_j^*, \ldots, v_n^*\} = \left\{ \left(\max_j v_{ij} \mid j = 1, \ldots, n \right) \mid i = 1 \ldots, m \right\}$$

$$A^- = \{v_1^-, \ldots, v_j^-, \ldots, v_n^-\} = \left\{ \left(\min_j v_{ij} \mid j = 1, \ldots, n \right) \mid i = 1 \ldots, m \right\}$$

Step 4: Calculate separation measure

The separation of each alternative from the positive-ideal solution, A^*, is given by

$$S_i^* = \sqrt{\sum_{j=1}^{n} \left(v_{ij} - v_j^*\right)^2}, \ i = 1, \ldots, m$$

Similarly, the separation from the negative-ideal solution, A^-, is given by

$$S_i^- = \sqrt{\sum_{j=1}^{n} \left(v_{ij} - v_j^-\right)^2}, \ i = 1, \ldots, m$$

Step 5: Calculate similarities to positive-ideal solution

$$C_i^* = S_i^- / \left(S_i^* + S_j^-\right), \ i = 1, \ldots, m$$

Note that $0 \le C_i^* \le 1$, where $C_i^* = 0$ when $A_i = A^-$, and $C_i^* = 1$ when $A_i = A^*$.

Step 6: Rank preference order

Choose an alternative with the maximum C_i^* or rank alternatives according to C_i^* in descending order.

2.5.2 *AHP*

The analytic hierarchy process (AHP) is essentially to formulise out the intuitive understanding of a complex problem using a hierarchical structure. The core of the AHP is to enable decision makers to structure an MADM problem in the form of an attribute hierarchy. A hierarchy has at least three levels: the focus or overall goal of the problem at the top, multiple attributes (criteria) that define alternatives in the middle, and competing alternatives at the bottom. When attributes are highly abstract, sub-attributes are generated sequentially through a multi-level hierarchy.

The AHP method has the following general steps:

Step 1: Construct a hierarchy for an MADM problem

Step 2: Make the relative importance among the attributes (criteria) by pairwise comparisons in a matrix

To help decision makers access the pairwise comparison, a Likert-type scale (for instance, nine-point scale) of importance between two elements is crested. The suggested numbers to express the degrees of preference between the two elements are shown in Table 2.4 (Yoon and Hwang, 1995). Intermediate value (2, 4, 6, and 8) can be used to represent the compromises between the preferences.

Step 3: Make pairwise comparisons of alternatives with respect to attributes (criteria) in a matrix

Step 4: Retrieve the weights of each element in the matrix generated in Steps 2 and 3

In this step, Saaty (1980) suggested the geometric mean of a row: (a) multiply the n elements in each row, take the nth root, and prepare a new column for the resulting numbers, then (b) normalise the new column (*i.e.*, divide each number by the sum of all numbers).

Step 5: Compute the contribution of each alternative to the overall goal by aggregating the resulting weights vertically

The overall priority for each alternative is obtained by summing the product of the attributes weight and the contribution of the alternative with respect to that attribute.

Table 2.4: Nine-point intensity scale for pairwise comparison

Preference on pairwise comparison	*Preference number*
Equally important	*1*
Moderately more important	*3*
Strongly more important	*5*
Very strong more important	*7*
Extremely more important	*9*

2.5.3 *A case-based example*

The following example illustrates the process of solving an MADM problem by the AHP method.

A financial company plans to develop its E-business systems and needs to select one from three IT companies as alternatives: company A_1, company A_2, and company A_3. Four attributes (criteria) that are cost (C), security (S), development period (P), and maintenance (M) are generated to evaluate these IT companies.

Step1: A hierarchy for the MADM problem is created as in Fig. 2.1

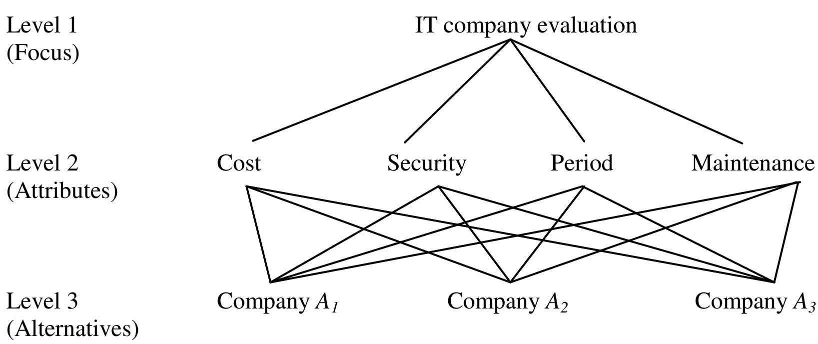


Fig. 2.1: A hierarchy for the IT company selection

Step 2: A matrix is made to express the relative importance among these attributes, that is,

$$\begin{bmatrix} 1 & C/S & C/P & C/M \\ S/C & 1 & S/P & S/M \\ P/C & P/S & 1 & P/M \\ M/C & M/S & M/P & 1 \end{bmatrix} = \begin{bmatrix} 1 & 5 & 3 & 9 \\ 1/5 & 1 & 1/5 & 2 \\ 1/3 & 5 & 1 & 7 \\ 1/9 & 1/2 & 1/7 & 1 \end{bmatrix}. \tag{2.5.1}$$

Here, '1' means '*equally important*,' and '*C/S* = 5' means that *C* (*Cost*) is '*strongly more important*' than *S* (*Security*).

Step 3: Four matrixes are made for pairwise comparisons of the three companies with respect to four attributes, that is,

$$\begin{array}{cccc} \text{For C} & \text{For S} & \text{For P} & \text{For M} \\ \begin{bmatrix} 1 & 1/7 & 3 \\ 7 & 1 & 2 \\ 1/3 & 1/2 & 1 \end{bmatrix} & \begin{bmatrix} 1 & 5 & 1/3 \\ 1/5 & 1 & 1/5 \\ 3 & 5 & 1 \end{bmatrix} & \begin{bmatrix} 1 & 3 & 5 \\ 1/3 & 1 & 7 \\ 1/5 & 1/7 & 1 \end{bmatrix} & \begin{bmatrix} 1 & 1/5 & 2 \\ 5 & 1 & 1/5 \\ 1/2 & 5 & 1 \end{bmatrix} \end{array}. \tag{2.5.2}$$

Step 4: Retrieve the weights of each element in the matrix generated in Steps 2 and 3

From (2.5.1), we have

$$\begin{bmatrix} (1\times5\times3\times9)^{1/4} = 3.41 \\ (1/5\times1\times1/5\times2)^{1/4} = 0.53 \\ (1/3\times5\times1\times7)^{1/4} = 1.85 \\ (1/9\times1/2\times1/7\times1)^{1/4} = 0.30 \end{bmatrix} = \begin{bmatrix} 0.58 \\ 0.09 \\ 0.28 \\ 0.05 \end{bmatrix}.$$

From (2.5.2), we have

$$\begin{bmatrix} 0.20 & 0.30 & 0.60 & 0.24 \\ 0.65 & 0.09 & 0.32 & 0.32 \\ 0.15 & 0.61 & 0.08 & 0.44 \end{bmatrix}.$$

Step 5: We now compute the contribution of each alternative to the overall goal:

$$\begin{bmatrix} A_1 \\ A_2 \\ A_3 \end{bmatrix} = \begin{bmatrix} 0.58\times0.20+0.09\times0.30+0.28\times0.60+0.05\times0.24 \\ 0.58\times0.65+0.09\times0.09+0.28\times0.32+0.05\times0.32 \\ 0.58\times0.15+0.09\times0.61+0.28\times0.08+0.05\times0.44 \end{bmatrix} = \begin{bmatrix} 0.3248 \\ 0.4908 \\ 0.1844 \end{bmatrix}.$$

The result shows that company A_2 has the highest score, and therefore it can be selected for the E-business system development.

2.6 Summary

Both MODM and MADM issues are the main focuses of the book. Both MODM and MADM methods will be extended to group decision making, and deal with uncertainty by fuzzy techniques. Furthermore, these fuzzy MODM and fuzzy MADM methods are built in the fuzzy multi-objective and fuzzy multi-criteria decision support systems and applied in real world applications in the rest chapters of the book.

This page intentionally left blank

Chapter 3

Group Decision Making

Group decision making is defined as a decision situation in which there are more than one individual involved. These group members have their own attitudes and motivations, recognise the existence of a common problem, and attempt to reach a collective decision. In this chapter, we will first discuss the concepts and characteristics of group decision making, and then review some popular group decision-making methods, which have been used in the development of fuzzy group decision support systems. These group decision-making techniques and their applications will be presented in Chapters 10, 11, 12, and 13.

3.1 Decision Groups

Decision making requires multiple perspectives of different people as one decision maker may have not enough knowledge to well solve a problem alone. This is particularly true when the decision environment becomes more complex. Therefore, more organisational decisions are made now in groups than ever before. These decisions could be designing products, developing policies and strategies, selecting employees, and arranging various resources. Such groups are called *decision groups*. In an organisation, a decision group is a self-regulating, self-contained task-oriented work group such as a committee. Group-based decision making has become a key component to the functioning of an organisation.

Group decision making (GDM) is the process of arriving at a judgment or a solution for a decision problem based on the input and feedback of multiple individuals. It is a group work cooperatively to achieve a satisfactory solution for the group rather than the best solution as it almost does not exist. In general, a group satisfactory solution is one that is most acceptable by the group of individuals as a whole. Since the impact of the selection of the satisfactory solution affects organisational performance, it is crucial to make the group decision-making process as efficient and effective as possible. It therefore is very important to determine what makes a decision making effective and to increase the level of overall satisfaction for the solution across the group.

We need to distinguish between *non-cooperative multi-member decision making* and *cooperative group decision making*. In the former decision-making situation, decision makers play the role of antagonists or disputants. Conflict and competition are common forms of this non-cooperative decision making. In the group decision-making environment, decision makers recognise the existence of a common problem, attempt to reach a common decision in a friendly and trusting manner, and share the responsibility. Consensus, negotiation, voting schemes, and even resource to a third party to dissolve differences are examples of this type of group decision making. Within the cooperative group decision making category, there are still two different situations. One class is under a team decision structure. For example, an individual manager has the authority to make a particular decision, but several support assistants work together with the manager toward the same goal to the decision. In contrast to the team decision structure, in a group decision structure, group members share a similar rested interest in the decision outcome and an equal say in its formation. Group members generally work in a formal environment, an example of which is an organisational committee. This book mainly focuses on this group decision structure.

3.2 Characteristics

To understand how effective support can be provided to decision makers who work in groups for making decisions for their organisations, we need to analyse the main characteristics of group decision as follows:

- The group performs a decision-making task.

- The group decision covers the whole process of transfer from generating ideas for solving a problem to implementing solutions.
- Group members may be located in the same or different places.
- Group members may work at the same or different times.
- Group members may work for the same or different departments or organisations.
- The group can be at any managerial levels.
- There can be conflict opinions in group decision process among group members.
- The decision task might have to be accomplished in a short time.
- Group members might not have complete information for decision tasks.
- Some required data, information or knowledge for a decision may be located in many sources and some may be external to the organisation.

From the above characteristics, the group members are allowed in different locations and may be working at different times. They need to communicate, collaborate, and access a diverse set of information sources, which can be met with the development of the Internet and its derivatives, intranets and extranets. The *Internet*, as the platform on which most group online communications for collaboration occur, supports the inter-organisational decision making through online group collaboration tools and access to data, information and knowledge from inside and outside the organisation. In Chapter 11, we will present a web-based group decision support system and explore these issues in depth. The *intranet***,** basically an internal Internet, can effectively support Intra-organisational networked group decision making. It allows a decision group within an organisation to work with Internet tools and procedures. An *extranet* can link a decision group like an intranet for group members from several different organisations. For example, some automobile manufacturers have involved their suppliers and dealers in extranets to help them deal with customer complaints about their products.

Another key issue for these characteristics is about information sharing in a decision group. Even in hierarchical organisations, decision making is usually a shared process within a decision group. In the decision group, group members are typically in equal or near-equal status. The outcome of the decision meeting depends not only on their knowledge, opinions, and judgments, but also on the composition of the group and the decision-making method and process used by the group. Differences in opinions are settled either by the ranking person present or, more often, by negotiation or arbitration. Although it may be too expensive for all group members to have complete information for their decision tasks, information sharing is the most important element to improve the quality of group decision.

Another related issue is about bargaining and negotiation in group decision process. A decision group should be negotiable in order to achieve a consensus-based solution. When a common decision fails, it becomes necessary for group members to start bargaining or negotiating until a consensus is reached. While bargaining involves discussions within a specific criterion or issues, negotiation includes many criteria or issues in the discussion and search for consensus.

3.3 Models

Due to the importance and complexity of group decision making, decision making models are needed to establish a systematic means of supporting effective and efficient group decision making.

There are two kinds of basic models of group decision making. The first one, the *rational model*, is grounded on objectives, alternatives, consequences, and optimality. This model assumes that complete (or most) information regarding the decision to be made is available and one correct conception of the decision can be determined. It further assumes that decision makers consistently assess the advantages and disadvantages of any alternatives with goals and objectives in mind. They then evaluate the consequences of selecting or not selecting each alternative. The alternative that provides the maximum utility (*i.e.*, the optimal choice) will be selected. Another basic decision-making model is the *political model*. In contrast to the rational model, the individuals involved do not accomplish the decision task through the rational choice in regard to organisational objectives. Decision makers are motivated by and act on their own needs and perceptions. This process involves a cycle of negotiation and discussion among group members in order, for each one, to get their perspective to be the one of choices. More specifically, this process involves all decision makers trying to sway powerful people within the situation to adopt their viewpoints and influence the remaining members.

The rational model utilises a logical and sequential approach to make group decisions by evaluating alternatives based on the information at hand and then choosing the optimal alternative. But, assumptions of this model may not be totally realistic. In real environments, group decision making has to confront many conditions. Due to different experiences and opinions for decision objectives and assessment-criteria, individuals involved in the process may bring their own perceptions and mental models into a decision situation. They may have different information at hand and share only partially overlapping goals. Therefore, information incompleteness, conflicts of interest, and inconsistence of assessing criteria are inevitable. The decision-making procedure has to be performed through negotiation and discussion among group members to individual goals, powers, or favors. The rational model is hard to handle such a situation. The political model does not involve making full information available or a focus on the optimal viewpoint. It operates based upon negotiation that is often influenced by individual powers and favors. Thus, such a model is suitable to deal with a situation where information is withheld and subsequently incomplete and individual favors are uncertain or inaccurate. But its risk is that the '*best*' solution or decision may not be selected. Furthermore, the nature of negotiation can produce effects that are long-lasting and detrimental. Once they discover it, individuals involved in the decision may not appreciate the duplicity inherent in the process. Therefore, a combination of both models could be a better way in practice, particularly, when the environment has more uncertain factors. In Chapter 10, we will present a rational-political model for group decision making in an uncertain environment, which takes advantage of both rational and political models of group decision making.

In Chapter 2, we introduced MODM and MADM methodologies and applications where they mostly address a single decision maker. When a group decision involves an MODM or MADM problem, it is called *multi-objective group decision making* (MOGDM) or *multi-attribute group decision making* (MAGDM), which is also called MCGDM in many situations. In an MOGDM, some decision makers may generate relevant objectives for the problem. Others may share some, but none or all of their objectives. When some selected objectives are accepted, the group members are allowed to use related MODM methods to arrive at a solution. The problem is no longer the design of the most preferred objective according to one individual's preference structure. The analysis must be extended to account for the conflicts and aggregation among different group members who have different preference on the objectives and different values on the goals. We will discuss this issue in depth in Chapters 12 and 13. Similarly, MAGDM decision makers within the group should agree with certain rules to follow for achieving a solution. In general, the group's decision is usually understood to be the reduction of different individual preferences among alternatives and criteria in a given set to a single collective preference or group preference. This issue will be discussed in depth in Chapter 10.

3.4 Process

Because the performance of group decision making involves taking into account the needs and opinions of group members, the ability and the process of reaching a consensus decision effectively and even efficiently are critical to the functioning of the group. There are a variety of ways to make decisions as a group. Here we only indicate the main differences between group decision making and individual decision making in the decision-making process. Comparing with the decision-making process presented in Section 1.2, the analysis presented below offers an effective structure for choosing an appropriate course of action for a particular group task.

Step 1: Define the decision problem

It is important for group decision makers to understand clearly what they are trying to decide so that they have a common goal to focus discussions on and form a problem statement.

Step 2: Determine requirements

Once group decision makers have defined their decision problem, they will examine the data and resources that they already have, and identify what additional information they may need. Discussion based on information sharing is very important.

Step 3: Establish objectives and goals

When some opinions on objectives are conflicted with each other in a group, discussion, negotiation, even a voting will be made until an agreement for objectives and goals are accepted by the group.

Step 4: Generate alternatives

Following the above requirements and objectives, we can generate alternatives for potential solutions to the problem. This involves collecting as many alternatives as possible to make sure group members participate in the generation process. But some similar alternatives proposed by different members should be merged, and a set of alternatives will be finally accepted by the group.

Step 5: Determine criteria

To identify the criteria would determine whether a chosen solution is successful. Ideally, a solution will be feasible, move the group forward, and meet the needs of group members. Similar criteria will be merged and weights may be given by all members through discussion and negotiation. The individual group member may want to rank the criteria in order of the importance, and an agreement on the weights of the criteria may be needed.

Step 6: Select a group decision-making method or tool

Based on the situation of the decision group (for example, at the same place or different locations, has a leader or not), a method or tool can be chosen.

Step 7: Evaluate alternatives and select the best one

Now, decision makers are ready to evaluate alternatives according to the criteria identified in Step 5. They may be able to combine their ideas to create a solution. Ideally, everyone would agree with a solution (a consensus), but not everyone may agree. In this case, the group will need to use a different decision-making method.

Step 8: Validate solutions

Based on the criteria identified in Step 5, this group will evaluate if the decision was successful. Failure will lead to a return to an earlier step.

Step 9: Implement the solution

This involves identifying the resources necessary to implement the group decision.

3.5 Methods

There are several kinds of decision-making methods that a group may use. In general, each kind of methods follows a rule or a principle. We briefly describe some popular ones, with their advantages and disadvantages.

- Authority rule

Most groups have a leader who has an authority to make the ultimate decision for a group. The group can generate ideas and hold open discussions, but at any time the leader may make a decision upon a given plan. The effectiveness of the kind of methods depends a great deal upon whether the leader is a sufficiently good listener to have culled the right information on which to make the decision. Obviously, the method can generate a final decision fast. But, this method does not maximise the strengths of the individuals in the group.

- Majority rule

Some group decisions are made based on a vote (maybe in an informal way) for alternatives or individual opinions following a period of discussions. The majority's opinion is as the solution of the group for the decision problem. This method can make a group decision fast, and follows a clear rule of using democratic participation in the process. But sometimes decisions made by this method are not well implemented due to an insufficient period of discussions.

- Negative minority rule

A common form of negative minority rule is that the group has a number of alternatives. It holds a vote for the most unpopular alternative and eliminates it. It then repeats this process until only one alternative is left. This is also a democratic method and will be very useful when there are many ideas and few voters. But obviously this method is slow and sometimes, group members may feel resentful at having their ideas voted as unpopular.

- Ranking rule

Several similar ranking methods have been used in practice and all assume the group has a number of alternatives. One is to let group members individually give a score to each alternative. Suppose the group has five alternatives, each member ranks each alternative from 1 (lowest) to 10 (highest). The votes are then calculated and the alternative with the highest total score is selected. This method includes a voting procedure and, therefore, gives the impression that the final decision represents each person's opinion. But it takes time and can result in a decision that no one fully supports.

- Consensus rule

Consensus in decision making means that all members genuinely agree that the decision is acceptable. With this rule, the decision is discussed and negotiated in the group until everyone affected by understandings and agreements with what will be done. Therefore, all members feel that they have had an equal opportunity to influence the decision and will continue to support the group. Because there are time constraints in coming to a group decision and there is no perfect system, a decision by consensus rule is one of the most effective methods. To successfully use this method, communications have been sufficiently open in such a way that everyone in the group feels that they have had their fair chance to influence the decision. However, it is one of the time-consuming techniques for group decision making, and some times it may be difficult to reach a consensus in a group. To overcome the disadvantages, some other methods are developed by combining this rule with other ones.

In this book, the fuzzy group decision-making method and fuzzy group decision support system to be presented in Chapters 10 and 11 are mainly based on the '*consensus*' rule, which also combines with the ranking and majority rules.

Researchers have developed some detail methods and techniques for improving the processes of group decision making by using the above-mentioned rules. Two most popular and representative techniques are the *Delphi technique* (also called *Delphi method*) and the *nominal group technique* (also called *multi-voting technique*).

The *Delphi technique* was developed by Gordon and Helmer in 1953 at RAND. It aims at building an interdisciplinary consensus about an opinion, without necessarily having people meet face to face, such as through surveys, questionnaires, e-mails *etc*. Many applications have shown that the technique is effective in allowing a group of individuals, as a whole, to deal with a complex problem. It is particularly appropriate when decision making is required in a political or emotional environment, or when the decisions affect strong factions with opposing preferences. It comprises a series of questionnaires to a pre-selected group of experts. These questionnaires are designed to elicit and develop individual responses to the problems posed and to enable experts to refine their views as the group's work progresses in accordance with the assigned task. For example, with a number of research grant applications, the research office will ask a group of experts to fill up an evaluation form (questionnaire) to put their review results on. The research office will then collect these experts' evaluation results for getting a decision on these applications.

The *Nominal Group Technique* is for achieving team consensus quickly when the team is ranking several alternatives or selecting the best choice among them. The technique basically consists of having each team member come up with their personal ranking of the options or choices, and collation of everyone's rankings into the team consensus. It can build every member's commitment to whatever choice or ranking the team makes because every member was given a fair chance to participate. It can therefore eliminate peer pressure in the team's selection or ranking process and make the team's consensus visible. Defining a problem statement, generating a list of alternatives, and finalising the list of alternatives are the three main steps to apply for within this technique.

3.6 Group Support Systems and Groupware

Many computerised tools have been developed to provide group work support. These tools are called *groupware* because their primary objective is to support group work. The work itself may be known as *computer-supported cooperative work* (CSCW). Groupware continues to evolve to support effective group work. Most group work takes place in meetings. The goal of groupware, as it was specifically developed as *group support systems* (GSS), is to support the work of groups throughout every work activity such as idea generation, consensus building, anonymous ranking, voting, and so on, normally occurring at meetings.

Group support systems represent a class of computer-based technologies and methodologies that are developed to support group work and to improve the efficiency and effectiveness of group meetings. GSS can particularly enhance creativity in the decision-making process when it is specifically developed as group decision support systems (GDSS), which will be discussed in Chapter 4. Numerous authors have described applications of GSS in a variety of areas, including telecommuting, teleconferencing, supply chain management, and electronic commerce.

Though many types of GSS have been developed, two fundamentally different viewpoints have underpinned most of the systems. One view assumes that the task of a group is to exercise discretion, which implies that the support provided must allow group members to consider uncertainty, form preferences, make tradeoffs, and take decisions. This approach recognises that most group decision making should rely on the application of modelling and decision theory, an understanding of group processes, and the use of information technology. This has been called *Decision Conference* (DC). Another view is driven by communication needs and utilises computer-based information technology as a means of facilitating group communication. This kind of systems assumes that interpersonal communication is the primary activity of group decision making and that the function of GSS is to improve the group's communication through the application of information technology. Systems supporting this view are usually called *Electronic Meeting Systems* (EMS).

However, a change has occurred in GSS, in particular GDSS, research and applications. In the 1980's, GSS research largely was concerned with *decision rooms* and suggestions of the impact that GSS could have. The recent research has recognised a much broader application and role for GSS, which are now viewed as organised searching for alternatives, communication, deliberation, planning, problem solving, negotiation, consensus building, and vision sharing, as well as decision making for group members, not necessarily in the same place or at the same time. With the Internet development, both kinds of commercially GSS systems offer business users a structure within which they can make group work more quickly, with more inputs from a wide network of experts, and with vastly improved coordination.

Gray and Mandviwalla (1999) indicated that we have reached a point where we need to expand what we can do with GDSS. The growth of GDSS can come in: (1) increasing the capabilities available to groups so that they match all aspects of meeting; (2) increasing the range of applications so that they can support more organisational decision-making task; and (3) improving the effectiveness of group so as to achieve more productive and effective group decision making. In Chapters 4 and 11, these issues will be further explored and presented.

A GSS is a generic term that includes all forms of collaborative computing that enhances group work. Though a complete GSS is still considered a specially designed information system, many of the special capabilities of GSS have been embedded in productivity tools. More commercial software, for example, Microsoft NetMeeting Client (part of Windows) has been developed. And more and more GSS are easy to use because they have a Windows GUI or a Web browser interface.

3.7 Summary

Individuals, organisations, and community groups are often faced with important decisions to make. For a group decision to be successful, it must find a suitable group decision model and method to creatively solve their problems and focus on reaching their goals. In this chapter, we introduce some popular models and discuss related methods for group decision making. We present content-oriented group decision-making issues and analyse how to find an optimal or a satisfactory solution given certain group constraints, or objectives. We also discuss process-oriented group decision issues, which are based on the observation that the group goes through certain phases in the group decision-making process, and on the belief that there could be an arranged way to effectively deal with these phases. These contents will be used in Chapters 4, 10, and 11.

This page intentionally left blank

Chapter 4

Decision Support Systems

The central purpose of Decision Support Systems (DSS) is to improve the quality and effectiveness of decision making. DSS have been widely used by managers as a specific management tool and approach, and have become a means of reducing the uncertainty and risk traditionally associated with decision making. The term DSS has been sometimes used as an umbrella term to describe any and every computerised system used to support decision making in an organisation.

We first briefly introduce concepts of DSS and discuss major characteristics of DSS in the chapter. We then present the main types of DSS and particularly discuss multi-objective DSS, multi-attribute DSS, group DSS, intelligent DSS, and Web-based DSS, respectively. Finally, we explain the components of DSS and their functions.

4.1 Concepts

Since the term DSS was coined in the early 1970s, the topic of DSS has stimulated great interest in both its research and applications. The classic definitions of DSS identified it as a system intended to support managerial decision makers in organisations for ill-structured (semi-structured or un-structured) decision situations. For example, Gorry and Scott Morton (1971) defined DSS as interactive computer-based systems, which help decision makers utilise *data* and *models* to solve ill-structured problems. Another classic definition of DSS, provided by Keen and Scott Morton (1978), is that DSS couple the intellectual resources of individuals with the capabilities of the computer to improve the quality of decisions. A DSS is a computer-based support system for management decision makers who deal with ill-structured problems. However, the term DSS is a content-free expression; that is, it means different things to different people. Therefore, there is no universally accepted definition of DSS. Not specifically stated, but implied in these definitions, is the notion that the system would be computer-based, would operate interactively online, and preferably would have graphical output capabilities (Turban and Aronson, 1998).

From the application point of view, a DSS can be seen as an approach for supporting decision making. It uses an interactive, flexible and adaptable system especially developed for supporting the solution for a specific ill-structured management problem. It uses data, provides an easy user interface, and can incorporate decision makers' own insights. In addition, a DSS usually uses models, which are built by an interactive and iterative process.

In summary, the definition of DSS that we will use in this book is described as follows: A DSS is a computer-based information system, which supports decision makers and confronts ill-structured problems through direct interaction with data and analysis models.

Each part of this definition has a key concept that contributes to the unique character of DSS. A further discussion for the balance between D, S, and S of DSS has also been made by Keen and Scoot-Morton (1978). '*Decision (D)*' relates to the non-technical, functional and analytic aspects of DSS and to criteria for selecting applications. '*Support (S)*' focuses on the implementation and understanding of the way real people operate and how to help them. '*System (S)*' directly emphasises the design and development of technology. Therefore, the relevant research problems have been identified in this area, such as different approaches to the building of DSS, different methods implemented in DSS, and different tools used by DSS.

A DSS is intended to support, rather than replace, managerial decision making; to be an adjunct to decision makers to extend their capabilities but not to replace their judgment in ill-structured decisions; and to be with a view to improving decision making effectiveness, rather than efficiency. Sometime, there may be no optimum solution for some decision problems because these are ill-structured situations. The decision therefore must evolve through the interaction of decision makers with resources such as data and analysis models.

The literature on DSS has always had an emphasis on increased effectiveness of decision making, that is, an increase in quality of the decision, as the main benefit of DSS. Some evaluation researches have proposed the effects of DSS on decision outcomes development. These studies evaluated the improvements in decision quality typically associated with DSS which are due primarily to '*development*' or '*reliance*' effects. Some researches also examined how the introduction of DSS contributes to decision quality after controlling for task familiarity. Also, a good DSS environment improves the decision making process, by speeding up the learning process of decision makers and providing reliable methods.

4.2 Characteristics

The technology for DSS should consist of three sets of capabilities in the areas of *dialog* (*D*), *data* (*D*) and *modelling* (*M*), what Sprague and Watson (1980) called the *DDM paradigm*. They also pointed out that a good DSS should have a balance among the three capabilities. The first '*D*' means that DSS should be easy to use to allow non-technical decision makers to interact fully with it. The second '*D*' indicates that DSS should have access to a wide variety of data sources, bases, formats, and types adapted in it. The '*M*' indicates that DSS should provide modelling. However, in practice, a DSS may be strong in only one area and weak in the others of the three, which is based on the requirements of decision makers.

A DSS can be employed as a stand-alone tool used by an individual decision maker in one location, or it can be distributed throughout an organisation and in several organisations. It can be integrated with other DSS or information system applications, and it can be distributed internally and externally, using networking and Web technologies.

Turban and Aronson (1998) listed some ideal characteristics and capabilities of DSS:

- ill-structured decision programs
- for managers at different levels
- for groups and individuals, but humans control the machine
- support intelligence, design, choice phases
- support variety of decision styles and processes
- adaptability and flexibility in carrying out a decision support task and approach of the users
- interactive and extremely user friendly so as to be easy for non-computer people
- combine the use of models and analytic techniques
- data access and retrieval
- integration and Web connection

With these characteristics, DSS can improve decision makers' efficiency, effectiveness, and productivity in decision making. It also can improve decision problem solving and facilitate communication within an organisation. This book will particularly show some characteristics in the list of 'for group and individuals,' 'support a variety of decision processes,' 'flexibility in carrying out a decision task,' 'interactive and extremely user friendly so as to be easy for non-computer people,' 'combine the use of models and analytic techniques,' and 'integration and Web connection' in those DSS we developed and discussed in this book. These characteristics are provided by the DSS major components, which will be discussed in Section 4.9.

4.3 Types

Based on these characteristics, we can identify five main types of DSS.

(1) Model-driven DSS

This type of DSS is the milestone of the beginning of DSS. Nowadays, it emphasises on the access to various models, such as statistical and optimisation models. In addition, some advanced model-driven DSS can simulate a situation, which is programmed by the user or developer to support decision making. Therefore, model-driven DSS can be used to analyse business or other situations and generate a solution to help decision makers' right decision. Multi-criteria DSS (MCDSS), which includes multi-objective DSS (MODSS) and multi-attributes DSS (MADSS) here, is a typical model-driven DSS where MCDM models are adopted in the DSS.

Both DSS and MCDM seek to support all phases of decision making, although these two disciplines are different in offering relative support roles and in support mechanisms. The similarity of decision making problems addressed by the fields of DSS and MCDM would suggest that they could borrow and build from each other. A marriage between DSS and MCDM promises to be practical and intellectually fruitful. Therefore, an integration of MCDM and DSS – MCDSS was proposed as a '*specific*' type of systems within the broad family of DSS.

Even though they include much the same components as classical DSS, MCDSS have special characteristics, including:

- they allow analysis of multiple criteria (objectives or attributes);
- they use a variety of multi-criteria decision models (methods) to compute efficient solutions; and most importantly,
- they incorporate users' input (interaction) in various phases of modelling and getting.

Decision makers can make interaction in various stages of model management, model development, and problem solving. MCDSS intend to provide the necessary computerised assistance to decision makers to solve multi-criteria decision problems. Decision makers are encouraged to explore the support tools available in an interactive fashion with the aim of further defining the nature of the problem. The ultimate success of DSS lies in their ability to help decision makers solve ill-structured problems through the direct interaction with analytical models. Such ability can be enhanced by combining the various features of MCDM with DSS.

In MCDSS, MCDM complements DSS and vice versa due to the differences in underlying philosophies, objectives, support mechanisms, and relative support roles. MCDSS, as the integration of DSS and MCDM, is construed to be the application of ideas, concepts, and strategies initially developed in one area, to problems better addressed in the other domain. Researchers of both areas have accepted this standpoint.

(2) Data-driven DSS

This type of DSS collects and provides real time access to a large operational or even data warehouse to support decision making. Such database or data warehouse can have internal or external data. It can also provide queries and management reports according to user's requirement. The more advanced data-driven DSS is combined with online analytical processing (OLAP) and data mining (such as, spatial data mining, correlation mining, linking mining, and Web mining). Therefore, it can be used to analysis the historical data and find data associations in order to help users identify happening facts.

(3) Knowledge-driven DSS or Intelligent DSS (IDSS)

This type of DSS often includes a rule-based system to suggest decision makers to take certain kind of actions. On the other hand, it can be regarded as a person-computer system with some specialised problem-solving expertise. In fact, the decision-making process itself is one of the intelligent activities of human beings. The term *intelligence* in DSS is the ability of DSS to use possessed information, knowledge, and inference in order to achieve new objectives in new circumstances. Over the past twenty years, DSS designers have tried to use various intelligent methods to handle complex situations and improve the performance of DSS. Knowledge-based reasoning, machine learning, data mining, data fusion, soft computing, and intelligent agencies all have contributed greatly in the development of IDSS.

(4) Group Decision Support Systems (GDSS)

This type of DSS, as we have mentioned in Chapter 2, allows multiple users to work collaboratively in the group for a decision problem. GDSS, in general, support a decision meeting where each member can give their opinions through a computer or a facilitator (coordinator). However, getting a group of decision makers together in one place and at one time can be difficult and expensive. Attempts to improve the work of groups with the aid of information technology have been described as distributed electronic meeting systems, Delphi technique-based collaborative systems, and distributed online GDSS.

(5) Web-based DSS

The implementation of Web-based DSS has been popular since the mid-1990s when Internet technology develops rapidly around the world. Web-based DSS use Web browser to access Internet or Intranet. In addition, TCP/IP protocols are used to communicate with the server/client architecture, which can be applied in Web-based GDSS as well. With information technology, Web-based DSS can be model-driven, date-driven, knowledge-driven, communications-driven or a hybrid of them. Recent developments in e-commerce, e-business, e-government, and e-service provide a fertile ground for this new type of DSS applications.

It should be indicated that decision is regarded differently in different decision theories; and, furthermore, different sciences are contributing to decision making paradigms and have different classifications and categories. We have only outlined the five main types of DSS, which are in line with the scope of this book.

4.4 Multi-Objective DSS

As the separate areas, MODM and MADM tend to draw from different sources for their solution procedures. MCDSS can thus be broadly categorised into MODSS and MADSS. These two categories have different requirements of data and model management for effective decision support, and have different elements of methodology matched with practical.

MODSS is applied to support the decision making in which decision problems can be described by an MODM model, *e.g.*, (2.2.1). MODSS has gained widespread attention in its algorithms, methodology implementation, as well as their applications. Compared to MADSS, MODSS require more model management functions than data management functions. Problem structuring in MODSS mainly includes generating *objectives*, *constraints* and *decision variables*, three important components of the MODM model. The model constructing has to be performed initially by decision makers. Therefore, it should be completed manually first and then can be evolved gradually over a number of iterations. The model can be structured in an interactive fashion of an MODSS using a graphical user interface environment. Two matrices, called *objective matrix* and *constraint matrix*, have to be generated first to construct an MODM model (Quaddus, 1997).

Table 4.1 lists 17 MODSS including decision support tool (C) or specific application (S), and analyses the six main characteristics/ functions: method-base (MB) or model management, database (DB) or data management ability, knowledge-base or other intelligent component (I), group environment (GE), and graphical user interface (GUI).

Table 4.1: List of some selected MODSS

No	*Name*	*C*	*S*	*MB*	*DB*	*I*	*GE*	*GUI*	*References*
1	IMOP	*		*					(Werners, 1987)
2	DINAS		*						(Ogryczak, Studzinski, and Zorychta, 1989)
3	VIG	*						*	(Korhonen and Wallenius, 1990)
4	HYBRID	*							(Poh and Quaddus, 1990)
5	Interactive MOLP	*		*	*			*	(Korhonen, Lewandowski, and Wallenius, 1991)
6	R&DPS MODSS		*		*			*	(Stewart, 1991)
7	WQ-PMODSS		*		*			*	(Yakowitz *et al.*, 1993)
8	ISGPII	*						*	(Hwang, Lai, and Liu, 1993)
9	MOLP-PC	*		*					(Poh, Quaddus, and Chin, 1995)
10	IMOST	*		*	*	*		*	(Lai, 1995)
11	PDSS	*			*			*	(Paige *et al.*, 1996a)
12	FORMDSS		*		*		*	*	(Tecle, Shrestha, and Duckstein, 1998)
13	GMCRII	*					*	*	(Hipel, 1992)
14	IMOGDSS	*		*	*	*	*	*	(Lu and Quaddus, 2001)
15	WMODSS	*		*	*			*	(Lu, Zhang, and Shi, 2003)
16	FMODSS	*		*	*			*	(Wu, Lu, and Zhang, 2004)
17	FMOGDSS	*		*	*		*	*	(Wu, Lu, and Zhang, 2007)

'*' denotes '*yes*'

Within the 17 selected MODSS, Interactive Multiple Objective Programming (IMOP) can provide with solutions to multi-objective programming problems subject to both strict and flexible constraints. An integral part of the system is an extension of a fuzzy-set approach assessing possible solutions by their degrees of membership to the objectives and constraints. The Dynamic Interactive Network Analysis System (DINAS) enabled the solution of various multi-objective transhipment problems with facility location. It used an extension of the classical reference-point approach to handle multiple objectives. In this system, decision makers can specify acceptable and required values for given objectives. The Visual Interactive Goal Programming (VIG) was designed to support both the modelling and solving of an MOLP problem based on goal programming. HYBRID used the solution of a two-person zero-sum game with mixed strategies to generate efficient solutions, and then proceeded to modify the feasible region using responses from decision makers. The Interactive MODSS had an MODM method-base of five popular MOLP methods. These methods can be used in stand alone mode or in any sequence the user wishes. Stewart (1991) developed an MODSS for the selection of a portfolio of R&D projects, which was carried out for a large electricity utility corporation. The R&DPS MODSS was constructed around a reference point approach for the underlying MODM problems. WQ-PMODSS is to predict the impact of alternative management systems on surface and groundwater quality as well as farm income. ISGPII is an interactive MODSS to provide a process of psychological convergence for decision makers, whereby it learns to recognise good solutions and their importance in the system, and to design an optimal system, instead of optimising a given system. MOLP-PC is an integrated MODSS, which has a method-base of fourteen popular MODM methodologies for solving MOLP problems. IMOST was investigated to improve the flexibility and robustness of MODM methodologies. The interactive concept provided a learning process about the system, whereby decision makers can learn to recognise good solutions, the relative importance of factors in the system, and then design a high-productivity and zero-buffer system instead of optimising a given system. A prototype decision support system (PDSS), using multi-objective decision theory and embedded simulation models, was developed to evaluate landfill cover designs for low level radioactive waste disposal sites. FORMDSS is a multi-objective and/or multi-person DSS for analysing multiple resource forest management problems. The procedure includes formulating the problem in a multi-objective and group decision-making framework, and solving it using two solution techniques which consist of a distance-based compromise programming and a cooperative game theoretic approach of the Nash equilibrium type. GMCRII is an MODSS tool for providing strategic advice in multi-participant multi-objective decision-making situations. WMODSS is a Web-based MODSS, which can support online decision making for MOLP problems.

The intelligent multi-objective group DSS (IMOGDSS) was developed for solving MOLP problems under an individual and/or a group decision making environment. We will discuss this system in details in Chapter 12. FMODSS and FMOGDSS are two fuzzy technique based DSS, which will be presented in details in Chapters 8 and 13, respectively.

4.5 Multi-Attribute DSS

Multi-attribute decision support systems (MADSS) employ one or more MADM methods for generating alternatives and selecting solutions. The model management function of an MADSS concentrates more on the *problem structuring* and *model structuring*. Watson and Buede (1987) defined the problem structuring as the identification of decision makers, the determination of decision making boundaries, the determination of the principal objective, the willingness and ability of other decision makers to provide inputs to the analysis. The problem structuring function can be integrated with the MADSS for better problem structuring support. It involves the creation of an evaluation tool for comparing the alternatives.

A variety of MADSS can be found in the literature. Some earlier developed systems have a weak integrated function of DSS, a simple model management ability and data management ability. Some recently developed systems achieved improvements in the implementation of model management, data management, intelligent support, GUI and multiple decision makers environment. Table 4.2 lists 17 MADSS including tools (C) and applications (S), and analyses their main characteristics related to: method-base (MB); database (DB); intelligence (I); group environment (GE); and GUI.

Table 4.2: List of some selected MADSS

No	*Name*	*C*	*S*	*MB*	*DB*	*I*	*GE*	*GUI*	*References*
1	PREFCALC	*							(Lagreze and Shakun, 1984)
2	HIVIEW	*							(Barclay, 1987)
3	EQUITY		*						(Barclay, 1988)
4	VISA	*					*	*	(Belton and Vickers, 1989)
5	CRITERIUM	*					*		(Sygenex, 1989) (Bois *et al.*, 1989)
6	PROMETHEE	*						*	(Bois *et al.*, 1989)
7	MCDM advisor	*		*		*		*	(Korhonen, Lewandowski and Wallenius, 1991)
8	EXPERT CHOICE	*					*		(Expert-Choice-Inc, 1992)
9	GRADS		*					*	(Klimberg, 1992)
10	MCDA-DSS	*		*	*			*	(Antunes *et al.*, 1994)
11	TSDSS		*					*	(Yau and Davis, 1994)
12	MCView	*			*			*	(Vetschera, 1994)
13	ICDSS		*		*			*	(Agrell, 1995)
14	InterQuad	*							(Sun and Steuer, 1996)
15	IMADS	*		*	*	*			(Poh, 1998)
16	ALLOCATE		*					*	(Quaddus and Klass, 1998)
17	Web-based FGDSS	*		*	*		*	*	(Lu, Zhang, and Wu, 2005)

'*' denotes '*yes*'

We have only briefly overviewed the salient features without evaluating the advantages and disadvantages of these MADSS. The details are available from the cited references. The Web-based FGDSS listed in Table 4.2 uses MCDM method and will be presented in Chapter 11 in details.

4.6 Group DSS

Decision support systems have been well researched and a variety of interactive solution methods of group decision making (GDM) have been derived. Systems that combine appropriate technologies and methodologies of DSS and GDM show the potential to enhance the efficiency and effectiveness of group decision work. Such applications of information technology to support the decision work of groups have been referred to as *group decision support systems* (GDSS) (Gray, 1987). A GDSS is characterised as an interactive computer-based information system that combines the capabilities of communication technologies, database technologies, computer technologies, and decision technologies to support the identification, analysis, formulation, evaluation, and solution of problems by a group in a user-friendly computing environment. Therefore, GDSS is a collection of hardware, software, people and procedures appropriately arranged in an interactive computer-based environment that supports a group of decision makers who are engaged in a decision-making process. A GDSS has two major goals in fulfilling its mission. These are improving the productivity of idea generation by speeding up the decision-making process and increasing the effectiveness of decision making by optimising quality of resulting decisions. In addition, every GDSS has a set of features, which have to be considered when a GDSS is designed. Since it is a software system, it must be user-friendly. Apart from that, it must support a group of decision makers for improving the decision-making process. Generally a GDSS environment may include a group facilitator, who coordinates the group actions throughout the decision making process.

Group decision support systems typically offer a wide range of capabilities, including computerised support for interactive modelling, group preference aggregation mechanisms, communication, idea generation, and freedom of expression to reach towards an optimal group solution. Importantly, GDSS is used in decision groups, not in general group meetings, and to support decision making, not only creating alternatives. This is the foundational difference between GDSS and group support systems (GSS).

Interest in the development of GDSS emerged in the early 1980s. The growing availability of local area networks and group communication services in the past few years, such as e-mail, online chart room, is making this GDSS increasingly available. A variety of academic articles on GDSS with these new technologies have promoted the incorporation of quantitative decision making models, such as MODM and MADM in GDSS.

The focus in GDSS research has been primarily on the group's decision models, methods, interaction and communication with a strong emphasis on consensus-building. However, the appropriateness of any decision model and method within a GDSS depends on the conditions of members, tasks, and decision environment. Also, GDSS design must take into consideration of the members' behaviours as well as technical issues in order to develop useful and effective systems. A variety of comprehensive and integrative frameworks, which combine the behavioural characteristics of GDM with the technical specifications of DSS, are used in the development of GDSS. Communication channels in GDSS include face to face and computer mediated communication channels. Some experiments have shown that the face-to-face channel of communication and the computer mediated communication all have their own advantages in GDSS. Other topics about GDSS design, the use of a GDSS to facilitate group consensus, the interacting effects of GDSS and uncertainty issues will be discussed in Chapters 10-13.

Group decision support systems have gained such a high level of popularity that it is currently used widely in industry. Moreover, researchers are utilising the mechanism of GDSS in various academic research areas, such as automated facilitation, speech recognition, automatic idea consolidation, multi-lingual groups, knowledge management, fuzzy logic, and team situation awareness. Two more significant changes are to apply Web technology into GDSS to build

Web-based GDSS and to add intelligent technology into GDSS to build intelligent GDSS, which will be discussed with the development of real systems in Chapters 11, 12, and 13.

4.7 Intelligent DSS

The complexity of decision making is increasing. The active involvement of the user and the computer in an intelligent way is necessary in decision process. With the complex decision-making environment, the insufficiency of conventional DSS is emerging in the following factors:

- The conventional DSS mainly adopt static models to manipulate data, which require decision makers to have knowledge in both issues of decision domain and models. Therefore, the function of DSS is passive in the decision support process and cannot provide active support while the decision environment changes.
- Under the dominated process of decision makers, a conventional DSS uses models to solve problems; it needs decision process companied with definite calculability. It is therefore hard to provide support for some existing unstructured issues in decision-making process.
- A conventional DSS is based on quantitative mathematic models, it lacks of providing support measure for qualitative issues, such as data imprecise problems, uncertain problems, and inference problems.

Artificial intelligence (AI), including knowledge-based systems/ expert systems (KBS/ES), natural language analysis, machine leaning and inference, has experienced significant progress in research and implementation. As a powerful tool, AI allows a human-being to easily control and direct power sources in the accomplishment of a task by providing cognitive amplification or augmentation. In particular, ES can build the domain expert knowledge-base and make the machine learning achieve the human expert in some domains. The ES application in management aims at specific domain decision issues.

Knowledge is represented in many different ways, such as frames, semantic nets, and rules *etc*. Knowledge is often processed in inference machines, which normally performs symbol processing. A knowledge-based ES uses knowledge and problem-solving techniques on a skill level comparable to those of human experts and intends to serve as consultants for decision making. These systems consist of a knowledge-base, containing facts, rules, heuristics, situation patterns, and an inference system that makes decisions within a domain. A knowledge-based ES enables information system builders to move problem domain knowledge from the human to the computer so as to support problem recognition, problem structure and problem solving. It provides expertise when human expertise is not available, more uniformly, and sometimes faster and assist managers in making complex decisions. It has become a trend that DSS products incorporate, and will eventually encompass, tools and techniques from AI, particularly from knowledge-based ES.

Based on the two factors: (1) AI can solve some qualitative, approximate, and imprecise knowledge for DSS; (2) AI technique yields potential benefits to decision makers, and IDSS takes the advantages of both AI and DSS, Turban and Aronson (1998) proposed two kinds of possible connections between DSS and AI techniques, in particular, ES: (1) intelligent techniques integration into the conventional DSS components; and (2) intelligent techniques as an additional component of DSS. The studies about these combinations have received great attention.

Intelligent DSS (IDSS), as the combination of AI/ES and DSS, can be seen as a DSS that provides access to data and knowledge-bases and/or conducts inference to support effective decision making in complex problem domains. The research of IDSS has focused on from decision making component expanding to integration, from quantitative model to knowledge-based decision-making approach. This makes the theory and measure of IDSS more mature and accurate. Development and implementation of IDSS and their applications in practice have become an active research area, from which IDSS provide a number of advantages and potential benefits:

- IDSS can deliver automated *decision analysis* assistance and offer a wide range of realistic possibilities for helping decision makers. The knowledge-base has become a form of combined data/model-base, the inference engine can be viewed as a knowledge-base management system, and the language system is a part of the dialog. These give decision makers the opportunity to explore 'what if...' situations with different types of inputs. Also, it can handle uncertainty when data are incomplete and uncertain. In particular, IDSS enable us to analyse data and applies processing rules to determine whether variances are significant and explain in terms of factors that contributed to the variances.

- IDSS have great potential in improving the quality of decision making. Combined with formal decision making methods and ES technology, the IDSS is capable of delivering more reliable decision support tools for users. The IDSS is to assist managers with assessment of the relative importance of competitive priorities in their organisations. As human experts, IDSS have the potential to facilitate effective and swift decision making in the selection of appropriate applications that best match an organisation's manufacturing strategy. The intelligent process of IDSS can be realised by system detects data trends in a dynamic environment, incorporates optimisation modules to recommend a near-optimum decision, and includes self-learning modules to improves efficiency of decision making.
- IDSS can be used by more decision makers including those who have *little knowledge* in decision models, methods and analysis skills. IDSS' ability to provide unprecedented level of automated guidance on the analysis of a class of decisions, thereby enabling end-users with little knowledge in decision analysis to be effective decision analysts in that domain. Comparing with the conventional DSS, an IDSS could explain why certain variances were deemed significant and why certain factors were found to have caused a variance to be significant. To the extent that, IDSS could determine the cause for many of the significant variances, managers need only focus attention on examining the generation processes for the most significant unexplained variances.

We will present an intelligent multi-objective group DSS, which combines an ES with DSS, in Chapter 12.

4.8 Web-Based DSS

The Web has grown rapidly because of its single user interface paradigm, distributed architecture, and the growth of open standards. Web browsers make it easy for various users to gain access to many diverse sources of information. In particular, Web technologies have provided a new media for sharing information about decision making and become the preferred platform of choice for the delivery of DSS.

Traditional DSS requires software to be installed on individual workstations or computers with a particular standard user environment. Web-based DSS have gone a long way towards solving a number of these user flexibility and accessibility problems and have opened up DSS capabilities to a broader user group.

Web-based DSS take Web technologies to deliver the decision making process among a different group of geographically distributed end-users. This geographical freedom allows DSS to be utilised across long location and at any time. DSS also can be easily integrated into existing applications or systems in Web domains such as corporate intranets, enterprise-wide extranets and Internet. For example, it can support government policy makers, business managers, and citizens in decision making through an e-government service system.

The Web-based DSS architecture combines typical DSS structure and Web infrastructure. Most Web-based DSS are built using three-tier architecture. Decision makers using a Web browser send a request using the hypertext transfer protocol (HTTP) to a Web server. The Web server processes the request and the results are returned to the user's Web browser for display (see Fig. 4.1). Web applications are designed to allow any authorised user, with a Web browser and an Internet connection, to interact with them (Power and Kaparthi, 2002). The application code usually resides on a remote server and the user interface is presented at the client's Web browser.

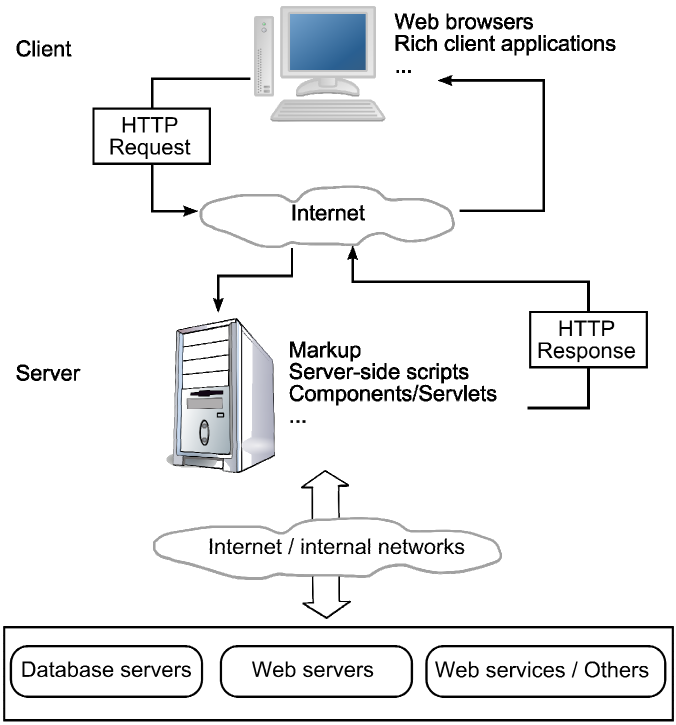


Fig. 4.1: Web-based DSS Architecture

Web technologies such as HTML and browser applications were developed in the early 1990s. While the broad use of the Web was after 1996, Web-based DSS didn't begin its growth until 1999. This rapid growth was quickly followed in 2000 by the introduction of DSS supporting Application Service Providers (ASPs) (Bhargava and Power, 2001). The latter of these facilitating the decision support process among many decision makers without restrictions for time and location. At present, utilising ActiveX or Java-enabled Web browser software, DSS can be distributed without regard to platform or geography. In summary, Web-based DSS have shown their advantages in applications:

- Web-based DSS have reduced technological barriers to make decision-relevant information and decision support tools available to use in geographically distributed locations. Because of the Web infrastructure, enterprise-wide DSS can now be implemented at a relatively low cost in geographically dispersed companies to dispersed stakeholders, including suppliers and customers. Using Web-based DSS, organisations can provide DSS capability to managers over an intranet, to suppliers over an extranet, or to customers and any stakeholder over the global Internet.

- Developing Web-based DSS will increase the use of DSS and decision information in the organisation. Because Web-based DSS improves the rapid delivery of 'best practices' analysis and decision-making frameworks, and can promote more consistent decision making on repetitive decision tasks across a geographically distributed organisation. The Web-based IDSS also provides a way to manage a company's knowledge repository and to bring knowledge resources into the decision-making process.
- The Web-based DSS can also reduce some of the problems associated with the competing 'thick-client' enterprise-wide DSS architecture where special software or configuration needs to be installed on users' computers. It becomes much easier to add new users, and their initial training needs are minimal. Web-based DSS thus reduce the costs of operations, support, maintenance, and administration.

Many DSS software companies provide case studies of successful Web-based DSS implementations at their Web sites. The following three real-world application directions of Web-based DSS further show the advantages over the conventional DSS:

(1) Supporting Staff

GroupSystems (Morehouse, 2005) is used to communicate information and to provide input, discuss solutions, and create reports of recommended action. Many people participate in the crisis management activities using GroupSystems OnLine. Also, some companies developed DSS for executives, salespeople, and analysts. When users need information such as sales trends, they query the DSS themselves and get an answer in a few minutes. Furthermore, they can quickly analyse the data in different ways using Internet access.

(2) Supporting Customers

The Customer DSS is a Web-based, marketing model that establishes a link between a company and its customers and provides assistance to the decision-making process (O'Keefe and McEachern, 1998). Many Web sites have decision support applications for customers. Active Buyer Guide (http://www.activebuyersguide.com) offers product recommendations directly from its shopping site. For example, users can use a '*Compare*' feature to make pairwise comparisons of car models across pre-specified attributes.

(3) Supporting Suppliers

Some companies have created DSS for extranets as well as for intranets. For example, Artesyn Technologies (www.artesyn.com) has virtual design decision support tools to provide customers of its power supply products with pre-sales technical support. Also, some DSS can provide suppliers with Web access to sales forecasts and decision support capabilities. Some companies are also creating their Web-based DSS in which suppliers can use to control costs or reduce inventories.

4.9 Components

As already explained in Section 4.2, a database management system (DBMS), a model-base management system (MBMS), and a user interface are the main components of a general structure of a DSS. With the development of new information technologies, more components, such as knowledge-base, can be added into a DSS (Fig. 4.2). However, in a particular DSS application, it may only compose some of these components.

Data management is the most important component of a DSS. It mainly includes a database, which contains relevant internal or external data for the situation and is managed by a DBMS**.** The data management component can be interconnected with data warehouse, and internal and external decision-making data.

Model/method management as another important component is a software package that includes statistical, optimisation, or other quantitative models and methods that provide the system's analytical capabilities and appropriate software management. Modelling languages for building custom models can also be included by MBMS. This component can be connected to internal or external storage models, embedded in a model-base. The model-base provides simplified representations or abstraction of reality. A method-base is related to the model-base. It often includes a set of algorithms used to solve proposed models in the model-base.

Knowledge management component can support any of the other subsystems or act as an independent component. It can be interconnected with the organisation's knowledge-base, which may consist of a rule-base, a fact-base, and so on.

User interface means that user communicates with the DSS. A better decision can be derived through a DSS from the intensive interaction between computers and decision makers.

These components of DSS can be connected to the corporate intranet, an extranet, and/or the Internet. However, in practice, DSS may have some special components, such as text-base, multi-media database, and so on.

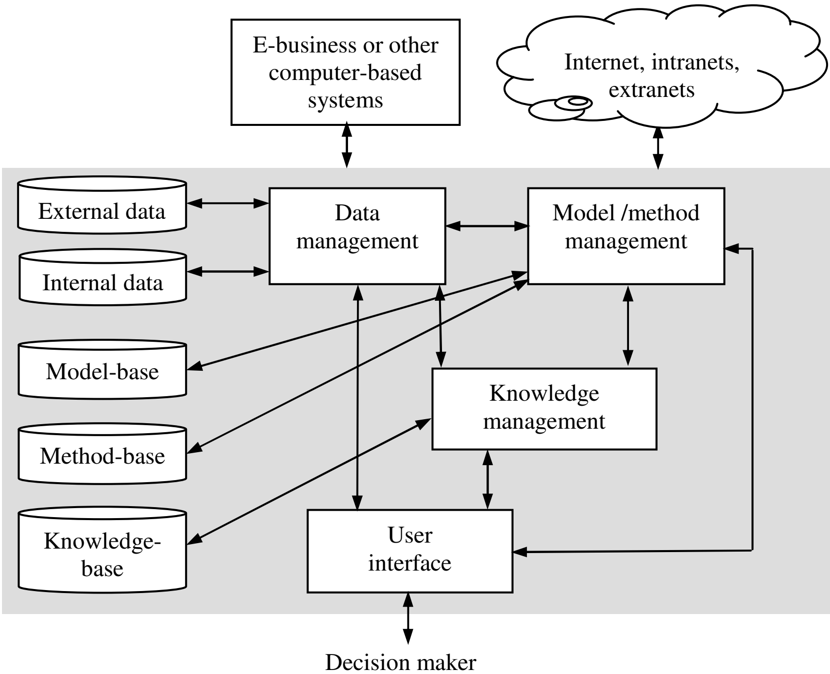


Fig. 4.2: The main components of a DSS

4.10 Summary

Decision support system is an interactive computer-based information system, which can help decision makers utilise data, models, and methods to solve ill-structured decision problems. DSS incorporate database, model-base, method-base, and the intellectual resources of individuals or groups with the capabilities of the computer to improve the quality of decisions. Models, intelligence, methodology, and interaction in DSS are the key elements to formulate decision support. For several main types of DSS listed in this chapter each of them varies with their particular models and methodologies. DSS in the broad sense as the application oriented result of decision analysis and intelligence. Fuzzy set theory as a kind of intelligent technologies has been applied in

MODSS, MADSS, GDSS, and Web-based DSS. This is the main focus of this book, which will be discussed in the rest of this book.

Chapter 5

Fuzzy Sets and Systems

Fuzzy sets, introduced by Zadeh in 1965, provide us a new mathematical tool to deal with uncertainty of information. Since then, fuzzy set theory has been rapidly developed and many successful real applications of fuzzy sets and systems in wide-ranging fields have been appearing. In this chapter, we will review basic concepts of fuzzy sets, fuzzy relations, fuzzy numbers, linguistic variables, and fuzzy linear programming, which will be used in the rest chapters of the book.

5.1 Fuzzy Sets

5.1.1 *Definitions*

Definition 5.1.1 (Fuzzy set) Let X be a universal set. Then a *fuzzy set* $\tilde{A}$ of X is defined by its membership function.

$$\begin{aligned} \mu_{\tilde{A}} &: X \to [0, 1], \\ & x \mapsto \mu_{\tilde{A}}(x) \in [0, 1]. \end{aligned} \tag{5.1.1}$$

The value of $\mu_{\tilde{A}}(x)$ represents the grade of membership of x in X and is interpreted as the degree to which x belongs to $\tilde{A}$, therefore the closer the value of $\mu_{\tilde{A}}(x)$ is 1, the more belongs to $\tilde{A}$.

A crisp or ordinary set A of X can also be viewed as a fuzzy set in X with a membership function as its characteristic function, *i.e.*,

$$\mu_A(x) = \begin{cases} 1 & x \in A \\ 0 & x \notin A \end{cases}. \tag{5.1.2}$$

A fuzzy set $\tilde{A}$ can be characterised as a set of ordered pairs of elements x and grade $\mu_{\tilde{A}}(x)$ and is noted

$$\tilde{A} = \{(x, \mu_{\tilde{A}}(x)) \mid x \in X\}. \tag{5.1.3}$$

where each pair $(x, \mu_{\tilde{A}}(x))$ is called a *singleton*.

When X is a countable or finite set, a fuzzy set $\tilde{A}$ on X is expressed as

$$\tilde{A} = \sum_{x_i \in X} \mu(x_i)/x_i\ . \tag{5.1.4}$$

When X is a finite set whose elements are $x_1, x_2, \ldots, x_n$, a fuzzy set $\tilde{A}$ on X is expressed as

$$\tilde{A} = \{(x_1, \mu_{\tilde{A}}(x_1)), (x_2, \mu_{\tilde{A}}(x_2)), \cdots, (x_n, \mu_{\tilde{A}}(x_n))\}. \tag{5.1.5}$$

When X is an infinite and uncountable set, a fuzzy set $\tilde{A}$ on X is expressed as

$$\tilde{A} = \int_X \mu(x)/x. \tag{5.1.6}$$

These expressions mean that the grade of x is $\mu_{\tilde{A}}(x)$ and the operations '+,' 'Σ,' and '∫' do not refer to ordinary addition and integral but they are union, and '/' does not indicate an ordinary division but it is merely a marker.

The following basic notions are defined for fuzzy sets.

Definition 5.1.2 (Support of a fuzzy set) Let $\tilde{A}$ be a fuzzy set on X. Then the *support* of $\tilde{A}$, denoted by supp($\tilde{A}$), is the crisp set given by

$$\text{supp}(\tilde{A}) = \{x \in X \mid \mu_{\tilde{A}}(x) > 0\}. \tag{5.1.7}$$

Definition 5.1.3 (Normal fuzzy set) Let $\tilde{A}$ be a fuzzy set on X. The *height* of $\tilde{A}$, denoted by hgt($\tilde{A}$), is defined as

$$\text{hgt}(\tilde{A}) = \sup_{x \in X} \mu_{\tilde{A}}(x). \tag{5.1.8}$$

If hgt($\tilde{A}$) = 1, then the fuzzy set $\tilde{A}$ is called a *normal fuzzy set*, otherwise it is called *subnormal*.

Definition 5.1.4 (Empty fuzzy set) A fuzzy set $\tilde{A}$ is empty, denoted by $\varnothing$, if $\mu_{\tilde{A}}(x) = 0$ for all $x \in X$.

5.1.2 *Operations and properties*

Definition 5.1.5 (Subset) Let $\tilde{A}$ and $\tilde{B}$ be two fuzzy sets on *X*. The fuzzy set $\tilde{A}$ is called a *subset* of $\tilde{B}$ (or $\tilde{A}$ is contained in $\tilde{B}$), denoted by $\tilde{A} \subset \tilde{B}$, if $\mu_{\tilde{A}}(x) \le \mu_{\tilde{B}}(x)$ for all $x \in X$.

Definition 5.1.6 (Equal) Let $\tilde{A}$ and $\tilde{B}$ be two fuzzy sets on *X*. The fuzzy sets $\tilde{A}$ and $\tilde{B}$ are *equal*, denoted by $\tilde{A} = \tilde{B}$, if $\tilde{B} \subset \tilde{A}$ and $\tilde{A} \subset \tilde{B}$.

Definition 5.1.7 (Union) Let $\tilde{A}$ and $\tilde{B}$ be two fuzzy sets on *X*. The *union* of two fuzzy sets $\tilde{A}$ and $\tilde{B}$, denoted by $\tilde{A} \cup \tilde{B}$, if for all $x \in X$,

$$\begin{aligned} \mu_{\tilde{A} \cup \tilde{B}}(x) &= \max\left(\mu_{\tilde{A}}(x), \mu_{\tilde{B}}(x)\right) \\ &= \mu_{\tilde{A}}(x) \vee \mu_{\tilde{B}}(x). \end{aligned} \tag{5.1.9}$$

Definition 5.1.8 (Intersection) Let $\tilde{A}$ and $\tilde{B}$ be two fuzzy sets on *X*. The *intersection* of two fuzzy sets $\tilde{A}$ and $\tilde{B}$, denoted by $\tilde{A} \cap \tilde{B}$, if for all $x \in X$,

$$\begin{aligned} \mu_{\tilde{A} \cup \tilde{B}}(x) &= \min\left(\mu_{\tilde{A}}(x), \mu_{\tilde{B}}(x)\right) \\ &= \mu_{\tilde{A}}(x) \wedge \mu_{\tilde{B}}(x). \end{aligned} \tag{5.1.10}$$

Definition 5.1.9 (Complementation) Let $\tilde{A}$ be a fuzzy set on *X*. The *complementation* of a fuzzy set $\tilde{A}$, denoted by $\tilde{A}^c$, if for all $x \in X$,

$$\mu_{\tilde{A}^c}(x) = 1 - \mu_{\tilde{A}}(x). \tag{5.1.11}$$

Theorem 5.1.1 Let $\tilde{A}$, $\tilde{B}$ and $\tilde{C}$ be fuzzy sets on *X*. We have
(1) $\varnothing \subset \tilde{A} \subset X$;
(2) (Reflexive law): $\tilde{A} \subset \tilde{A}$;
(3) (Transferability): If $\tilde{A} \subset \tilde{B}$ and $\tilde{B} \subset \tilde{C}$, then $\tilde{A} \subset \tilde{C}$;
(4) (Commutativity law): $\tilde{A} \cup \tilde{B} = \tilde{B} \cup \tilde{A}$ and $\tilde{A} \cap \tilde{B} = \tilde{B} \cap \tilde{A}$;
(5) (Associativity law): $\left(\tilde{A} \cup \tilde{B}\right) \cup \tilde{C} = \tilde{A} \cup \left(\tilde{B} \cup \tilde{C}\right)$ and $\left(\tilde{A} \cap \tilde{B}\right) \cap \tilde{C} = \tilde{A} \cap \left(\tilde{B} \cap \tilde{C}\right)$;

(6) (Distributivity law): $(\tilde{A} \cup \tilde{B}) \cap \tilde{C} = (\tilde{A} \cap \tilde{C}) \cup (\tilde{B} \cap \tilde{C})$, and
$(\tilde{A} \cap \tilde{B}) \cup \tilde{C} = (\tilde{A} \cup \tilde{C}) \cap (\tilde{B} \cup \tilde{C})$;
(7) (Absorption): $(\tilde{A} \cup \tilde{B}) \cap \tilde{A} = \tilde{A}$ and $(\tilde{A} \cap \tilde{B}) \cup \tilde{A} = \tilde{A}$;
(8) (De Morgan's laws): $(\tilde{A} \cup \tilde{B})^c = \tilde{A}^c \cap \tilde{B}^c$ and $(\tilde{A} \cap \tilde{B})^c = \tilde{A}^c \cup \tilde{B}^c$;
(9) (Involution): $(\tilde{A}^c)^c = \tilde{A}$.

It should be noted that the complementarity law and mutually exclusive law are no longer valid for fuzzy sets:

$$\tilde{A} \cap \tilde{A}^c \neq \varnothing \text{ and } \tilde{A} \cup \tilde{A}^c \neq X .$$

5.1.3 *Decomposition theorem and the extension principle*

Definition 5.1.10 (α-cut) Let $\tilde{A}$ be a fuzzy set on X and $\alpha \in [0,1]$. The *α-cut* of the fuzzy set $\tilde{A}$ is the crisp set A_α given by

$$A_\alpha = \{x \in X \mid \mu_{\tilde{A}}(x) \geq \alpha\}. \tag{5.1.12}$$

Theorem 5.1.2 Let $\tilde{A}$ and $\tilde{B}$ be two fuzzy sets on X and $\alpha \in [0,1]$. We have

(1) If $\alpha_1 \leq \alpha_2$, then $A_{\alpha_1} \supset A_{\alpha_2}$;
(2) $(\tilde{A} \cup \tilde{B})_\alpha = A_\alpha \cup B_\alpha$
(3) $(\tilde{A} \cap \tilde{B})_\alpha = A_\alpha \cap B_\alpha$

Theorem 5.1.3 (Decomposition theorem) Let $\tilde{A}$ be a fuzzy set on X with the membership function $\mu_{\tilde{A}}(x)$, A_α be the α-cut of the fuzzy set $\tilde{A}$ and $\chi_{A_\alpha}(x)$ be the characteristic function of the crisp set A_α for $\alpha \in [0,1]$. Then the fuzzy set $\tilde{A}$ can be represented by

$$\tilde{A} = \bigcup_{\alpha \in [0,1]} \alpha A_\alpha, \tag{5.1.13}$$

and its membership function can be represented by

$$\mu_{\tilde{A}}(x) = \sup_{\alpha \in [0.1]} (\alpha \wedge \chi_{A_\alpha}(x)). \tag{5.1.14}$$

Definition 5.1.11 (Convex fuzzy set) A fuzzy set $\tilde{A}$ on R^n is called a *convex fuzzy set* if its α-cuts A_α are the crisp convex sets for all $\alpha \in [0,1]$.

Theorem 5.1.4 A fuzzy set $\tilde{A}$ on R^n is a convex fuzzy set if and only if

$$\mu_{\tilde{A}}(\alpha x_1 + (1-\alpha)x_2) \geq \min\left(\mu_{\tilde{A}}(x_1), \mu_{\tilde{A}}(x_2)\right) \tag{5.1.15}$$

for any $x_1, x_2 \in R^n$ and $\alpha \in [0, 1]$.

Definition 5.1.12 (Bounded fuzzy set) Let $\tilde{A}$ be a fuzzy set on R^n. $\tilde{A}$ is called a *bounded fuzzy set* if its α-cuts A_α are the crisp bounded sets for all $\alpha \in [0, 1]$.

Definition 5.1.13 (Extension principle) Let f: $X \rightarrow Y$ be a mapping from a set X to a set Y. Then the extension principle allows us to define the fuzzy set $\tilde{B}$ on Y induced by the fuzzy set $\tilde{A}$ on X through f as follows:

(1) $\mu_{f(\tilde{A})}(y) = \sup\limits_{\substack{f(x)=y \\ x \in X}} \left(\mu_{\tilde{A}}(x)\right)$,

(2) $\mu_{f^{-1}(\tilde{B})}(x) = \mu_{\tilde{B}}\left(f(x)\right)$.

5.2 Fuzzy Relations

Let X and Y be two crisp sets and $X \times Y$ be the Cartesian product.

Definition 5.2.1 (Fuzzy relation) A fuzzy relation $\tilde{R}$ on $X \times Y$ is defined as

$$\tilde{R} = \left\{\left((x, y), \mu_{\tilde{R}}(x, y)\right) \mid (x, y) \in X \times Y\right\}, \tag{5.2.1}$$

where $\mu_{\tilde{R}} : X \times Y \rightarrow [0, 1]$ is a grade of membership function. If X = Y, then $\tilde{R}$ is called a *fuzzy relation* on X.

Definition 5.2.2 (Fuzzy reflexivity) Let $\tilde{R}$ be a fuzzy relation on the Cartesian product $S = X \times X$. $\tilde{R}$ is called *reflexive* if for all $x \in X$ we have

$$\mu_{\tilde{R}}(x, x) = 1. \tag{5.2.2}$$

If for at least one x in X but not for all x's, (5.2.2) is not true the fuzzy relation $\tilde{R}$ is called *irreflexive*. If (5.2.2) is not satisfied for any x, then $\tilde{R}$ is called *antireflexive*.

Definition 5.2.3 (Fuzzy symmetry) Let $\tilde{R}$ be a fuzzy relation on the Cartesian product $S = X \times X$. $\tilde{R}$ is called *symmetric* if for all x, $y \in X$ we have

$$\mu_{\tilde{R}}(x, y) = \mu_{\tilde{R}}(y, x). \tag{5.2.3}$$

If (5.2.3) is not satisfied for some pairs (x, y), then we say $\tilde{R}$ is antisymmetric. If it is not satisfied for all pairs (x, y), then we say the fuzzy relation $\tilde{R}$ is asymmetric.

Definition 5.2.4 (Fuzzy max-min transitivity) Let $\tilde{R}$ be a fuzzy relation on the Cartesian product $S = X \times X$. $\tilde{R}$ is called *max-min transitive* if for all (x, y), (y, z) $\in X \times X$, we have

$$\mu_{\tilde{R}}(x, z) \geq \vee_{y}\left(\mu_{\tilde{R}}(x, y) \wedge \mu_{\tilde{R}}(y, z)\right), \tag{5.2.4}$$

where all the maxima with respect to y are taken for all the minima inside the brackets in (5.2.4). A transitivity can be defined for other operations such as product (•) instead of min ($\wedge$) in (5.2.4), in such a case we have what is called the *fuzzy max-product transitivity*. In fact, the transitivity can be defined for any triangular conorms (S) and triangular norms (T) instead of max ($\vee$) and min ($\wedge$), respectively, in (5.2.4), it is called the *fuzzy S-T transitivity*. A fuzzy relation that does not satisfy (5.2.4) for all pairs, then we say $\tilde{R}$ is non-transitive. If it fails to satisfy (5.2.4) for all pairs, then it is called *anti-transitive*.

Definition 5.2.5 A fuzzy relation is called a *fuzzy proximity* or *fuzzy tolerance relation* if it is reflexive and symmetric. A fuzzy relation is called a *fuzzy similarity relation* if it is reflexive, symmetric, and transitive.

Definition 5.2.6 (Fuzzy composition) Let $\tilde{A}$ and $\tilde{B}$ be two fuzzy sets on $X \times Y$ and $Y \times Z$, respectively. A fuzzy relation $\tilde{R}$ on $X \times Z$ is defined as

$$\tilde{R} = \left\{\left((x, z), \mu_{\tilde{R}}(x, z)\right) \mid (x, z) \in X \times Z\right\}, \tag{5.2.5}$$

where

$$\begin{aligned} &\mu_{\tilde{R}} : X \times Y \to [0, 1] \\ &\qquad (x, z) \mapsto \mu_{\tilde{R}}(x, z) = \mu_{\tilde{A} \circ \tilde{B}}(x, z) = \underset{y \in Y}{S}\left(T\left(\mu_{\tilde{A}}(x, y), \mu_{\tilde{B}}(y, z)\right)\right) \end{aligned} \tag{5.2.6}$$

for $x \in X$ and $z \in Z$, 'T' and 'S' are triangular norms and triangular conorms, respectively.

5.3 Fuzzy Numbers

Definition 5.3.1 (Fuzzy number) A fuzzy set $\tilde{a}$ on R is called a *fuzzy number* if it satisfies the following conditions:

(1) $\tilde{a}$ is normal, *i.e.*, there exists an $x_0 \in R$ such that $\mu_{\tilde{a}}(x_0) = 1$;
(2) a_α is a closed interval for every $\alpha \in (0, 1]$, noted by $[a_\alpha^L, a_\alpha^R]$;
(3) The support of $\tilde{a}$ is bounded.

Let $F(R)$ be the set of all fuzzy numbers on R. By the decomposition theorem of fuzzy sets, we have

$$\tilde{a} = \bigcup_{\alpha \in [0,1]} \alpha [a_\alpha^L, a_\alpha^R], \tag{5.3.1}$$

for every $\tilde{a} \in F(R)$. For any real number $\lambda \in R$, we define $\mu_\lambda(x)$ by

$$\mu_\lambda(x) = \begin{cases} 1 & \text{iff} \quad x = \lambda \\ 0 & \text{iff} \quad x \neq \lambda \end{cases}.$$

Then $\lambda \in F(R)$.

Definition 5.3.2 A triangular fuzzy number $\tilde{a}$ can be defined by a triplet $\left(a_0^L, a, a_0^R\right)$ and the membership function $\mu_{\tilde{a}}(x)$ is defined as:

$$\mu_{\tilde{a}}(x) = \begin{cases} 0 & x < a_0^L \\ \dfrac{x - a_0^L}{a - a_0^L} & a_0^L \le x \le a \\ \dfrac{a_0^R - x}{a_0^R - a} & a \le x \le a_0^R \\ 0 & a_0^R < x \end{cases}, \tag{5.3.2}$$

where $a = a_1^L = a_1^R$.

Definition 5.3.3 If $\tilde{a}$ is a fuzzy number and $a_\lambda^L > 0$ for any $\lambda \in (0, 1]$, then $\tilde{a}$ is called a *positive fuzzy number*. Let $F_+^*(R)$ be the set of all finite positive fuzzy numbers on R.

Definition 5.3.4 For any $\tilde{a}, \tilde{b} \in F(R)$ and $0 < \alpha \in R$, the sum, scalar product and product of two fuzzy numbers $\tilde{a} + \tilde{b}$, $\tilde{a} - \tilde{b}$, $\alpha\tilde{a}$ and $\tilde{a} \times \tilde{b}$ are defined by the membership functions

$$\mu_{\tilde{a}+\tilde{b}}(t) = \sup_{t=u+v} \min\{\mu_{\tilde{a}}(u), \mu_{\tilde{b}}(v)\}, \tag{5.3.3}$$

$$\mu_{\tilde{a}-\tilde{b}}(t) = \sup_{t=u-v} \min\{\mu_{\tilde{a}}(u), \mu_{\tilde{b}}(v)\}, \tag{5.3.4}$$

$$\mu_{\alpha\tilde{a}}(t) = \sup_{t=\alpha u} \min\{\mu_{\tilde{a}}(u)\}, \tag{5.3.5}$$

$$\mu_{\tilde{a}\times\tilde{b}}(t) = \sup_{t=u\times v} \min\{\mu_{\tilde{a}}(u), \mu_{\tilde{b}}(v)\}, \tag{5.3.6}$$

where we set $\sup\{\phi\} = -\infty$.

Theorem 5.3.1 For any $\tilde{a}, \tilde{b} \in F(R)$ and $0 < \alpha \in R$,

$$\tilde{a} + \tilde{b} = \bigcup_{\lambda\in(0,1]} \lambda[a_\lambda^L + b_\lambda^L, a_\lambda^R + b_\lambda^R],$$

$$\begin{aligned}\tilde{a} - \tilde{b} &= \bigcup_{\lambda\in(0,1]} \lambda[a_\lambda^L - b_\lambda^R, a_\lambda^R - b_\lambda^L] \\ &= \tilde{a} + (-\tilde{b}) \\ &= \bigcup_{\lambda\in(0,1]} \lambda[a_\lambda^L + (-b_\lambda^R), a_\lambda^R + (-b_\lambda^L)],\end{aligned}$$

$$\alpha\tilde{a} = \bigcup_{\lambda\in(0,1]} \lambda[\alpha a_\lambda^L, \alpha a_\lambda^R],$$

$$\tilde{a} \times \tilde{b} = \bigcup_{\lambda\in(0,1]} \lambda[a_\lambda^L \times b_\lambda^L, a_\lambda^R \times b_\lambda^R].$$

Definition 5.3.5 For any $\tilde{a} \in F_+^*(R)$ and $0 < \alpha \in Q_+$ (Q_+ is a set of all positive rational numbers), the positive fuzzy number $\tilde{a}$ power of λ is defined by the membership function

$$\mu_{\tilde{a}^\alpha}(t) = \sup_{t=u^\alpha} \min\{\mu_{\tilde{a}}(u)\} \tag{5.3.7}$$

where we set $\sup\{\phi\} = -\infty$.

Theorem 5.3.2 For any $\tilde{a} \in F_+^*(R)$ and $0 < \alpha \in Q_+$,

$$\tilde{a}^\alpha = \bigcup_{\lambda\in(0,1]} \lambda[(a_\lambda^L)^\alpha, (a_\lambda^R)^\alpha].$$

Definition 5.3.6 Let $\tilde{a}$ and $\tilde{b}$ be two fuzzy numbers. We define

(1) $\tilde{a} \geq \tilde{b}$ iff $a_\lambda^L \geq b_\lambda^L$ and $a_\lambda^R \geq b_\lambda^R$, $\lambda \in (0,1]$;

(2) $\tilde{a} = \tilde{b}$ iff $\tilde{a} \geq \tilde{b}$ and $\tilde{b} \geq \tilde{a}$;

(3) $\tilde{a} > \tilde{b}$ iff $\tilde{a} \geq \tilde{b}$ and there exists a $\lambda_0 \in (0,1]$ such that $a^L_{\lambda_0} > b^L_{\lambda_0}$ or $a^R_{\lambda_0} > b^R_{\lambda_0}$.

Definition 5.3.7 If $\tilde{a}$ is a fuzzy number and $0 < a^L_\lambda \leq a^R_\lambda \leq 1$, for any $\lambda \in (0,1]$, then $\tilde{a}$ is called a *normalised positive fuzzy number*. Let $F^*_N(R)$ be the set of all normalised positive triangular fuzzy numbers on R.

Definition 5.3.8 Let $\tilde{a}, \tilde{b} \in F(R)$. Then the quasi-distance function of $\tilde{a}$ and $\tilde{b}$ is defined as

$$d(\tilde{a}, \tilde{b}) = \left(\int_0^1 \frac{1}{2}[(a^L_\lambda - b^L_\lambda)^2 + (a^R_\lambda - b^R_\lambda)^2] d\lambda \right)^{\frac{1}{2}} \tag{5.3.8}$$

Definition 5.3.9 Let $\tilde{a}, \tilde{b} \in F(R)$. Then the fuzzy number $\tilde{a}$ is said to closer to the fuzzy number $\tilde{b}$ as $d(\tilde{a}, \tilde{b})$ approaches 0.

Proposition 5.3.1 If both $\tilde{a}$ and $\tilde{b}$ are real numbers, then the quasi-distance measurement $d(\tilde{a}, \tilde{b})$ is identical to the Euclidean distance.

Proposition 5.3.2 Let $\tilde{a}, \tilde{b} \in F(R)$. 1) If they are identical, then $d(\tilde{a}, \tilde{b}) = 0$. 2) If $\tilde{a}$ is a real number or $\tilde{b}$ is a real number and $d(\tilde{a}, \tilde{b}) = 0$, then $\tilde{a} = \tilde{b}$.

Proposition 5.3.3 Let $\tilde{a}, \tilde{b}, \tilde{c} \in F(R)$. Then $\tilde{b}$ is closer to $\tilde{a}$ than $\tilde{c}$ if and only if $d(\tilde{b}, \tilde{a}) < d(\tilde{c}, \tilde{a})$.

Proposition 5.3.4 Let $\tilde{a}, \tilde{b} \in F(R)$. If $d(\tilde{a}, 0) < d(\tilde{b}, 0)$ then $\tilde{a}$ is closer to 0 than $\tilde{b}$.

Definition 5.3.10 Let $\tilde{a}_i \in F(R), i = 1,2,\cdots,n.$ We define $\tilde{a} = (\tilde{a}_1, \tilde{a}_2, \cdots, \tilde{a}_n)$

$$\mu_{\tilde{a}} : R^n \to [0,1]$$

$$x \mapsto \bigwedge_{i=1}^{n} \mu_{\tilde{a}_i}(x_i),$$

where $x = (x_1, x_2, \ldots, x_n)^T \in R^n$, and $\tilde{a}$ is called an *n-dimensional fuzzy number* on R^n. If $\tilde{a}_i \in F(R), i = 1,2,\cdots,n,$ $\tilde{a}$ is called an *n-dimensional finite fuzzy number* on R^n.

Let $F(R^n)$ be the set of all n-dimensional fuzzy numbers on R^n.

Proposition 5.3.5 For every $\tilde{a} \in F(R^n)$, $\tilde{a}$ is normal.

Proposition 5.3.6 For every $\tilde{a} \in F(R^n)$, the λ-section of $\tilde{a}$ is an n-dimensional closed rectangular region for any $\lambda \in (0,1]$.

Proposition 5.3.7 For every $\tilde{a} \in F(R^n)$, $\tilde{a}$ is a convex fuzzy set, *i.e.*,

$$\mu_{\tilde{a}}(\lambda x + (1-\lambda)y) \geqq \mu_{\tilde{a}}(x) \wedge \mu_{\tilde{a}}(y),$$

whenever $\lambda \in [0, 1]$, $x = (x_1, x_2, \ldots, x_n)^T$, $y = (y_1, y_2, \ldots, y_n)^T \in R^n$.

Proposition 5.3.8 For every $\tilde{a} \in F(R^n)$ and $\lambda_1, \lambda_2 \in (0, 1]$, if $\lambda_1 \leqq \lambda_2$, then $a_{\lambda_2} \subset a_{\lambda_1}$.

Definition 5.3.11 For any $\tilde{a}, \tilde{b} \in F(R^n)$ and $0 < \alpha \in R$, the sum, scalar product and product of two fuzzy numbers $\tilde{a} + \tilde{b}$, $\tilde{a} - \tilde{b}$, $\alpha\tilde{a}$ and $\tilde{a} \times \tilde{b}$ are defined by the membership functions

$$\mu_{\tilde{a}+\tilde{b}}(x) = \mathop{\wedge}_{i=1}^{n} \mu_{\tilde{a}_i+\tilde{b}_i}(x_i), \tag{5.3.9}$$

$$\mu_{\tilde{a}-\tilde{b}}(x) = \mathop{\wedge}_{i=1}^{n} \mu_{\tilde{a}_i-\tilde{b}_i}(x_i), \tag{5.3.10}$$

$$\mu_{\alpha\tilde{a}}(x) = \mathop{\wedge}_{i=1}^{n} \mu_{\alpha\tilde{a}_{i_i}}(x_i), \tag{5.3.11}$$

$$\mu_{\tilde{a}\times\tilde{b}}(x) = \mathop{\wedge}_{i=1}^{n} \mu_{\tilde{a}_i\times\tilde{b}_i}(x_i). \tag{5.3.12}$$

Definition 5.3.12 For any $\tilde{a} = (\tilde{a}_1, \tilde{a}_2, \cdots, \tilde{a}_n), \tilde{a}_i \in F_+^*(R) (i = 1,2,\cdots,n)$ and $0 < \alpha \in Q_+$,

$$\mu_{\tilde{a}^\alpha}(x) = \mathop{\wedge}_{i=1}^{n} \mu_{\tilde{a}_i^\alpha}(x_i)\}. \tag{5.3.13}$$

Definition 5.3.13 For any n-dimensional fuzzy numbers $\tilde{a}, \tilde{b} \in F(R^n)$, and $\alpha \in (0,1]$ we define

(1) $\tilde{a} \succeq_{\alpha} \tilde{b}$ iff $a_\lambda^L \geqq b_\lambda^L$ and $a_\lambda^R \geqq b_\lambda^R, \lambda \in [\alpha, 1]$;

(2) $\tilde{a} \succeq_{\alpha} \tilde{b}$ iff $a_\lambda^L \geq b_\lambda^L$ and $a_\lambda^R \geq b_\lambda^R, \lambda \in [\alpha, 1]$;

(3) $\tilde{a} \succ_{\alpha} \tilde{b}$ iff $a_\lambda^L > b_\lambda^L$ and $a_\lambda^R > b_\lambda^R, \lambda \in [\alpha, 1]$.

We call the binary relations $\succeq, \succeq$, and $\succ$ a *fuzzy max order*, a *strict fuzzy max order*, and *a strong fuzzy max order*, respectively.

Definition 5.3.14 Let $\tilde{a}, \tilde{b} \in F(R)$ be two fuzzy numbers, the *ranking* of two fuzzy numbers are defined as:

$$\tilde{a} \leq \tilde{b} \text{ if } m(\tilde{a}) < m(\tilde{b}) \tag{5.3.14}$$

or

$$m(\tilde{a}) = m(\tilde{b}) \text{ and } \sigma(\tilde{a}) \geq \sigma(\tilde{b}) \tag{5.3.15}$$

where the mean $m(\tilde{a})$ and the standard deviation $\sigma(\tilde{a})$ are defined as:

$$m(\tilde{a}) = \frac{\int_{s(\tilde{a})} x\tilde{a}(x)dx}{\int_{s(\tilde{a})} \tilde{a}(x)dx} \tag{5.3.16}$$

$$\sigma(\tilde{a}) = \left(\frac{\int_{s(\tilde{a})} x^2\tilde{a}(x)dx}{\int_{s(\tilde{a})} \tilde{a}(x)dx} - (m(\tilde{a})^2 \right)^{1/2} \tag{5.3.17}$$

Where $s(\tilde{a}) = \{x \mid \tilde{a}(x) > 0\}$ is the support of fuzzy number $\tilde{a}$.

For triangular fuzzy number $\tilde{a} = (l, m, n)$,

$$m(\tilde{a}) = \frac{1}{3}(l + m + n) \tag{5.3.18}$$

$$\sigma(\tilde{a}) = \frac{1}{18}(l^2 + m^2 + n^2 - lm - ln - mn) \tag{5.3.19}$$

5.4 Linguistic Variables

Any linguistic description is a formal representation of systems made through fuzzy set theory, fuzzy relations, and fuzzy operators. It offers an alternative to describe and use human languages in related analysis models and systems. Informal linguistic descriptions used by humans in daily life and in the performance of skilled tasks, such as control of industrial facilities, troubleshooting, aircraft landing, decision making, text searching and so on, are usually the starting point for the development of linguistic descriptions.

In the situations mentioned above information cannot be described and assessed precisely in a quantitative manner but may be in a qualitative one. These situations often involve attempting to qualify an event or an object by our human perception, and therefore often lead to use words in natural languages instead of numerical values. For example, in group decision making, an individual's role can be described by using linguistic terms such as *important person*. To express decision makers' judgment for a comparison of a pair of assessment-criteria, '*equally important*' or '*A is more important than B*' could be used. In other cases, precise quantitative information cannot be obtained due to its unavailability or its high computational cost. Hence, an approximate *fuzzy value* can be applicable. For example, when evaluating the satisfactory for a product, terms like *very good*, *good*, *medium*, or *bad* can be used instead of numeric values. Similarly, to express decision makers' preference for an alternative linguistic term such as *low* and *high* could be used.

Since these linguistic terms reflect the uncertainty, inaccuracy and fuzziness of decision makers, fuzzy sets and fuzzy relations are good for modelling linguistic variable deal with qualitative assessments described in a human-like language.

A *linguistic variable* is a quintuple $(X,T(X),U,G,M,)$, where X is the name of the variable, $T(X)$ is the term set, *i.e.*, the set of names of linguistic values of X, U is the universe of discourse, G is the grammar to generate the names and M is a set of semantic rules for associating each X with its meaning.

Linguistic terms have been defined as general as possible, but it is possible to precise their membership function parameters to provide more accuracy in the solution map. For example, to express decision makers' preference for an alternative, five linguistic terms are defined in an *interval ranging* from 0 to 1 and shown in Table 5.1 with their general membership functions. For example, the linguistic term '*High*' can be represented as its membership function as in Fig. 5.1.

Table 5.1: Some definitions of linguistic variable-*Preference*

Linguistic terms	*Membership functions*
Very low	$\bigcup_{\lambda\in[0,1]} \lambda[0, \frac{\sqrt{1-\lambda}}{10}]$
Low	$\bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{\lambda}}{10}, \frac{\sqrt{9-8\lambda}}{10}]$
Medium	$\bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{16\lambda+9}}{10}, \frac{\sqrt{49-24\lambda}}{10}]$
High	$\bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{32\lambda+49}}{10}, \frac{\sqrt{100-19\lambda}}{10}]$
Very high	$\bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{19\lambda+81}}{10}, 1]$

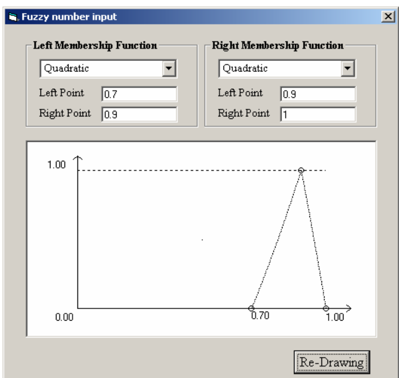


Fig. 5.1: Membership function of the linguistic term '*High*'

The concept of linguistic variables has been applied to handle many kinds of linguistic terms and approximate reasoning in many areas especially in decision-making problems.

5.5 Fuzzy Linear Programming

Zimmermann first introduced fuzzy set theory into conventional linear programming problems in 1976. He considered a linear programming problem with a fuzzy goal and fuzzy constraints. Following the fuzzy decision proposed by Bellman and Zadeh (1970) together with linear membership functions, he proved that there exists an equivalent linear programming problem. Since then, fuzzy linear programming has been developed in number of directions with many successful applications, including fuzzy multi-objective programming, fuzzy bilevel programming, and fuzzy dynamic programming.

5.5.1 *Zimmermann's model*

We introduce an n-dimensional row vector $c = (c_1, c_2, \ldots, c_n)$, an n-dimensional column vector $x = (x_1, x_2, \ldots, x_n)^T$, an n-dimensional column vector $b = (b_1, b_2, \ldots, b_m)^T$, and an $m\times n$ matrix $A = (a_{ij})$, a linear programming problem can be described as follows:

$$\begin{aligned} &\min \;\; z = cx \\ &\text{subject to } Ax \le b \\ &\qquad\qquad\quad x \ge 0. \end{aligned} \tag{5.5.1}$$

In contrast to a conventional linear programming problem, Zimmermann proposed to soften the rigid requirements of decision makers to strictly minimise the objective function and to strictly satisfy the constraints. Namely, when the imprecision or fuzziness of decision makers' judgment is softened the usual linear programming problem (5.5.1) can be covered into the following fuzzy version:

$$\begin{aligned} &z_0 \preceq cx \\ &Ax \preceq b \\ &x \ge 0, \end{aligned} \tag{5.5.2}$$

where the symbol '$\preceq$' denotes a relaxed or fuzzy version of the ordinary inequality '$\le$.' More explicitly, these fuzzy inequalities representing decision makers' fuzzy goal and fuzzy constraints mean that 'the objective function cx should be essentially smaller than or equal to an aspiration level z_0 of decision makers' and 'the constraints Ax should be essentially smaller than or equal to b,' respectively.

5.5.2 *Fuzzy parameters*

In most real-world situations, the possible values of the parameters of linear programming problems are often only imprecisely or ambiguously known to experts when establishing a fuzzy linear programming model. With this observation, it would be certainly more appropriate to interpret experts' understanding of the parameters as fuzzy numerical data, which can be represented by means of fuzzy sets of the real line known as fuzzy numbers. This fuzzy linear programming problem with fuzzy parameters model is

$$\begin{aligned} &\min \quad z = \tilde{c}x \\ &\text{subject to } \tilde{A}x \preceq \tilde{b} \\ &\qquad\qquad x \geq 0, \end{aligned} \tag{5.5.3}$$

where $\tilde{c}$ is an n-dimensional row fuzzy vector $\tilde{c} = (\tilde{c}_1, \tilde{c}_2, \cdots, \tilde{c}_n)$, an n-dimensional column vector $x = (x_1, x_2, \ldots, x_n)^T$, an n-dimensional column fuzzy vector $\tilde{b} = (\tilde{b}_1, \tilde{b}_2, \cdots, \tilde{b}_m)^T$, and an $m \times n$ fuzzy matrix $\tilde{A} = (\tilde{a}_{ij})$. The symbol '$\preceq$' denotes a fuzzy ordinary relation between two fuzzy numbers.

5.6 Summary

This chapter introduced fuzzy sets related concepts, which will be used in the rest chapters of this book. Linguistic terms such as *high* and *low*, *more* or *less* discussed in this chapter can be used for fuzzy multi-attribute decision making. The fuzzy linear programming will be applied in fuzzy multi-objective decision making. For any related approaches to solve fuzzy linear programming problems, we refer the relevant references at the end of the book.

This page intentionally left blank

Part II

Fuzzy Multi-Objective Decision Making

This page intentionally left blank

Chapter 6

Fuzzy MODM Models

Many decision problems are involved in multiple objectives, called *multi-objective decision making* (MODM). Most MODM problems can be formulated by *multi-objective linear programming* (MOLP) models. Referring to the imprecision and insufficient inherent in human judgments, uncertainties may be affected and incorporated in the parameters of an MOLP model, which is called a *Fuzzy MOLP* (FMOLP) model. Uncertainties are also involved in the goals of decision makers for their multiple objectives, called *fuzzy multi-objective linear goal programming* (FMOLGP).

In this chapter, we first illustrate what is an FMOLP problem. We then give a general FMOLP model in which fuzzy parameters of objective functions and constraints are described by membership functions. To solve such an FMOLP problem, we propose an optimal solution concept of FMOLP. Importantly, we develop a general solution transformation theorem and a set of related workable solution transformation theorems. Based on these theorems, we obtain an optimal solution of the FMOLP by solving an associated MOLP problem. We further introduce an FMOLGP model and its related theorems. We will apply these models and theorems in Chapter 7 to develop related methods to get an optimal solution for the FMOLP problem.

6.1 A Problem

As the example described in Chapter 2, a manufacturing company has six machine types - milling machine, lathe, grinder, jig saw, drill press, and band saw - whose capacities are to be devoted to produce three products

x_1, x_2, and x_3. Decision makers have three objectives of maximising profits, quality, and worker satisfaction. When formulating the problem, various uncertain factors of the real world system will determine the parameters of objective functions and constraints in the MOLP model by the experts. Naturally, the parameters of its objective functions and constraints are assigned with some uncertainties, expressed by fuzzy numbers. As shown in Table 6.1, for example, to produce one unit of x_1 needs about 12 hours of milling machine and about 3 hours of lathe.

Table 6.1: Production planning data

Machine	*Product x_1 (unit)*	*Product x_2 (unit)*	*Product x_3 (unit)*	*Machine (available hours)*
Milling machine	About 12	About 17	About 0	About 1400
Lathe	About 3	About 9	About 8	About 1000
Grinder	About 10	About 13	About 15	About 1750
Jig saw	About 6	About 0	About 16	About 1325
Drill press	About 0	About 12	About 7	About 900
Band saw	About 9.5	About 9.5	About 4	About 1075
Profits	About 50	About 100	About 17.5	
Quality	About 92	About 75	About 50	
Worker Satisfaction	About 25	About 100	About 75	

Therefore, with these imprecise values, this problem can be described by an FMOLP model as follows:

$$\max \begin{pmatrix} \tilde{f}_1(x) \\ \tilde{f}_2(x) \\ \tilde{f}_3(x) \end{pmatrix} = \max \begin{pmatrix} \tilde{50}\, x_1 + \tilde{100}\, x_2 + \tilde{17.5}\, x_3 \\ \tilde{92}\, x_1 + \tilde{75}\, x_2 + \tilde{50}\, x_3 \\ \tilde{25}\, x_1 + \tilde{100}\, x_2 + \tilde{75}\, x_3 \end{pmatrix} \tag{6.1.1}$$

$$\text{s.t.} \begin{cases} \tilde{g}_1(x) = \tilde{12}\,x_1 + \tilde{17}\,x_2 \precsim \tilde{1400} \\ \tilde{g}_2(x) = \tilde{3}\,x_1 + \tilde{9}\,x_2 + \tilde{8}\,x_3 \precsim \tilde{1000} \\ \tilde{g}_3(x) = \tilde{10}\,x_1 + \tilde{13}\,x_2 + \tilde{15}\,x_3 \precsim \tilde{1750} \\ \tilde{g}_4(x) = \tilde{6}\,x_1 + \tilde{16}\,x_3 \precsim \tilde{1325} \\ \tilde{g}_5(x) = \tilde{12}\,x_2 + \tilde{7}\,x_3 \precsim \tilde{900} \\ \tilde{g}_6(x) = \tilde{9.5}\,x_1 + \tilde{9.5}\,x_2 + \tilde{4}\,x_3 \precsim \tilde{1075} \\ x_1, x_2, x_3 \geq 0 \end{cases}$$

Here, $\tilde{\alpha}$ means '*about* α,' for example, $\tilde{50}$ represents '*about* 50.' We can see that all parameters of objective functions and constraints are fuzzy numbers. Obviously, a real number is a special case of a fuzzy number. In following parts, the term '*fuzzy parameters*' contains the case of '*real numbers*.'

6.2 Fuzzy Parameter-Based MOLP Models

6.2.1 *A general FMOLP model*

Consider a situation in which all parameters of the objective functions and the constraints are fuzzy numbers represented in any form of membership functions. Such an FMOLP problem can be formulated as follows, in general.

$$\text{(FMOLP)} \begin{cases} \max & \langle \tilde{c}, x \rangle_F = \left(\sum_{i=1}^{n} \tilde{c}_{1i} x_i, \sum_{i=1}^{n} \tilde{c}_{2i} x_i, \cdots, \sum_{i=1}^{n} \tilde{c}_{ki} x_i \right)^T \\ \text{s.t.} & \tilde{A}x \precsim \tilde{b},\ x \geqq 0, \end{cases} \tag{6.2.1}$$

where

$$\tilde{c} = \begin{pmatrix} \tilde{c}_{11} & \tilde{c}_{12} & \cdots & \tilde{c}_{1n} \\ \tilde{c}_{21} & \tilde{c}_{22} & \cdots & \tilde{c}_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{c}_{k1} & \tilde{c}_{k2} & \cdots & \tilde{c}_{kn} \end{pmatrix}, \tilde{A} = \begin{pmatrix} \tilde{a}_{11} & \tilde{a}_{12} & \cdots & \tilde{a}_{1n} \\ \tilde{a}_{21} & \tilde{a}_{22} & \cdots & \tilde{a}_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{a}_{m1} & \tilde{a}_{m2} & \cdots & \tilde{a}_{mn} \end{pmatrix},$$

$$\tilde{b} = (\tilde{b}_1, \tilde{b}_2, \cdots, \tilde{b}_m)^T \in F^*(R^m),$$

and $\tilde{c}_{sj}$, $\tilde{a}_{ij} \in F^*(R)$, $s = 1, 2, \cdots, k$, $i = 1, 2, \cdots, m$, $j = 1, 2, \cdots, n$.

For the sake of simplicity, we set $\tilde{X} = \{x; \tilde{A}x \preceq \tilde{b}, x \geqq 0\}$ and assume that $\tilde{X}$ is compact. In the FMOLP problem, for each $x \in \tilde{X}$, the value of the objective function $\langle \tilde{c}, x \rangle_F$ is a fuzzy number. Thus, we introduce the following concepts of optimal solutions to the FMOLP problems.

Definition 6.2.1 A point $x^* \in R^n$ is called a *complete optimal solution* to the FMOLP problem if it holds that $\langle \tilde{c}, x^* \rangle_F \succeq \langle \tilde{c}, x \rangle_F$ for all $x \in \tilde{X}$.

Definition 6.2.2 A point $x^* \in R^n$ is called a *Pareto optimal solution* to the FMOLP problem if there is no $x \in \tilde{X}$ such that $\langle \tilde{c}, x \rangle_F \succeq \langle \tilde{c}, x^* \rangle_F$ holds.

Definition 6.2.3 A point $x^* \in R^n$ is called a *weak Pareto optimal solution* to the FMOLP problem if there is no $x \in \tilde{X}$ such that $\langle \tilde{c}, x \rangle_F \succ \langle \tilde{c}, x^* \rangle_F$ holds.

The efficient approach for solving the FMOLP problem is to transform it into an associative crisp programming. As normal MOLP problems have been well studied, the main idea here is to define an associative MOLP and then setting up the relationship between the solution of FMOLP (6.2.1) and the solution of the associated MOLP problem. By the definition of the *Pareto optimal solution* of the MOLP, other related methods can be designed and developed for solving the FMOLP problem.

We consider the associated MOLP problem with the FMOLP problem as follows:

$$(\text{MOLP}) \begin{cases} \max & \left(\langle c_\lambda^L, x \rangle, \langle c_\lambda^R, x \rangle \right)^T, \\ \text{s.t.} & A_\lambda^L x \leqq b_\lambda^L,\ A_\lambda^R x \leqq b_\lambda^R,\ x \geqq 0,\ \forall \lambda \in [0,1] \end{cases} \tag{6.2.2}$$

where

$$C_\lambda^L = \begin{bmatrix} c_{11\lambda}^L & c_{12\lambda}^L & \cdots & c_{1n\lambda}^L \\ c_{21\lambda}^L & c_{22\lambda}^L & \cdots & c_{2n\lambda}^L \\ \vdots & \vdots & \ddots & \vdots \\ c_{k1\lambda}^L & c_{k2\lambda}^L & \cdots & c_{kn\lambda}^L \end{bmatrix}, \quad C_\lambda^R = \begin{bmatrix} c_{11\lambda}^R & c_{12\lambda}^R & \cdots & c_{1n\lambda}^R \\ c_{21\lambda}^R & c_{22\lambda}^R & \cdots & c_{2n\lambda}^R \\ \vdots & \vdots & \ddots & \vdots \\ c_{k1\lambda}^R & c_{k2\lambda}^R & \cdots & c_{kn\lambda}^R \end{bmatrix},$$

$$A_\lambda^L = \begin{bmatrix} a_{11\lambda}^L & a_{12\lambda}^L & \cdots & a_{1n\lambda}^L \\ a_{21\lambda}^L & a_{22\lambda}^L & \cdots & a_{2n\lambda}^L \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1\lambda}^L & a_{m2\lambda}^L & \cdots & a_{mn\lambda}^L \end{bmatrix}, A_\lambda^R = \begin{bmatrix} a_{11\lambda}^R & a_{12\lambda}^R & \cdots & a_{1n\lambda}^R \\ a_{21\lambda}^R & a_{22\lambda}^R & \cdots & a_{2n\lambda}^R \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1\lambda}^R & a_{m2\lambda}^R & \cdots & a_{mn\lambda}^R \end{bmatrix},$$

$$b_\lambda^L = \left[b_{1\lambda}^L, b_{2\lambda}^L, \cdots, b_{m\lambda}^L\right]^T, \qquad b_\lambda^R = \left[b_{1\lambda}^R, b_{2\lambda}^R, \cdots, b_{m\lambda}^R\right]^T.$$

In the following, we introduce the concepts of optimal solutions of the MOLP problem.

Definition 6.2.4 A point $x^* \in R^n$ is called a *complete optimal solution* to the MOLP problem if it holds that $(\langle c_\lambda^L, x^* \rangle, \ \langle c_\lambda^R, x^* \rangle)^T \geqq (\langle c_\lambda^L, x \rangle, \ \langle c_\lambda^R, x \rangle)^T$, for all $x \in X = \{x;\, A_\lambda^L x \leqq b_\lambda^L,\, A_\lambda^R x \leqq b_\lambda^R,\, x \geqq 0,\, \lambda \in [0,1]\}$ and $\lambda \in [0,1]$.

Definition 6.2.5 A point $x^* \in R^n$ is called a *Pareto optimal solution* to the MOLP problem if there is no $x \in X$ such that $(\langle c_\lambda^L, x^* \rangle, \ \langle c_\lambda^R, x^* \rangle)^T \leq (\langle c_\lambda^L, x \rangle, \langle c_\lambda^R, x \rangle)^T$, $\lambda \in [0,1]$ holds.

Definition 6.2.6 A point $x^* \in R^n$ is called a *weak Pareto optimal solution* to the MOLP problem if there is no $x \in \widetilde{X}$ such that $(\langle c_\lambda^L, x^* \rangle, \ \langle c_\lambda^R, x^* \rangle)^T < (\langle c_\lambda^L, x \rangle, \langle c_\lambda^R, x \rangle)^T$, $\lambda \in [0,1]$ holds.

Theorem 6.2.1 Let $x^* \in R^n$ be a *feasible solution* to the FMOLP problem. Then

(1) x^* is a complete optimal solution to the FMOLP problem, if and only if x^* is a complete optimal solution to the MOLP problem.

- x^* is a Pareto optimal solution to the FMOLP problem, if and only if x^* is a Pareto optimal solution to the MOLP problem.
- x^* is a weak Pareto optimal solution to the FMOLP problem, if and only if x^* is a weak Pareto optimal solution to the MOLP problem.

Proof: The proof is obvious from Definitions 6.2.1 - 6.2.6.

6.2.2 *An FMOLP$_\alpha$ model*

A feasible solution must satisfy the constraints for all $\lambda \in [0,1]$. However, in general, this is a too strong condition to get an optimal solution. Now we consider a typical parameter c_i represented by a fuzzy number $\tilde{c}_i$. The possibility of such a parameter c_i taking values in the range $[c_{i\lambda}^L, c_{i\lambda}^R]$ is λ or above. While the possibility of c_i taking values beyond $[c_{i\lambda}^L, c_{i\lambda}^R]$ is less than λ. Thus, one would be generally more interested in a solution using parameters c_i taking values in $[c_{i\lambda}^L, c_{i\lambda}^R]$ with $\lambda \geq \alpha > 0$. As a special case, if the parameters involved are either a real number of a fuzzy number with a triangular membership function, then, we will have the usual non-fuzzy optimisation problem (*e.g.*, α = 1). To formulate this idea, we introduce the following FMOLP$_\alpha$ model.

$$(\text{FMOLP}_\alpha)\begin{cases}\max & \langle \tilde{c}, x\rangle_F = \sum_{i=1}^{n} \tilde{c}_i x_i \\ \text{s.t.} & \tilde{A}x \underset{=\alpha}{\prec} \tilde{b},\ x \underset{=}{>} 0,\end{cases} \quad 0 \leq \alpha \leq 1 \tag{6.2.3}$$

where

$$\tilde{c} = \begin{pmatrix} \tilde{c}_{11} & \tilde{c}_{12} & \cdots & \tilde{c}_{1n} \\ \tilde{c}_{21} & \tilde{c}_{22} & \cdots & \tilde{c}_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{c}_{k1} & \tilde{c}_{k2} & \cdots & \tilde{c}_{kn} \end{pmatrix}, \tilde{A} = \begin{pmatrix} \tilde{a}_{11} & \tilde{a}_{12} & \cdots & \tilde{a}_{1n} \\ \tilde{a}_{21} & \tilde{a}_{22} & \cdots & \tilde{a}_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{a}_{m1} & \tilde{a}_{m2} & \cdots & \tilde{a}_{mn} \end{pmatrix},$$

$$\tilde{b} = (\tilde{b}_1, \tilde{b}_2, \cdots, \tilde{b}_m)^T \in F^*(R^m),$$

and $\tilde{c}_{sj}, \tilde{a}_{ij} \in F^*(R)$, $s = 1, 2, \cdots, k$, $i = 1, 2, \cdots, m$, $j = 1, 2, \cdots, n$.

Associated with the FMOLP$_\alpha$ problem, we consider the following MOLP$_\alpha$ problem:

$$(\text{MOLP}_\alpha)\begin{cases}\max & \left(\langle c_\lambda^L, x\rangle, \langle c_\lambda^R, x\rangle\right)^T, \\ \text{s.t.} & A_\lambda^L x \underset{=}{<} b_\lambda^L,\ A_\lambda^R x \underset{=}{<} b_\lambda^R,\ x \underset{=}{>} 0,\ \forall \lambda \in [\alpha, 1]\end{cases} \tag{6.2.4}$$

where

$$C_\lambda^L = \begin{bmatrix} c_{11\lambda}^L & c_{12\lambda}^L & \cdots & c_{1n\lambda}^L \\ c_{21\lambda}^L & c_{22\lambda}^L & \cdots & c_{2n\lambda}^L \\ \vdots & \vdots & \ddots & \vdots \\ c_{k1\lambda}^L & c_{k2\lambda}^L & \cdots & c_{kn\lambda}^L \end{bmatrix}, \quad C_\lambda^R = \begin{bmatrix} c_{11\lambda}^R & c_{12\lambda}^R & \cdots & c_{1n\lambda}^R \\ c_{21\lambda}^R & c_{22\lambda}^R & \cdots & c_{2n\lambda}^R \\ \vdots & \vdots & \ddots & \vdots \\ c_{k1\lambda}^R & c_{k2\lambda}^R & \cdots & c_{kn\lambda}^R \end{bmatrix},$$

$$A_\lambda^L = \begin{bmatrix} a_{11\lambda}^L & a_{12\lambda}^L & \cdots & a_{1n\lambda}^L \\ a_{21\lambda}^L & a_{22\lambda}^L & \cdots & a_{2n\lambda}^L \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1\lambda}^L & a_{m2\lambda}^L & \cdots & a_{mn\lambda}^L \end{bmatrix}, \quad A_\lambda^R = \begin{bmatrix} a_{11\lambda}^R & a_{12\lambda}^R & \cdots & a_{1n\lambda}^R \\ a_{21\lambda}^R & a_{22\lambda}^R & \cdots & a_{2n\lambda}^R \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1\lambda}^R & a_{m2\lambda}^R & \cdots & a_{mn\lambda}^R \end{bmatrix},$$

$$b_\lambda^L = \left[b_{1\lambda}^L, b_{2\lambda}^L, \cdots, b_{m\lambda}^L\right]^T, \qquad b_\lambda^R = \left[b_{1\lambda}^R, b_{2\lambda}^R, \cdots, b_{m\lambda}^R\right]^T.$$

Now, we introduce the concepts of optimal solutions of the MOLP_α problem.

Definition 6.2.7 A point $x^* \in R^n$ is called a *complete optimal solution* to the FMOLP problem if it holds that $\langle \tilde{c}, x^* \rangle_F \succeq_\alpha \langle \tilde{c}, x \rangle_F$ for all $x \in \tilde{X}_\alpha$.

Definition 6.2.8 A point $x^* \in R^n$ is called a *Pareto optimal solution* to the FMOLP problem if there is no $x \in \tilde{X}_\alpha$ such that $\langle \tilde{c}, x \rangle_F \succeq_\alpha \langle \tilde{c}, x^* \rangle_F$ holds.

Definition 6.2.9 A point $x^* \in R^n$ is called a *weak Pareto optimal solution* to the FMOLP problem if there is no $x \in \tilde{X}_\alpha$ such that $\langle \tilde{c}, x \rangle_F \succ_\alpha \langle \tilde{c}, x^* \rangle_F$ holds.

Definition 6.2.10 A point $x^* \in R^n$ is called a *complete optimal solution* to the MOLP_α problem if it holds that $(\langle c_\lambda^L, x^* \rangle, \langle c_\lambda^R, x^* \rangle)^T \geqq (\langle c_\lambda^L, x \rangle, \langle c_\lambda^R, x \rangle)^T$ for all $x \in X_\alpha = \{x;\ A_\lambda^L x \leqq b_\lambda^L,\ A_\lambda^R x \leqq b_\lambda^R,\ x \geqq 0,\ \lambda \in [\alpha, 1]\}$ and $\lambda \in [\alpha, 1]$.

Definition 6.2.11 A point $x^* \in R^n$ is called a *Pareto optimal solution* to the MOLP_α problem if there is no $x \in X_\alpha$ such that $(\langle c_\lambda^L, x^* \rangle, \langle c_\lambda^R, x^* \rangle)^T \leq (\langle c_\lambda^L, x \rangle, \langle c_\lambda^R, x \rangle)^T$, $\lambda \in [\alpha, 1]$ holds.

Definition 6.2.12 A point $x^* \in R^n$ is called a *weak Pareto optimal solution* to the MOLP_α problem if there is no $x \in X_\alpha$ such that $(\langle c_\lambda^L, x^* \rangle, \langle c_\lambda^R, x^* \rangle)^T < (\langle c_\lambda^L, x \rangle, \langle c_\lambda^R, x \rangle)^T$, $\lambda \in [\alpha, 1]$ holds.

Theorem 6.2.2 Let $x^* \in R^n$ be a feasible solution to the FMOLP_α problem. Then

- x^* is a complete optimal solution to the FMOLP$_\alpha$ problem, if and only if x^* is a complete optimal solution to the MOLP$_\alpha$ problem.
- x^* is a Pareto optimal solution to the FMOLP$_\alpha$ problem, if and only if x^* is a Pareto optimal solution to the MOLP$_\alpha$ problem.
- x^* is a weak Pareto optimal solution to the FMOLP$_\alpha$ problem, if and only if x^* is a weak Pareto optimal solution to the MOLP$_\alpha$ problem.

Proof: The proof follows from Definitions 6.2.7-6.2.12 and Theorem 6.2.1.

Based on these definitions and relationships proposed, we will develop the solution transformation theorems in the next section.

6.3 Solution Transformation Theories

This section gives a workable approach to transform an FMOLP problem, with any form of fuzzy numbers as parameters, to an MOLP problem, then to solve it through solving the associated MOLP problem.

6.3.1 *General MOLP transformation*

Lemma 6.3.1 If a fuzzy set $\tilde{c}$ on R has a trapezoidal membership function with Fig. 6.1:

$$\mu_{\tilde{c}}(x) = \begin{cases} 0 & x < c_\beta^L \\ \dfrac{\alpha-\beta}{c_\alpha^L - c_\beta^L}\left(x - c_\beta^L\right) + \beta & c_\beta^L \leqq x < c_\alpha^L \\ \alpha & c_\alpha^L \leqq x \leqq c_\alpha^R \\ \dfrac{\alpha-\beta}{c_\alpha^R - c_\beta^R}\left(x - c_\beta^R\right) + \beta & c_\alpha^R < x \leqq c_\beta^R \\ 0 & c_\beta^R < x \end{cases}$$

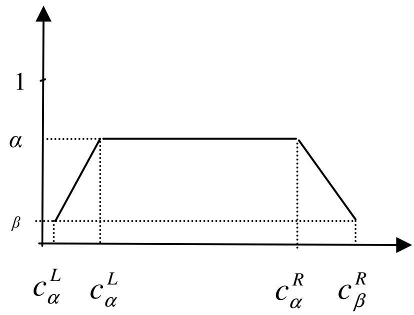


Fig. 6.1: Trapezoidal membership function

and there exists an $x^* \in X^n$ such that $\langle c_\beta^L, x\rangle \leqq \langle c_\beta^L, x^*\rangle$, $\langle c_\alpha^L, x\rangle \leqq \langle c_\alpha^L, x^*\rangle$, $\langle c_\beta^R, x\rangle \leqq \langle c_\beta^R, x^*\rangle$, $(0 \le \beta < \alpha \le 1)$ and $\langle c_\alpha^R, x\rangle \leqq \langle c_\alpha^R, x^*\rangle$, for any $x \in X^n$, then

$$\langle c_\lambda^L, x\rangle \leqq \langle c_\lambda^L, x^*\rangle,$$
$$\langle c_\lambda^R, x\rangle \leqq \langle c_\lambda^R, x^*\rangle,$$

for any $\lambda \in [\beta, \alpha]$.

Proof. As λ-sections of a trapezoidal fuzzy set $\tilde{c}$ are

$$c_\lambda^L = \frac{\lambda-\beta}{\alpha-\beta}\left(c_\alpha^L - c_\beta^L\right) + c_\beta^L \text{ and } c_\lambda^R = \frac{\lambda-\beta}{\alpha-\beta}\left(c_\alpha^R - c_\beta^R\right) + c_\beta^R.$$

Therefore, we have

$$\begin{aligned}
\langle c_\lambda^L, x\rangle &= \left\langle \frac{\lambda-\beta}{\alpha-\beta}\left(c_\alpha^L - c_\beta^L\right), x\right\rangle + \langle c_\beta^L, x\rangle \\
&= \frac{\lambda-\beta}{\alpha-\beta}\left(\langle c_\alpha^L, x\rangle - \langle c_\beta^L, x\rangle\right) + \langle c_\beta^L, x\rangle \\
&= \frac{\lambda-\beta}{\alpha-\beta}\langle c_\alpha^L, x\rangle + \frac{\alpha-\lambda}{\alpha-\beta}\langle c_\beta^L, x\rangle \\
&\leqq \frac{\lambda-\beta}{\alpha-\beta}\langle c_\alpha^L, x^*\rangle + \frac{\alpha-\lambda}{\alpha-\beta}\langle c_1^L, x^*\rangle \\
&= \left\langle \frac{\lambda-\beta}{\alpha-\beta}\left(c_\alpha^L - c_\beta^L\right), x^*\right\rangle + \langle c_\beta^L, x^*\rangle = \langle c_\lambda^L, x^*\rangle,
\end{aligned}$$

from $\langle c_1^L, x\rangle \leqq \langle c_1^L, x^*\rangle$, $\langle c_\alpha^L, x\rangle \leqq \langle c_\alpha^L, x^*\rangle$ and $0 \le \beta \le \lambda \le \alpha \le 1$, we can prove $\langle c_\lambda^R, x\rangle \leqq \langle c_\lambda^R, x^*\rangle$, from the similar reason.

Theorem 6.3.1 If each of the fuzzy parameters $\tilde{c}_{sj}$, $\tilde{a}_{ij}$ and $\tilde{b}_i$ has a trapezoidal membership function:

$$\mu_{\tilde{z}}(t)=\begin{cases}0 & t<z_\beta^L \\ \dfrac{\alpha-\beta}{z_\alpha^L-z_\beta^L}\left(t-z_\beta^L\right)+\beta & z_\beta^L \leqq t<z_\alpha^L \\ \alpha & z_\alpha^L \leqq t<z_\alpha^R \\ \dfrac{\alpha-\beta}{z_\beta^R-z_\alpha^R}\left(-t+z_\beta^R\right)+\beta & z_\alpha^R \leqq t \leqq z_\beta^R \\ 0 & z_\beta^R<t\end{cases}, \tag{6.3.1}$$

where $\tilde{z}$ denotes $\tilde{c}_{sj}$, $\tilde{a}_{ij}$ or $\tilde{b}_i$ respectively, then the space of feasible solutions X is defined by the set of $x \in X$ with x_i, for $i = 1,2,\ldots,n$, satisfying

$$\begin{cases}\sum_{j=1}^{n} a_{ij\alpha}^{L} x_j \leqq b_{i\alpha}^{L} \\ \sum_{j=1}^{n} a_{ij\alpha}^{R} x_j \leqq b_{i\alpha}^{R} \\ \sum_{j=1}^{n} a_{ij\beta}^{L} x_j \leqq b_{i\beta}^{L} \\ \sum_{j=1}^{n} a_{ij\beta}^{R} x_j \leqq b_{i\beta}^{R} \\ x_i \geqq 0\end{cases}. \tag{6.3.2}$$

Proof. From Theorem 6.2.1, X is defined by

$$X=\{x\in R^n \mid \sum_{j=1}^{n} a_{ij\lambda}^{L}x_j \leqq b_{i\lambda}^{L}, \sum_{j=1}^{n} a_{ij\lambda}^{R}x_j \leqq b_{i\lambda}^{R}, x \geqq 0, \forall\lambda\in[\beta,\alpha] \text{ and } i=1,2,\cdots,m\} \tag{6.3.3}$$

That is, X is the set of $x \in R^n$ with $x \geqq 0$ satisfying

$$I_{i\lambda}=\sum_{j=1}^{n} a_{ij\lambda}^{L}x_j-b_{i\lambda}^{L}\leqq 0,\ J_{i\lambda}=\sum_{j=1}^{n} a_{ij\lambda}^{R}x_j-b_{i\lambda}^{R}\leqq 0,\ \forall\lambda\in[\beta,\alpha] \text{ and } i=1,2,\cdots,m. \tag{6.3.4}$$

For fuzzy sets with trapezoidal membership functions, we have

$$a_{ij\lambda}^{L}=\frac{a_{ij\alpha}^{L}-a_{ij\beta}^{L}}{\alpha-\beta}(\lambda-\beta)+a_{ij\beta}^{L},\quad a_{ij\lambda}^{R}=\frac{a_{ij\alpha}^{R}-a_{ij\beta}^{R}}{\alpha-\beta}(\lambda-\beta)+a_{ij\beta}^{R}, \tag{6.3.5}$$

$$b_{i\lambda}^{L}=\frac{b_{i\alpha}^{L}-b_{i\beta}^{L}}{\alpha-\beta}(\lambda-\beta)+b_{i\beta}^{L},\quad b_{i\lambda}^{R}=\frac{b_{i\alpha}^{R}-b_{i\beta}^{R}}{\alpha-\beta}(\lambda-\beta)+b_{i\beta}^{R}, \tag{6.3.6}$$

Substituting (6.3.5) and (6.3.6) into (6.3.4), we have

$$I_{i\lambda} = \sum_{j=1}^{n} [\frac{a_{ij\alpha}^{L} - a_{ij\beta}^{L}}{\alpha - \beta}(\lambda - \beta) + a_{ij\beta}^{L}]x_j - [\frac{b_{i\alpha}^{L} - b_{i\beta}^{L}}{\alpha - \beta}(\lambda - \beta) + b_{i\beta}^{L}], \tag{6.3.7}$$
$$= \frac{\lambda - \beta}{\alpha - \beta}\left(\sum_{j=1}^{n} a_{ij\alpha}^{L} x_j - b_{i\alpha}^{L}\right) + \frac{\alpha - \lambda}{\alpha - \beta}\left(\sum_{j=1}^{n} a_{ij\beta}^{L} x_j - b_{i\beta}^{L}\right)$$

$$J_{i\lambda} = \sum_{j=1}^{n} [\frac{a_{ij\alpha}^{R} - a_{ij\beta}^{R}}{\alpha - \beta}(\lambda - \beta) + a_{ij\beta}^{R}]x_j - [\frac{b_{i\alpha}^{R} - b_{i\beta}^{R}}{\alpha - \beta}(\lambda - \beta) + b_{i\beta}^{R}], \tag{6.3.8}$$
$$= \frac{\lambda - \beta}{\alpha - \beta}\left(\sum_{j=1}^{n} a_{ij\alpha}^{R} x_j - b_{i\alpha}^{R}\right) + \frac{\alpha - \lambda}{\alpha - \beta}\left(\sum_{j=1}^{n} a_{ij\beta}^{R} x_j - b_{i\beta}^{R}\right)$$

Now, our problem becomes to show that $I_{i\lambda} \leqq 0, J_{i\lambda} \leqq 0, \forall \lambda \in [\beta, \alpha]$ and $i = 1,2,\cdots,m$ if (6.3.2) is satisfied. From (6.3.2),

$$\begin{cases} \sum_{j=1}^{n} a_{ij\alpha}^{L} x_j - b_{i\alpha}^{L} \leqq 0 & (6.3.9\,\text{a}) \\ \sum_{j=1}^{n} a_{ij\alpha}^{R} x_j - b_{i\alpha}^{R} \leqq 0 & (6.3.9\,\text{b}) \\ \sum_{j=1}^{n} a_{ij\beta}^{L} x_j - b_{i\beta}^{L} \leqq 0 & (6.3.9\,\text{c}) \\ \sum_{j=1}^{n} a_{ij\beta}^{R} x_j - b_{i\beta}^{R} \leqq 0. & (6.3.9\,\text{d}) \end{cases}$$

Thus, from (6.3.9a) and (6.3.9c), we have, for any $\lambda \in [\beta, \alpha]$ and $i = 1,2,\ldots,m$,

$$I_{i\lambda} = \frac{\lambda - \beta}{\alpha - \beta}\left(\sum_{j=1}^{n} a_{ij\alpha}^{L} x_j - b_{i\alpha}^{L}\right) + \frac{\alpha - \lambda}{\alpha - \beta}\left(\sum_{j=1}^{n} a_{ij\beta}^{L} x_j - b_{i\beta}^{L}\right) \leqq 0$$

and from (6.3.9b) and (6.3.9d), we have, for any $\lambda \in [\beta, \alpha]$ and $i = 1,2,\ldots,m$,

$$J_{i\lambda} = \frac{\lambda - \beta}{\alpha - \beta}\left(\sum_{j=1}^{n} a_{ij\alpha}^{R} x_j - b_{i\alpha}^{R}\right) + \frac{\alpha - \lambda}{\alpha - \beta}\left(\sum_{j=1}^{n} a_{ij\beta}^{R} x_j - b_{i\beta}^{R}\right) \leqq 0.$$

Corollary 6.3.1 If all the fuzzy parameters $\tilde{c}_{sj}$, $\tilde{a}_{ij}$ and $\tilde{b}_i$ have a piecewise trapezoidal membership function

$$\mu_{\tilde{z}}(t) = \begin{cases} 0 & t < z^L_{\alpha_0} \\ \dfrac{\alpha_1 - \alpha_0}{z^L_{\alpha_1} - z^L_{\alpha_0}}\left(t - z^L_{\alpha_0}\right) + \alpha_0 & z^L_{\alpha_0} \leqq t < z^L_{\alpha_1} \\ \dfrac{\alpha_2 - \alpha_1}{z^L_{\alpha_2} - z^L_{\alpha_1}}\left(t - z^L_{\alpha_1}\right) + \alpha_1 & z^L_{\alpha_1} \leqq t < z^L_{\alpha_2} \\ \cdots & \cdots \\ \alpha & z^L_{\alpha_n} \leqq t < z^R_{\alpha_n} \\ \dfrac{\alpha_n - \alpha_{n-1}}{z^R_{\alpha_{n-1}} - z^R_{\alpha_n}}\left(-t + z^R_{\alpha_{n-1}}\right) + \alpha_{n-1} & z^R_{\alpha_n} \leqq t < z^R_{\alpha_{n-1}} \\ \cdots & \cdots \\ \dfrac{\alpha_0 - \alpha_1}{z^R_{\alpha_1} - z^R_{\alpha_0}}\left(-t + z^R_{\alpha_0}\right) + \alpha_0 & z^R_{\alpha_1} \leqq t \leqq z^R_{\alpha_0} \\ 0 & z^R_{\alpha_0} < t \end{cases}, \qquad (6.3.10)$$

where $\tilde{z}$ denotes $\tilde{c}_{sj}$, $\tilde{a}_{ij}$ or $\tilde{b}_i$ respectively, then the space of feasible solutions X is defined by the set of $x \in X$ with x_i, for $i = 1,2,\ldots,n$, satisfying

$$\begin{cases} \sum_{j=1}^{n} a^L_{ij\,\alpha_0} x_j \leqq b^L_{i\,\alpha_0} \\ \sum_{j=1}^{n} a^R_{ij\,\alpha_0} x_j \leqq b^R_{i\,\alpha_0} \\ \sum_{j=1}^{n} a^L_{ij\,\alpha_1} x_j \leqq b^L_{i\,\alpha_1} \\ \sum_{j=1}^{n} a^R_{ij\,\alpha_1} x_j \leqq b^R_{i\,\alpha_1} \\ \qquad \vdots \\ \sum_{j=1}^{n} a^L_{ij\,\alpha_n} x_j \leqq b^L_{i\,\alpha_n} \\ \sum_{j=1}^{n} a^R_{ij\,\alpha_n} x_j \leqq b^R_{i\,\alpha_n} \\ \qquad x_i \geqq 0 \end{cases}. \qquad (6.3.11)$$

The result of this corollary, a solution transformation approach, will be used in Chapter 7.

Theorem 6.3.2 Let each of the fuzzy parameters be a piecewise trapezoidal membership function in FMOLP$_\alpha$ (6.2.3):

$$\mu_{\tilde{z}}(t)=\begin{cases}0 & t<z^L_{\alpha_0}\\ \dfrac{\alpha_1-\alpha_0}{z^L_{\alpha_1}-z^L_{\alpha_0}}\left(t-z^L_{\alpha_0}\right)+\alpha_0 & z^L_{\alpha_0}\leqq t<z^L_{\alpha_1}\\ \dfrac{\alpha_2-\alpha_1}{z^L_{\alpha_2}-z^L_{\alpha_1}}\left(t-z^L_{\alpha_1}\right)+\alpha_1 & z^L_{\alpha_1}\leqq t<z^L_{\alpha_2}\\ \cdots & \cdots\\ 1 & z^L_{\alpha_n}\leqq t<z^R_{\alpha_n}\\ \dfrac{\alpha_n-\alpha_{n-1}}{z^R_{\alpha_{n-1}}-z^R_{\alpha_n}}\left(-t+z^R_{\alpha_{n-1}}\right)+\alpha_{n-1} & z^R_{\alpha_n}\leqq t<z^R_{\alpha_{n-1}}\\ \cdots & \cdots\\ \dfrac{\alpha_0-\alpha_1}{z^R_{\alpha_1}-z^R_{\alpha_0}}\left(-t+z^R_{\alpha_0}\right)+\alpha_0 & z^R_{\alpha_1}\leqq t\leqq z^R_{\alpha_0}\\ 0 & z^R_{\alpha_0}<t\end{cases} \quad (6.3.12)$$

If a point $x^* \in R^n$ is a feasible solution to the FMOLP$_\alpha$ problem, then x^* is a complete optimal solution to the FMOLP$_\alpha$ problem if and only if x^* is a complete optimal solution to the MOLP$_\alpha$ problem:

$$
(\mathrm{MOLP}_\alpha)\quad
\begin{cases}
\max & \left\langle c^L_{\alpha_0}, x\right\rangle \\
& \left\langle c^R_{\alpha_0}, x\right\rangle \\
& \left\langle c^L_{\alpha_1}, x\right\rangle \\
& \vdots \\
& \left\langle c^R_{\alpha_{n-1}}, x\right\rangle \\
& \left\langle c^L_{\alpha_n}, x\right\rangle \\
& \left\langle c^R_{\alpha_n}, x\right\rangle \\
\mathrm{s.t} & \sum_{j=1}^{n} a^{\ L}_{ij\alpha_0} x_j \leqq b^L_{i\alpha_0} \\
& \sum_{j=1}^{n} a^{\ R}_{ij\alpha_0} x_j \leqq b^R_{i\alpha_0} \\
& \sum_{j=1}^{n} a^{\ L}_{ij\alpha_1} x_j \leqq b^L_{i\alpha_1} \\
& \vdots \\
& \sum_{j=1}^{n} a^{\ R}_{ij\alpha_{n-1}} x_j \leqq b^R_{i\alpha_{n-1}} \\
& \sum_{j=1}^{n} a^{\ L}_{ij\alpha_n} x_j \leqq b^L_{i\alpha_n} \\
& \sum_{j=1}^{n} a^{\ R}_{ij\alpha_n} x_j \leqq b^R_{i\alpha_n} \\
& x_i \geqq 0
\end{cases}
\tag{6.3.13}
$$

where $\alpha = \alpha_0 \le \alpha_1 \le \cdots \le \alpha_{n-1} \le \alpha_n = 1$.

Proof. If x^* is an optimal solution to the FMOLP$_\alpha$ problem, then for any $x \in \widetilde{X}_\alpha$, we have $\left\langle \tilde{c}, x^*\right\rangle_F \succeq_\alpha \left\langle \tilde{c}, x\right\rangle_F$. Therefore, for any $\lambda \in [\alpha, 1]$,

$$(\sum_{i=1}^{n} \tilde{c}_i x_i^*)^L_\lambda \geqq (\sum_{i=1}^{n} \tilde{c}_i x_i)^L_\lambda \quad \text{and} \quad (\sum_{i=1}^{n} \tilde{c}_i x_i^*)^R_\lambda \geqq (\sum_{i=1}^{n} \tilde{c}_i x_i)^R_\lambda,$$

that is,

$$\sum_{i=1}^{n} c^{\ L}_{i\lambda} x_i^* \geqq \sum_{i=1}^{n} c^{\ L}_{i\lambda} x_i \quad \text{and} \quad \sum_{i=1}^{n} c^{\ R}_{i\lambda} x_i^* \geqq \sum_{i=1}^{n} c^{\ R}_{i\lambda} x_i.$$

Hence x^* is a *complete optimal solution* to the MOLP$_\alpha$ problem by Definition 6.2.10.

If x^* is a complete optimal solution to the MOLP$_\alpha$ problem, then for all $x \in X_\alpha$, we have

$$(\langle c^L_{\alpha_i}, x^* \rangle, \langle c^R_{\alpha_i}, x^* \rangle)^T \geqq (\langle c^L_{\alpha_i}, x \rangle, \langle c^R_{\alpha_i}, x \rangle)^T, \quad i = 0, 1, \cdots, n$$

that is,

$$\sum_{j=1}^{n} c^L_{j\alpha_i} x^*_j \geqq \sum_{j=1}^{n} c^L_{j\alpha_i} x_j, \ \sum_{j=1}^{n} c^R_{j\alpha_i} x^*_j \geqq \sum_{j=1}^{n} c^R_{j\alpha_i} x_j, \ i = 1, 2, \cdots, n.$$

For any $\lambda \in [\alpha, 1]$, there exists an $i \in \{1, 2, \cdots, n\}$ so that $\lambda \in [\alpha_{i-1}, \alpha_i]$. As $\tilde{c}$ has piecewise trapezoidal membership functions, we have

$$c^L_\lambda = \frac{\lambda - \alpha_{i-1}}{\alpha_i - \alpha_{i-1}} \left(c^L_{\alpha_1} - c^L_{\alpha_{i-1}}\right) + c^L_{\alpha_{i-1}}$$

and

$$c^R_\lambda = \frac{\lambda - \alpha_{i-1}}{\alpha_i - \alpha_{i-1}} \left(c^R_{\alpha_i} - c^R_{\alpha_{i-1}}\right) + c^R_{\alpha_{i-1}}.$$

From Lemma 6.2.1, we have

$$\sum_{i=1}^{n} c^L_{i\lambda} x^*_i \geqq \sum_{i=1}^{n} c^L_{i\lambda} x_i$$

and

$$\sum_{i=1}^{n} c^R_{i\lambda} x^*_i \geqq \sum_{i=1}^{n} c^R_{i\lambda} x_i,$$

for any $\lambda \in [\alpha, 1]$. Therefore x^* is an optimal solution to the FMOLP$_\alpha$ problem.

Please note that the solution discussed in this theorem is a *completed optimal solution*. The following Theorem 6.3.3 concerns the *Pareto optimal solutions* whereas Theorem 6.3.4 is about the *weak Pareto optimal solutions* used in the transformation.

Theorem 6.3.3 Let each of the fuzzy parameters be a piecewise trapezoidal membership function in FMOLP$_\alpha$ (6.3.12)**.** Let a point $x^* \in \tilde{X}_\alpha$ be any feasible solution to the FMOLP$_\alpha$ problem. Then x^* is a *Pareto optimal solution* to the FMOLP$_\alpha$ problem if and only if x^* is a *Pareto optimal solution* to the MOLP$_\alpha$ problem in (6.3.13).

Proof. Let $x^* \in \tilde{X}_\alpha$ be a Pareto optimal solution to the FMOLP$_\alpha$ problem. On the contrary, we suppose that there exists an $\bar{x} \in X$ such that

$$(\langle c^L_{\alpha_i}, x^*\rangle, \ \langle c^R_{\alpha_i}, x^*\rangle)^T \le (\langle c^L_{\alpha_i}, \bar{x}\rangle, \langle c^R_{\alpha_i}, \bar{x}\rangle)^T, \quad i = 0, 1, \cdots, n \cdot \tag{6.3.14}$$

Therefore

$$0 \le (\langle c^L_{\alpha_i}, \bar{x}\rangle - \langle c^L_{\alpha_i}, x^*\rangle, \langle c^R_{\alpha_i}, \bar{x}\rangle - \langle c^R_{\alpha_i}, x^*\rangle,)^T, \quad i = 0, 1, 2, \cdots, n. \tag{6.3.15}$$

Hence

$$0 \le \langle c^L_{\alpha_i}, \bar{x}\rangle - \langle c^L_{\alpha_i}, x^*\rangle, 0 \le \langle c^R_{\alpha_i}, \bar{x}\rangle - \langle c^R_{\alpha_i}, x^*\rangle, \quad i = 0, 1, 2, \cdots, n. \tag{6.3.16}$$

That is,

$$\langle c^L_{\alpha_i}, \bar{x}\rangle \ge \langle c^L_{\alpha_i}, x^*\rangle, \langle c^R_{\alpha_i}, \bar{x}\rangle \ge \langle c^R_{\alpha_1}, x^*\rangle, \quad i = 1, 2, \cdots, n.$$

By using Lemma 6.3.1, for any $\lambda \in [\alpha, 1]$, we have

$$\langle c^L_\lambda, x^*\rangle \le \langle c^L_\lambda, \bar{x}\rangle$$

and

$$\langle c^R_\lambda, x^*\rangle \le \langle c^R_\lambda, \bar{x}\rangle,$$

that is, $\langle \tilde{c}, \bar{x}\rangle_F \succeq \langle \tilde{c}, x^*\rangle_F$. However, this contradicts the assumption that $x^* \in \tilde{X}_\alpha$ is a Pareto optimal solution to the FMOLP problem.

Let $x \in X_\alpha$ be a Pareto optimal solution to the MOLP$_\alpha$ problem. If x^* is not a Pareto optimal solution to the problem, then there exists an $\bar{x} \in \tilde{X}_\alpha$ such that $\langle \tilde{c}, \bar{x}\rangle_F \succeq \langle \tilde{c}, x^*\rangle_F$. Therefore, for any $\lambda \in [\alpha, 1]$, we have

$$(\sum_{i=1}^n \tilde{c}_i x_i^*)^L_\lambda \le (\sum_{i=1}^n \tilde{c}_i \bar{x}_i)^L_\lambda$$

and

$$(\sum_{i=1}^n \tilde{c}_i x_i^*)^R_\lambda \le (\sum_{i=1}^n \tilde{c}_i \bar{x}_i)^R_\lambda,$$

that is,

$$\langle c^L_\lambda, x^*\rangle \le \langle c^L_\lambda, \bar{x}\rangle$$

and

$$\langle c^R_\lambda, x^*\rangle \le \langle c^R_\lambda, \bar{x}\rangle.$$

Hence, for $\alpha = \alpha_0 \le \alpha_1 \le \cdots \le \alpha_{n-1} \le \alpha_n = 1$, we have

$$(\langle c^L_{\alpha_i}, x^*\rangle, \ \langle c^R_{\alpha_i}, x^*\rangle)^T \geqq (\langle c^L_{\alpha_i}, x\rangle, \langle c^R_{\alpha_i}, x\rangle)^T, \quad i = 0, 1, \cdots, n,$$

which contradicts the assumption that $x^* \in X_\alpha$ is a Pareto optimal solution to the MOLP$_\alpha$ problem.

Theorem 6.3.4 Let each of the fuzzy parameters be a piecewise trapezoidal membership function in the FMOLP$_\alpha$ problem as shown in (6.3.12), and a point $x^* \in X$ be a feasible solution to the FMOLP$_\alpha$ problem. Then x^* is a *weak Pareto optimal solution* to the FMOLP$_\alpha$ problem if and only if x^* is a *weak Pareto optimal solution* to the MOLP$_\alpha$ problem as shown in (6.3.13).

Proof. Similar to Theorem 6.3.3.

Therefore, if we use existing methods to get a complete optimal solution x^* to the MOLP$_\alpha$ problem, then x^* is a complete optimal solution to the FMOLP$_\alpha$ problem. This gives a way to solve FMOLP$_\alpha$ problems, which will be used in developing detailed algorithms in Chapter 7.

6.3.2 *Weighted MOLP transformation*

From Theorems 6.3.2 to 6.3.4, to find all complete optimal, Pareto optimal or all weak Pareto optimal solutions to the FMOLP problem, we need to find all complete or Pareto or weak Pareto optimal solutions to the MOLP problem. Now, associated with the MOLP problem, we consider the following weighting linear programming problem (Sakawa, 1993):

$$
(\mathrm{MOLP}_w)\quad \begin{cases} \max\ \langle w, \tilde{c}, x\rangle = \sum_{i=0}^{n}\left(w_i^L\langle c_{\alpha_i}^L, x\rangle + w_i^R\langle c_{\alpha_i}^R, x\rangle\right) \\ \text{s.t}\quad \sum_{j=1}^{n} a_{ij\,\alpha_0}^{\ L} x_j \leqq b_{i\alpha_0}^{L} \\ \qquad \sum_{j=1}^{n} a_{ij\,\alpha_0}^{\ R} x_j \leqq b_{i\alpha_0}^{R} \\ \qquad \sum_{j=1}^{n} a_{ij\,\alpha_1}^{\ L} x_j \leqq b_{i\alpha_1}^{L} \\ \qquad \vdots \\ \qquad \sum_{j=1}^{n} a_{ij\,\alpha_{n-1}}^{\ R} x_j \leqq b_{i\alpha_{n-1}}^{R} \\ \qquad \sum_{j=1}^{n} a_{ij\,\alpha_n}^{\ L} x_j \leqq b_{i\alpha_n}^{L} \\ \qquad \sum_{j=1}^{n} a_{ij\,\alpha_n}^{\ R} x_j \leqq b_{i\alpha_n}^{R} \\ \qquad x_i \geqq 0 \end{cases} \tag{6.3.17}
$$

where $w = (w_0^L, w_0^R, w_1^L, w_1^R, \cdots, w_n^L, w_n^R) \geq 0$, $\alpha = \alpha_0 \leq \alpha_1 \leq \cdots \leq \alpha_{n-1} \leq \alpha_n = 1$.

Theorem 6.3.5 Let each of the fuzzy parameters be a piecewise trapezoidal membership function in the FMOLP problem, as shown in (6.3.12) and a point $x^* \in \widetilde{X}$ be a feasible solution to the FMOLP problem. If it is an optimal solution of MOLP_w (6.3.17) for some w > 0, then it is a *Pareto optimal solution* to the FMOLP problem.

Proof. If an optimal solution x^* to the MOLP_w problem is not a Pareto optimal solution to the FMOLP problem, from Theorem 6.2.3, it is not a Pareto optimal solution to the MOLP problem, thus there exists an $\bar{x} \in X$ such that

$$
(\langle c_{\alpha_i}^L, x^*\rangle, \langle c_{\alpha_i}^R, x^*\rangle)^T \leq (\langle c_{\alpha_i}^L, \bar{x}\rangle, \langle c_{\alpha_i}^R, \bar{x}\rangle)^T, \quad i = 0, 1, \cdots, n. \tag{6.3.18}
$$

Hence, there exists at least a c_i^L or c_i^R, i = 0, 1, 2, … n such that '<' holds. Noting that $w = (w_0^L, w_0^R, w_1^L, w_1^R, \cdots, w_n^L, w_n^R) > 0$, this implies

$$\begin{aligned}\left\langle w, \tilde{c}, x^*\right\rangle &= \sum_{i=0}^{n}\left(w_i^L\left\langle c_{\alpha_i}^L, x^*\right\rangle + w_i^R\left\langle c_{\alpha_i}^R, x^*\right\rangle\right)\\ &< \sum_{i=0}^{n}\left(w_i^L\left\langle c_{\alpha_i}^L, \bar{x}\right\rangle + w_i^R\left\langle c_{\alpha_i}^R, \bar{x}\right\rangle\right)\\ &= \left\langle w, \tilde{c}, \bar{x}\right\rangle.\end{aligned}$$

However, this contradicts the assumption that x^* is an *optimal solution* to the MOLP$_w$ problem for some $w > 0$.

Theorem 6.3.6 Let each of the fuzzy parameters be a piecewise trapezoidal membership function in the FMOLP problem, as shown in (6.3.12) and a point $x^* \in \tilde{X}$ be any feasible solution to the FMOLP problem. If it is a Pareto optimal solution to the problem, then it is an optimal solution to MOLP$_w$ (6.3.17) for some $w > 0$.

Proof. If x^* is a Pareto optimal solution to the FMOLP problem, then it is a Pareto optimal solution to the MOLP problem from Theorem 6.3.3. By using Theorem 3.2 of Chapters 3, it is an optimal solution to the MOLP$_w$ problem for some $w > 0$.

Theorem 6.3.7 Let each of the fuzzy parameters be a piecewise trapezoidal membership function in the FMOLP problem, as shown in (6.3.12) and a point $x^* \in \tilde{X}$ be a feasible solution to the FMOLP problem. Then it is an optimal solution of MOLP$_w$ (6.3.17) for some $w > 0$ if and only if it is a weak Pareto optimal solution to the FMOLP problem.

Proof. Similar to Theorems 6.3.5 and 6.3.6.

Therefore, if we use existing methods to get a complete optimal solution x^* to the MOLP$_w$ problem, then x^* is a complete optimal solution to the FMOLP problem. This gives a way to solve the FMOLP problem.

6.3.3 *Constrained MOLP transformation*

Associated with the MOLP problem and the constrained linear programming (CLP) problem (Sakawa, 1993), we now consider the following constrained MOLP (CMOLP) problem:

$$
\text{(CMOLP)} \quad \begin{cases} \max \ \langle c_i, x \rangle \\ \text{s.t.} \ \langle c_j, x \rangle \geqq \varepsilon_j, \ j = 1, 2, \cdots, 2(n+1)); \ j \neq i \\ \quad \sum_{j=1}^{n} a_{ij}{}_{\alpha_0}^{L} x_j \leqq b_{i}{}_{\alpha_0}^{L} \\ \quad \sum_{j=1}^{n} a_{ij}{}_{\alpha_0}^{R} x_j \leqq b_{i}{}_{\alpha_0}^{R} \\ \quad \sum_{j=1}^{n} a_{ij}{}_{\alpha_1}^{L} x_j \leqq b_{i}{}_{\alpha_1}^{L} \\ \qquad \vdots \\ \quad \sum_{j=1}^{n} a_{ij}{}_{\alpha_{n-1}}^{R} x_j \leqq b_{i}{}_{\alpha_{n-1}}^{R} \\ \quad \sum_{j=1}^{n} a_{ij}{}_{\alpha_n}^{L} x_j \leqq b_{i}{}_{\alpha_n}^{L} \\ \quad \sum_{j=1}^{n} a_{ij}{}_{\alpha_n}^{R} x_j \leqq b_{i}{}_{\alpha_n}^{R} \\ \quad x_i \geqq 0 \end{cases} \tag{6.3.19}
$$

where $c_i = (c_{1\alpha_i}^{L}, c_{2\alpha_i}^{L}, \cdots, c_{n\alpha_i}^{L})^T$, $c_{n+1+i} = (c_{1\alpha_i}^{R}, c_{1\alpha_i}^{R}, \cdots, c_{1\alpha_i}^{R})^T \in R^n$, $i = 1, 2, \cdots, n, n+1, \cdots, 2(n+1)$, and ε_j is the minimum acceptable values for objectives corresponding to $j \neq i$.

Theorem 6.3.8 Let each of the fuzzy parameters be a piecewise trapezoidal membership function in the FMOLP problem as shown in (6.3.12) and a point $x^* \in \tilde{X}$ be any feasible solution to the FMOLP problem. If it is a unique optimal solution of CMOLP (6.3.19) for some ε_j, $j = 1, 2, \ldots, 2(n+1)$ and $j \neq i$, then it is a *Pareto optimal solution* to the FMOLP problem.

Proof. If a unique optimal solution x^* to the CMOLP problem is not *a* Pareto optimal solution to the FMOLP problem. Then it is not a Pareto optimal solution to the MOLP problem from Theorem 6.3.3, therefore there exists an $\bar{x} \in X$ such that

$$\left\langle c_{\alpha_i}^L, \bar{x} \right\rangle \geq \left\langle c_{\alpha_i}^L, x^* \right\rangle, \left\langle c_{\alpha_i}^R, \bar{x} \right\rangle \geq \left\langle c_{\alpha_1}^R, x^* \right\rangle, \quad i = 1, 2, \cdots, n.$$

This means

$$\varepsilon_{\mathrm{j}} \leq \left\langle c_j, x^* \right\rangle \leqq \left\langle c_j, \bar{x} \right\rangle, \ j = 1, 2, \cdots, 2(n+1); \ j \neq i, \ \ \left\langle c_i, x^* \right\rangle \leqq \left\langle c_i, \bar{x} \right\rangle,$$

which contradicts the assumption that x^* is a unique optimal solution of the CMOLP problem for some ε_j, j = 1, 2,…, 2(n+1) and $j \neq i$.

Theorem 6.3.9 Let each of the fuzzy parameters be a piecewise trapezoidal membership function in the FMOLP problem as shown in (6.3.12) and a point $x^* \in \tilde{X}$ be any feasible solution to the FMOLP problem. If it is a Pareto optimal solution to the problem, then it is an optimal solution of CMOLP for some ε_j, j = 1, 2,…, 2(n+1) and $j \neq i$.

Proof. If x^* is a Pareto optimal solution to the FMOLP problem, then it is a Pareto optimal solution to the MOLP problem from Theorem 6.3.3. Suppose x^* is not an optimal solution of the CMOLP problem for some ε_j, j = 1, 2, …, 2(n+1); $j \neq i$, then there exists an $\bar{x} \in X$ such that

$$\left\langle c_j, x^* \right\rangle = \varepsilon_{\mathrm{j}} \leqq \left\langle c_j, \bar{x} \right\rangle, \ j = 1, 2, \cdots, 2(n+1); \ j \neq i, \ \ \left\langle c_i, x^* \right\rangle < \left\langle c_i, \bar{x} \right\rangle,$$

which contradicts the fact that x^* is a Pareto optimal solution to the MOLP problem.

Theorem 6.3.10 Let each of the fuzzy parameters be a piecewise trapezoidal membership function in the FMOLP problem as shown in (6.3.12) and a point $x^* \in \tilde{X}$ be any feasible solution to the FMOLP problem. If it is an optimal solution of CMOLP for some ε_j, j = 1, 2,…, 2(n+1) and $j \neq i$, then it is a weak Pareto optimal solution to the FMOLP problem.

Proof. If an optimal solution x^* to the CMOLP problem is not a weak Pareto optimal solution to the FMOLP problem. Then it is not a weak Pareto optimal solution to the MOLP problem from Theorem 6.3.4, therefore, there exists an $\bar{x} \in X$ such that

$$\left\langle c_{\alpha_i}^L, \bar{x}\right\rangle > \left\langle c_{\alpha_i}^L, x^*\right\rangle, \left\langle c_{\alpha_i}^R, \bar{x}\right\rangle > \left\langle c_{\alpha_1}^R, x^*\right\rangle, \quad i = 1, 2, \cdots, n.$$

This means

$$\varepsilon_j \leqq \left\langle c_j, x^*\right\rangle < \left\langle c_j, \bar{x}\right\rangle, \ j = 1, 2, \cdots, 2(n+1);\ j \neq i, \quad \left\langle c_i, x^*\right\rangle < \left\langle c_i, \bar{x}\right\rangle,$$

which contradicts the assumption that x^* is an optimal solution of the CMOLP problem for some ε_j, $j = 1, 2, \ldots, 2(n+1)$; $j \neq i$.

Therefore, if we use existing methods to get a complete optimal solution x^* to the CMOLP problem, then x^* is a complete optimal solution to the FMOLP problem. This gives another way to solve FMOLP problems.

6.3.4 *Weighted maximum MOLP transformation*

Associated with the MOLP problem, we consider the following weighted maximum linear programming (WMLP) problem (Sakawa, 1993):

$$(\mathrm{MOLP}_{wm}) \quad \begin{cases} \max \min\limits_{i=1,2,\cdots,2(n+1)} w_i \left\langle c_i, x\right\rangle \\ \text{s.t. } \left\langle c_j, x\right\rangle \geqq \varepsilon_j, \ j = 1, 2, \cdots, 2(n+1));\ j \neq i \\ \qquad \sum\limits_{j=1}^{n} a_{ij}{}_{\alpha_0}^{L} x_j \leqq b_{i\alpha_0}^{L} \\ \qquad \sum\limits_{j=1}^{n} a_{ij}{}_{\alpha_0}^{R} x_j \leqq b_{i\alpha_0}^{R} \\ \qquad \sum\limits_{j=1}^{n} a_{ij}{}_{\alpha_1}^{L} x_j \leqq b_{i\alpha_1}^{L} \\ \qquad \vdots \\ \qquad \sum\limits_{j=1}^{n} a_{ij}{}_{\alpha_{n-1}}^{R} x_j \leqq b_{i\alpha_{n-1}}^{R} \\ \qquad \sum\limits_{j=1}^{n} a_{ij}{}_{\alpha_n}^{L} x_j \leqq b_{i\alpha_n}^{L} \\ \qquad \sum\limits_{j=1}^{n} a_{ij}{}_{\alpha_n}^{R} x_j \leqq b_{i\alpha_n}^{R} \\ \qquad x_i \geqq 0 \end{cases} \tag{6.3.20}$$

where $c_i = (c_{1\alpha_i}^L, c_{2\alpha_i}^L, \cdots, c_{n\alpha_i}^L)^T$, $c_{n+1+i} = (c_{1\alpha_i}^R, c_{1\alpha_i}^R, \cdots, c_{1\alpha_i}^R)^T \in R^n$, $i = 1, 2, \cdots, n, n+1, \cdots, 2(n+1)$, $w = (w_1, w_2, \cdots, w_{2n+1}, w_{2(n+1)}) \geq 0$ and $\alpha = \alpha_0 \leq \alpha_1 \leq \cdots \leq \alpha_{n-1} \leq \alpha_n = 1$.

Theorem 6.3.11 Let each of the fuzzy parameters be a piecewise trapezoidal membership function in the FMOLP problem as shown in (6.3.12) and a point $x^* \in \tilde{X}$ be a feasible solution to the FMOLP problem. If it is a unique optimal solution of MOLP$_{wm}$ (6.3.20) for some $w \geqq 0$, then it is a *Pareto optimal solution* to the FMOLP problem.

Proof. If a unique optimal solution x^* to the MOLP$_{wm}$ problem for some $w \geqq 0$ is not a Pareto optimal solution to the FMOLP problem. Then it is not a Pareto optimal solution to the MOLP problem from Theorem 6.3.3, therefore there exists an $\bar{x} \in X$ such that

$$\left\langle c_{\alpha_i}^L, \bar{x} \right\rangle \geq \left\langle c_{\alpha_i}^L, x^* \right\rangle, \left\langle c_{\alpha_i}^R, \bar{x} \right\rangle \geq \left\langle c_{\alpha_1}^R, x^* \right\rangle, \quad i = 1, 2, \cdots, n.$$

In view of $w = (w_1, w_2, \cdots, w_{2n+1}, w_{2(n+1)}) \geqq 0$, it follows

$$w_j \left\langle c_j, x^* \right\rangle \leqq w_j \left\langle c_j, \bar{x} \right\rangle, \; j = 1, 2, \cdots, 2(n+1).$$

Hence,

$$\min_{i=1,2,\cdots,2(n+1)} w_i \left\langle c_i, x^* \right\rangle \leqq \min_{i=1,2,\cdots,2(n+1)} w_i \left\langle c_i, \bar{x} \right\rangle,$$

which contradicts the assumption that x^* is a unique optimal solution of the MOLP$_{wm}$ problem for some $w = (w_1, w_2, \cdots, w_{2n+1}, w_{2(n+1)}) \geqq 0$.

Theorem 6.3.12 Let each of the fuzzy parameters be a piecewise trapezoidal membership function in the FMOLP problem as shown in (6.3.12) and a point $x^* \in \tilde{X}$ be any feasible solution to the FMOLP problem. If it is a Pareto optimal solution to the FMOLP problem, then it is an *optimal solution* of MOLP$_{wm}$ for some $w = (w_1, w_2, \cdots, w_{2n+1}, w_{2(n+1)}) > 0$.

Proof. If x^* is a Pareto optimal solution to the FMOLP problem then it is a Pareto optimal solution to the MOLP problem from Theorem 6.3.3. Here, without loss of generality, we assume that $\left\langle c_j, x \right\rangle > 0, \; j = 1, 2, \cdots, 2(n+1)$ for all $x \in X$ and choose

$w^* = (w_1^*, w_2^*, \cdots, w_{2(n+1)}^*) > 0$ such that $w_j^* \langle c_j, x^* \rangle = v,\ j = 1, 2, \cdots, 2(n+1)$. Now, we assume that x^* is not an optimal solution of the MOLP$_{wm}$ problem for $w^* = (w_1^*, w_2^*, \cdots, w_{2(n+1)}^*) > 0$, then there exists an $\bar{x} \in X$ such that

$$w_j^* \langle c_j, x^* \rangle < w_j^* \langle c_j, \bar{x} \rangle,\ j = 1, 2, \cdots, 2(n+1).$$

Noting $w^* = (w_1^*, w_2^*, \cdots, w_{2(n+1)}^*) > 0$, this implies

$$\langle c_j, x^* \rangle < \langle c_j, \bar{x} \rangle,\ j = 1, 2, \cdots, 2(n+1)$$

which contradicts the fact that x^* is a Pareto optimal solution to the MOLP problem.

Theorem 6.3.13 Let each of the fuzzy parameters be a piecewise trapezoidal membership function in the FMOLP problem as shown in (6.3.12) and a point $x^* \in \tilde{X}$ be a feasible solution to the FMOLP problem. If it is an *optimal solution* of MOLP$_{wm}$ for some $w \geq 0$, then it is a weak Pareto optimal solution to the FMOLP problem.

Proof. If an optimal solution x^* to the MOLP$_{wm}$ problem for some $w \geqq 0$ is not a weak Pareto optimal solution to the FMOLP problem. Then it is not a weak Pareto optimal solution to the MOLP problem from Theorem 6.3.4. Therefore, there exists an $\bar{x} \in X$ such that

$$\langle c_{\alpha_i}^L, \bar{x} \rangle > \langle c_{\alpha_i}^L, x^* \rangle,\ \langle c_{\alpha_i}^R, \bar{x} \rangle > \langle c_{\alpha_1}^R, x^* \rangle, \quad i = 1, 2, \cdots, n.$$

In view of $w = (w_1, w_2, \cdots, w_{2n+1}, w_{2(n+1)}) \geqq 0$, it follows

$$w_j \langle c_j, x^* \rangle \leq w_j \langle c_j, \bar{x} \rangle,\ j = 1, 2, \cdots, 2(n+1).$$

Hence,

$$\min_{i=1,2,\cdots,2(n+1)} w_i \langle c_i, x^* \rangle < \min_{i=1,2,\cdots,2(n+1)} w_i \langle c_i, \bar{x} \rangle,$$

which contradicts the assumption that x^* is a unique optimal solution of the MOLP$_{wm}$ problem for some $w = (w_1, w_2, \cdots, w_{2n+1}, w_{2(n+1)}) \geqq 0$.

Theorem 6.3.14 Let each of the fuzzy parameters be a piecewise trapezoidal membership function in the FMOLP as shown in (6.3.12) and a point $x^* \in \tilde{X}$ be a feasible solution to the FMOLP problem. If it is a weak Pareto optimal solution to the FMOLP problem, then it is an *optimal solution* of MOLP$_{wm}$ for some $w = (w_1, w_2, \cdots, w_{2n+1}, w_{2(n+1)}) > 0$.

Proof. If x^* is a weak Pareto optimal solution to the FMOLP problem then it is a weak Pareto optimal solution to the MOLP problem from Theorem 6.3.4. Here, without loss of generality, we can assume that $\langle c_j, x\rangle > 0,\ j = 1, 2, \cdots, 2(n+1)$ for all $x \in X$ and choose $w^* = (w_1^*, w_2^*, \cdots, w_{2(n+1)}^*) > 0$ such that $w_j^*\langle c_j, x^*\rangle = v,\ j = 1, 2, \cdots, 2(n+1)$. Now, we assume x^* is not an optimal solution of the MOLP$_{wm}$ problem for $w^* = (w_1^*, w_2^*, \cdots, w_{2(n+1)}^*) > 0$, then there exists an $\bar{x} \in X$ such that

$$w_j^*\langle c_j, x^*\rangle < w_j^*\langle c_j, \bar{x}\rangle,\ j = 1, 2, \cdots, 2(n+1).$$

Noting $w^* = (w_1^*, w_2^*, \cdots, w_{2(n+1)}^*) > 0$, this implies

$$\langle c_j, x^*\rangle < \langle c_j, \bar{x}\rangle,\ j = 1, 2, \cdots, 2(n+1)$$

which contradicts the fact that x^* is a Pareto optimal solution to the MOLP problem.

Therefore, if we use existing methods to get a complete optimal solution x^* to the MOLP$_{wm}$ problem, then x^* is a complete optimal solution to the FMOLP problem. This gives another way to solve FMOLP problems.

6.4 Fuzzy Multi-Objective Linear Goal Programming Models

In order to deal with FMOLP (6.2.1), under some circumstances, decision makers may want to specify fuzzy goals for the objective functions. The key idea behind goal programming is to minimise the deviations from a goal set by decision makers.

Considering the FMOLP$_\alpha$ problem, for the fuzzy objective functions $\langle \tilde{c}, x\rangle_F$, decision makers can specify fuzzy goals $\tilde{g} = (\tilde{g}_1, \tilde{g}_2, \ldots, \tilde{g}_k)^T$ under a satisfactory degree α, which reflects the desired values of the objective functions of decision makers. These fuzzy goals can be represented by fuzzy numbers with any form of membershp functions. By defining a fuzzy deviation function $\tilde{D}(\langle \tilde{c}, x\rangle_F, \tilde{g})$ as a fuzzy difference between a fuzzy objective function $\langle \tilde{c}, x\rangle_F$ and fuzzy goals $\tilde{g} = (\tilde{g}_1, \tilde{g}_2, \ldots, \tilde{g}_k)^T$, the fuzzy multi-objective linear goal programming (FMOLGP$_\alpha$) problem under a satisfactory degree α is formulated as following:

$$(\text{FMOLGP}_{\alpha})\begin{cases}\min_{\alpha}\ \tilde{D}\left(\langle\tilde{c}, x\rangle_F, \tilde{g}\right)\\ \text{subjiect to } \tilde{A}x \underset{=\alpha}{\prec} \tilde{b}\\ \qquad x \geqq 0\end{cases} \tag{6.4.1}$$

Then, the optimal solution of (6.4.1) can be obtained by solving the following non-fuzzy GP models:

$$(\text{GP}_{\alpha\lambda\text{-1}})\begin{cases}\min \max\limits_{\substack{i=1,\ldots,k\\ \lambda\in[\alpha,1]}} \left\{c_{i\lambda}^{L}x - g_{i\lambda}^{L}, c_{i\lambda}^{R}x - g_{i\lambda}^{R}\right\}\\ \text{subject to } A_{\lambda}^{L}x \le b_{\lambda}^{L},\ A_{\lambda}^{R}x \le b_{\lambda}^{R},\ \forall\lambda\in[\alpha,1]\\ \qquad x \geqq 0\end{cases} \tag{6.4.2}$$

or

$$(\text{GP}_{\alpha\lambda\text{-2}})\begin{cases}\min \max\limits_{\substack{i=1,\ldots,k\\ \lambda\in[\alpha,1]}} \left\{g_{i\lambda}^{L} - c_{i\lambda}^{L}x, g_{i\lambda}^{R} - c_{i\lambda}^{R}x\right\}\\ \text{subject to } A_{\lambda}^{L}x \le b_{\lambda}^{L},\ A_{\lambda}^{R}x \le b_{\lambda}^{R},\ \forall\lambda\in[\alpha,1]\\ \qquad x \geqq 0\end{cases} \tag{6.4.3}$$

where

$$\begin{pmatrix}c_{1\lambda}^{L}\\ c_{2\lambda}^{L}\\ \vdots\\ c_{k\lambda}^{L}\end{pmatrix}=\begin{bmatrix}c_{11\lambda}^{L} & c_{12\lambda}^{L} & \cdots & c_{1n\lambda}^{L}\\ c_{21\lambda}^{L} & c_{22\lambda}^{L} & \cdots & c_{2n\lambda}^{L}\\ \vdots & \vdots & \ddots & \vdots\\ c_{k1\lambda}^{L} & c_{k2\lambda}^{L} & \cdots & c_{kn\lambda}^{L}\end{bmatrix},\ \begin{pmatrix}c_{1\lambda}^{R}\\ c_{2\lambda}^{R}\\ \vdots\\ c_{k\lambda}^{R}\end{pmatrix}=\begin{bmatrix}c_{11\lambda}^{R} & c_{12\lambda}^{R} & \cdots & c_{1n\lambda}^{R}\\ c_{21\lambda}^{R} & c_{22\lambda}^{R} & \cdots & c_{2n\lambda}^{R}\\ \vdots & \vdots & \ddots & \vdots\\ c_{k1\lambda}^{R} & c_{k2\lambda}^{R} & \cdots & c_{kn\lambda}^{R}\end{bmatrix},$$

The adoption of $\text{GP}_{\alpha\lambda\text{-1}}$ (6.4.2) or $\text{GP}_{\alpha\lambda\text{-2}}$ (6.4.3) for solving the FMOLGP_{α} problem depends on the relationship of $\langle\tilde{c}, x\rangle_F$ and $\tilde{g}$, *i.e.*, if $\langle\tilde{c}, x\rangle_F \succeq \tilde{g}$ then $\text{GP}_{\alpha\lambda\text{-1}}$ (6.4.2) is used, otherwise, $\text{GP}_{\alpha\lambda\text{-2}}$ (6.4.3) is adopted.

Therefore, if we use existing methods to get a satisfactory solution x^* to the GP problem, then x^* is a satisfactory solution to the FMOLP problem. This gives another way to solve the FMOLP problems by providing goals of objectives, which will be used in developing detailed algorithms in Chapter 7.

6.5 Summary

The FMODM models extend MODM decision functions from crisp to imprecise scope. Two essential issues are summarised here to help readers better understand and use these proposed models.

(1) In the proposed FMOLP model, fuzzy parameters may appear in both objective functions and constraints. When only objective functions or only constraints include fuzzy parameters, the model is still applicable to deal with non-fuzzy parameters, as a real number is as a special case of a fuzzy number. Similarly, in the proposed FMOLGP, a goal with a real number is also as a special case of a fuzzy goal.

(2) Both FMOLP and FMOLGP models allow decision makers to use any form of membership functions for describing fuzzy parameters in objective functions and constraints, and fuzzy goals.

This page intentionally left blank

Chapter 7

Fuzzy MODM Methods

As described in Chapter 6, FMOLP is the most popular form of fuzzy multi-objective decision-making (FMODM) problems. To derive an optimal solution for an FMOLP problem, we will present three FMOLP methods in this chapter. The first one is a scalarisation-based FMOLP method. The second one is called fuzzy multi-objective linear goal programming (FMOLGP) method, which integrates fuzzy sets with goal programming to extend multi-objective decision analysis. Finally, we present an interactive FMOLP (IFMOLP) method, which has both interactive and goal features. We will implement the three methods in a fuzzy multi-objective decision support system in Chapter 8.

7.1 Related Issues

There are three issues involved in the development of an FMODM method. The first issue is about how to express fuzzy parameters of objective functions and constraints and fuzzy goals by membership functions. The second one is about the presentation form of a Pareto optimal solution for the FMODM problem. And, the third one is about the different processes of solving the FMODM problem.

For the first issue, as discussed in Chapter 5, fuzzy values of parameters are often generated by some experiments and therefore have different figures of data distributions. Some of them may be suitable to be described in a triangular form of membership functions, and some may be more suitable to be expressed in other forms such as a trapezoidal one. In order to deal with a wide range of expressions for fuzzy parameters, the methods introduced in this chapter will allow us to use any form of membership functions for describing parameters shown in both objective functions and constraints. Similarly, for fuzzy goals given by decision makers, the methods will allow us to use any form of membership functions as well.

The second issue involves the expression of a solution and its corresponded objective values for the FMODM problem. If an FMODM method is to provide us with useful assistance, its output, an optimal solution with optimal objective values, must be of sufficient quality and in a suitable form for the decision we are about to make. As discussed in Chapter 6, we suppose an FMODM model of a production planning. An optimal solution of the problem is the output of all products of this factory. Its corresponded objective values are the profit, quality, and worker satisfactory degree. Some FMODM methods have the objective functions, in corresponding to an optimal solution, described in crisp values. And some have optimal objective values described in fuzzy values. As the late case may be more appropriate in decision practice, the methods introduced in this chapter adopt the late approach.

The third issue is about the process of finding an optimal solution. This issue involves understanding the preferences of some decision makers involved the solution process of an MODM problem. It has been found that there are obvious different requirements from decision makers for the process of finding an optimal solution for an FMODM problem. Some decision makers expect to have a method that can fast generate an optimal solution for a given FMODM problem without any extra data providing from them. While some decision makers have had goals for their decision objectives in their FMODM problem and, therefore, prefer a method that can find an optimal solution, which can maximise to meet these goals. Furthermore, with the support of software, some decision makers desire to have a chance to explore more alternative solutions in an interactive fashion with the aim of finding a satisfying solution. For example, they desire to be allowed to continuing revise their goals or change the weights of objective functions, so that to get new optimal solutions. Obviously, the goals or weights given by decision makers may be affected with uncertainty due to the imprecise nature of data evaluated by these decision makers. When a value of a goal is described by a fuzzy number, it is called a *fuzzy goal*. In such a case, an FMODM method will deal with two kinds of fuzziness: a set of fuzzy parameters in the FMODM model and a set of fuzzy goals given by decision makers in a solution process.

This analysis justifies the main reason to develop several different kinds of FMODM methods and some specific features of them.

This chapter presents three FMODM methods: FMOLP (scalarisation-based), FMOLGP (goal-based), and IFMOLP (interactive-based).

There are four common features on the three methods. (1) The parameters in both fuzzy objective functions and constraints and fuzzy goals are described by any form of membership functions. (2) The values of objective functions, corresponding to an optimal solution, are described by that of membership functions as well. (3) The weights of objectives are flexible, given by decision makers. (4) An approximation approach is used in all the three methods. However, the FMOLP method is a non-interactive method, which can directly generate an optimal solution and therefore is suitable for decision makers who do not have deep knowledge about decision model and software monitory. The FMOLGP method and the IFMOLP method all allow decision makers to provide fuzzy goals in any form of membership functions. The IFMOLP method, in particular, has a strong feature of interaction with decision makers by allowing them to revise their fuzzy goals and satisfactory degrees for a solution. This method requires decision makers have enough knowledge on their decision problems and desires for interaction with a decision support system.

The three methods have been implemented in a fuzzy multi-objective decision support system (FMODSS). All examples illustrated in this chapter have been run by the FMODSS and some results are shown in figures captured from this system. More details about the FMODSS will be described in Chapter 8.

7.2 Fuzzy MOLP

In this section, we first describe the FMOLP method and then give a numerical example to illustrate it.

7.2.1 *Method description*

Refer to the description of an MOLP$_{\alpha\lambda}$ problem in Chapter 6, an FMOLP$_\alpha$ problem can be transformed into an MOLP$_{\alpha\lambda}$ problem, which is a crisp programming. And, the solution of the MOLP$_{\alpha\lambda}$ problem is equally the one of the corresponded FMOLP$_\alpha$ problem. As the MOLP$_{\alpha\lambda}$ problem has an infinite number of objective functions and an infinite number of constraints, an approximation approach will be appropriate.

As given in Chapter 6, we have the definition

$$X_\lambda = \{x \in R^n \mid A_\lambda^L x \le b_\lambda^L, A_\lambda^R x \le b_\lambda^R, x \ge 0\}, \ \lambda \in [\alpha, 1]. \qquad (7.2.1)$$

Based on (7.2.1), we propose a fuzzy scalarisation-based algorithm for solving the MOLP$_{\alpha\lambda}$ problem as follows, and therefore solve FMOLP problems. The FMOLP method is described as follows.

Step 1: Specify a satisfactory degree α $(0 \le \alpha \le 1)$ by decision makers.

Step 2: Give weights $w_1, w_2, \cdots, w_k$ for fuzzy objective functions $\tilde{f}_1, \tilde{f}_2, \cdots, \tilde{f}_k$, respectively, and $\sum_{i=1}^k w_i = 1$.

Step 3: Let the interval $[\alpha, 1]$ be decomposed into l mean sub-intervals with $(l+1)$ nodes $\lambda_i (i = \alpha, \cdots, l)$, where $\alpha = \lambda_0 < \lambda_1 < \cdots < \lambda_l = 1$. Based on the decomposition, we define the constraint $X^l = \bigcap_i^l X_{\lambda_i}$, and denote:

$$(\text{MOLP}_{\alpha\lambda})_l \begin{cases} \max \begin{pmatrix} c_{i\lambda_j}^L x \\ c_{i\lambda_j}^R x \end{pmatrix}, i = 1,2,\cdots,k, \ j = 1,2,\cdots,l. \\ \text{s.t.} \quad x \in X^l \end{cases} \qquad (7.2.2)$$

Step 4: Set $l = 1$, then solve (MOLP$_{\alpha\lambda}$)$_l$ with $(x)_l = (x_1, x_2, \ldots, x_n)_l$, and the solution obtained is subject to the constraint $x \in X^l$.

Step 5: Solve $(\mathrm{MOLP}_{\alpha\lambda})_{2l}$ with $(x)_{2l}$, and the solution obtained is subject to the constraint $x \in X^{2l}$.

Step 6: If $\|(x)_{2l} - (x)_l\| < \varepsilon$, then the solution x^* of the $\mathrm{MOLP}_{\alpha\lambda}$ problem is $(x)_{2l}$. Otherwise, update l to $2l$ and go back to Step 5.

Now we give some explanations about this method.

- In principle, it needs to give a tolerance $\varepsilon > 0$ or a value for the times of decomposition loop as a termination condition. As it may be hard for decision makers to give such a value, the DSS has set related values in its programming to control this process. This issue is also applicable for the methods in Sections 7.2 and 7.3.
- When decision makers do not provide any weights for objective functions, this method assumes all the weights of objectives are equal, that is, $w_1 = w_2 = \cdots = w_k$ for fuzzy objective functions, respectively, and $\sum_{i=1}^{k} w_i = 1$. This issue is also applicable for the methods in Sections 7.2 and 7.3.
- In Step 4, the interval $[\alpha, 1]$ is not split initially, and only $\lambda_0 = \alpha$ and $\lambda_1 = 1$ are considered. Hence, each fuzzy objective function $\tilde{f}_i(x) = \tilde{c}_i x$ in the FMOLP_α is converted into four non-fuzzy objective functions:

$$\begin{pmatrix} c_{i\alpha}^L x \\ c_{i1}^L x \\ c_{i1}^R x \\ c_{i\alpha}^R x \end{pmatrix}, \ i = 1, \dots, k. \tag{7.2.3}$$

  Similarly, each fuzzy constraint $\tilde{a}_s x \precsim_\alpha \tilde{b}_s$ in the FMOLP_α is converted into four non-fuzzy constraints, which are as follows:

$$\begin{pmatrix} a_{s\alpha}^L x \\ a_{s1}^L x \\ a_{s1}^R x \\ a_{s\alpha}^R x \end{pmatrix} \le \begin{pmatrix} b_{s\alpha}^L \\ b_{s1}^L \\ b_{s1}^R \\ b_{s\alpha}^R \end{pmatrix}, \ s = 1, \dots, m. \tag{7.2.4}$$

Therefore, an MOLP problem with non-fuzzy objective functions (7.2.3) and non-fuzzy constraints (7.2.4) is formed to find a solution $(x)_l$.

- In Step 5, the interval $[\alpha, 1]$ is further split. We suppose there are $(l+1)$ nodes $\lambda_i\ (i = 0, 2, 4, \ldots, 2l)$ in the interval $[\alpha, 1]$, and l new nodes $\lambda_i\ (i = 1, 3, \ldots, 2l-1)$ are inserted. The relationship between these new inserted nodes and previous ones is:

$$\lambda_{2i+1} = \frac{\lambda_{2i} + \lambda_{2i+2}}{2}, \quad i = 0, 1, \ldots, l-1. \tag{7.2.5}$$

  Therefore, each fuzzy objective function $\tilde{f}_i(x) = \tilde{c}_i x$ is converted into $2*(2l+1)$ non-fuzzy objective functions, and the same conversion for the constraints. Suppose that the number of fuzzy objective functions and fuzzy constraints are *k* and *m*, respectively, the total number of non-fuzzy objective functions and non-fuzzy constraints are $2*k*(2l+1)$ and $2*m*(2l+1)$, respectively. The solution $(x)_{2l}$ is now based on the set of updated (including original) non-fuzzy objective functions and non-fuzzy constraints.
- In Step 6, if the difference between solutions $(x)_l$ and $(x)_{2l}$ is within the preset tolerance, the solution in the current step, *i.e.*, $(x)_{2l}$ is the final result; otherwise, the method needs more iterations by inserting notes.
- We have seen that each step of the method includes two parts. One is to convert an FMOLPα problem into a non-fuzzy (MOLPαλ)l problem. The other is to solve the (MOLPαλ)l problem, that is, to derive an optimal solution from it.

Figure 7.1 shows the working process of the FMOLP method.

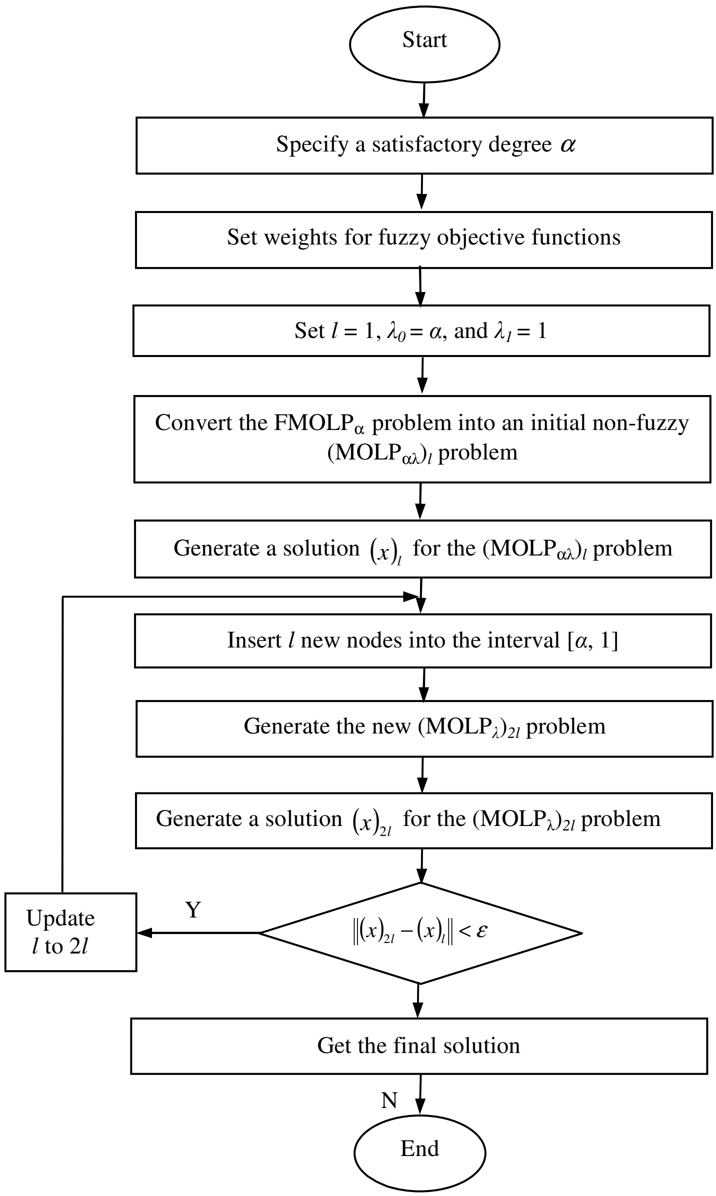


Fig. 7.1: Working process of the FMOLP method

7.2.2 *A numeral example*

Consider the following FMOLP$_\alpha$ problem with two fuzzy objective functions and four fuzzy constraints:

$$\max \tilde{f}(x) = \max \begin{pmatrix} \tilde{f}_1(x) \\ \tilde{f}_2(x) \end{pmatrix} = \max \begin{pmatrix} \tilde{c}_{11}x_1 + \tilde{c}_{12}x_2 \\ \tilde{c}_{21}x_1 + \tilde{c}_{22}x_2 \end{pmatrix} \tag{7.2.6}$$

$$\text{s.t.} \quad \begin{cases} \tilde{a}_{11}x_1 + \tilde{a}_{12}x_2 \preceq \tilde{b}_1 \\ \tilde{a}_{21}x_1 + \tilde{a}_{22}x_2 \preceq \tilde{b}_2 \\ \tilde{a}_{31}x_1 + \tilde{a}_{32}x_2 \preceq \tilde{b}_3 \\ \tilde{a}_{41}x_1 + \tilde{a}_{42}x_2 \preceq \tilde{b}_4 \\ x_1 \geq 0;\ x_2 \geq 0 \end{cases}$$

The membership functions of fuzzy parameters of the objective functions and constraints are as follows:

$$\mu_{\tilde{c}_{11}}(x) = \begin{cases} 0 & x < 1 \text{ or } 16 < x \\ (x^2 - 1)/3 & 1 \leq x < 2 \\ 1 & 2 \leq x \leq 3 \\ (256 - x^2)/247 & 3 < x \leq 16 \end{cases} \qquad \mu_{\tilde{c}_{12}}(x) = \begin{cases} 0 & x < 0 \text{ or } 24 < x \\ x^2 & 0 \leq x < 1 \\ 1 & 1 \leq x \leq 2 \\ (576 - x^2)/572 & 2 < x \leq 24 \end{cases}$$

$$\mu_{\tilde{c}_{21}}(x) = \begin{cases} 0 & x < -2 \text{ or } 3 < x \\ (4 - x^2)/3 & -2 \leq x < -1 \\ 1 & -1 \leq x \leq 0 \\ (13 - x)/13 & 0 < x \leq 13 \end{cases} \qquad \mu_{\tilde{c}_{22}}(x) = \begin{cases} 0 & x < 1 \text{ or } 25 < x \\ (x^2 - 1)/3 & 1 \leq x < 2 \\ 1 & 2 \leq x \leq 3 \\ (25 - x)/22 & 3 < x \leq 25 \end{cases}$$

$$\mu_{\tilde{a}_{11}}(x) = \begin{cases} 0 & x < -2 \text{ or } 0 < x \\ 2x + 4 & -2 \leq x < -1.5 \\ 1 & -1.5 \leq x \leq -0.5 \\ 4x^2 & -0.5 < x \leq 0 \end{cases} \qquad \mu_{\tilde{a}_{12}}(x) = \begin{cases} 0 & x < 0 \text{ or } 12 < x \\ x/2 & 0 \leq x < 2 \\ 1 & 2 \leq x \leq 4 \\ (e^{12} - e^x)/(e^{12} - e^4) & 4 < x \leq 12 \\ & 12 < x \end{cases}$$

$$\mu_{\tilde{a}_{21}}(x) = \begin{cases} 0 & x < 0 \text{ or } 10 < x \\ 2x & 0 \leq x < 0.5 \\ 1 & 0.5 \leq x \leq 1.5 \\ (e^{10} - e^x)/(e^{10} - e^{1.5}) & 1.5 < x \leq 10 \end{cases} \qquad \mu_{\tilde{a}_{22}}(x) = \begin{cases} 0 & x < 0 \text{ or } 18 < x \\ (e^x - 1)/(e^2 - 1) & 0 \leq x < 2 \\ 1 & 2 \leq x \leq 4 \\ (e^{18} - e^x)/(e^{18} - e^4) & 4 < x \leq 18 \end{cases}$$

$$\mu_{\tilde{a}_{31}}(x) = \begin{cases} 0 & x < 1 \text{ or } 18 < x \\ (e^x - e^1)/(e^3 - e^1) & 1 \leq x < 3 \\ 1 & 3 \leq x \leq 5 \\ (18 - x)/13 & 5 < x \leq 18 \end{cases} \qquad \mu_{\tilde{a}_{32}}(x) = \begin{cases} 0 & x < 0 \text{ or } 10 < x \\ x^2/4 & 0 \leq x < 2 \\ 1 & 2 \leq x \leq 4 \\ (10 - x)/6 & 4 < x \leq 10 \end{cases}$$

$$\mu_{\tilde{a}_{41}}(x)=\begin{cases}0 & x<0 \text{ or } 10<x\\ x/2 & 0\le x<2\\ 1 & 2\le x\le 4\\ \left(e^{10}-e^{x}\right)/\left(e^{10}-e^{4}\right) & 4<x\le 10\end{cases}\qquad \mu_{\tilde{a}_{42}}(x)=\begin{cases}0 & x<0 \text{ or } 10<x\\ 2x & 0\le x<0.5\\ 1 & 0.5\le x\le 1.5\\ \left(1000-x^{3}\right)/996.625 & 1.5<x\le 10\\ 0 & \end{cases}$$

$$\mu_{\tilde{b}_{1}}(x)=\begin{cases}0 & x<18 \text{ or } 30<x\\ \left(e^{x}-e^{18}\right)/\left(e^{20}-e^{18}\right) & 18\le x<20\\ 1 & 20\le x\le 22\\ \left(27000-x^{3}\right)/16352 & 22<x\le 30\end{cases}\qquad \mu_{\tilde{b}_{2}}(x)=\begin{cases}0 & x<24 \text{ or } 40<x\\ \left(e^{x}-e^{24}\right)/\left(e^{26}-e^{24}\right) & 24\le x<26\\ 1 & 26\le x\le 28\\ (40-x)/12 & 28<x\le 40\end{cases}$$

$$\mu_{\tilde{b}_{3}}(x)=\begin{cases}0 & x<42 \text{ or } 60<x\\ (x-42)/2 & 42\le x<44\\ 1 & 44\le x\le 46\\ \left(e^{60}-e^{x}\right)/\left(e^{60}-e^{46}\right) & 46<x\le 60\end{cases}\qquad \mu_{\tilde{b}_{1}}(x)=\begin{cases}0 & x<27 \text{ or } 40<x\\ \left(x^{2}-729\right)/112 & 27\le x<29\\ 1 & 29\le x\le 31\\ \left(64000-x^{3}\right)/34209 & 31<x\le 40\end{cases}$$

In this example, the fuzzy parameters are represented in different forms of membership functions, such as linear, quadratic, cubic, and exponent.

We now show the process of getting the solution for the problem by using the FMOLP method.

Step 1: Set $\alpha = 0$, the FMOLP$_\alpha$ problem becomes a general FMOLP problem.

Step 2: Give equal weights w_1 and w_2 for objective functions $\tilde{f}_1$ and $\tilde{f}_2$, respectively, and $w_1 + w_2 = 1$.

Step 3: We convert the FMOLP$_\alpha$ problem into a non-fuzzy MOLP$_\lambda$ problem as follows:

$$\max \begin{bmatrix} \sqrt{3\lambda+1} & \sqrt{\lambda} \\ \sqrt{256-247\lambda} & \sqrt{576-572\lambda} \\ \sqrt{4-3\lambda} & \sqrt{3\lambda+1} \\ 13-13\lambda & 25-22\lambda \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \tag{7.2.7}$$

$$\text{st.}\begin{bmatrix} (\lambda-4)/2 & 2\lambda \\ \sqrt{\lambda}/2 & \ln\left(e^{12}-\left(e^{12}-e^{4}\right)\lambda\right) \\ \lambda/2 & \ln\left(\left(e^{2}-1\right)\lambda+1\right) \\ \ln\left(e^{10}-\left(e^{10}-e^{1.5}\right)\lambda\right) & \ln\left(e^{18}-\left(e^{18}-e^{4}\right)\lambda\right) \\ \ln\left(\left(e^{3}-e^{1}\right)\lambda+e^{1}\right) & 2\sqrt{\lambda} \\ 18-13\lambda & 10-6\lambda \\ 2\lambda & \lambda/2 \\ \ln\left(e^{10}-\left(e^{10}-e^{4}\right)\lambda\right) & \sqrt[3]{1000-996.625\lambda} \end{bmatrix}\begin{bmatrix} x_1 \\ x_1 \end{bmatrix} \le \begin{bmatrix} \ln\left(\left(e^{20}-e^{18}\right)\lambda+e^{18}\right) \\ \sqrt[3]{27000-16352\lambda} \\ \ln\left(\left(e^{26}-e^{24}\right)\lambda+e^{24}\right) \\ 40-12\lambda \\ 42+2\lambda \\ \ln\left(e^{60}-\left(e^{60}-e^{46}\right)\lambda\right) \\ \sqrt{729+112\lambda} \\ \sqrt[3]{64000-34209\lambda} \end{bmatrix}$$

where $\forall\lambda \in [0,1]$.

Step 4: Refer to the MOLP$_\lambda$ problem, the interval $[0,1]$ is not split, and only $\lambda_0 = 0$ and $\lambda_1 = 1$ are considered. Totally, 8 non-fuzzy objective functions and 16 non-fuzzy constraints are then generated. The result of this conversion is as follows:

$$\max\begin{bmatrix} 1 & 0 \\ 2 & 1 \\ 16 & 24 \\ 3 & 2 \\ 2 & 1 \\ 1 & 2 \\ 13 & 25 \\ 0 & 3 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \tag{7.2.8}$$

s.t.

$$\begin{bmatrix} -2 & 0 \\ -1.5 & 2 \\ 0 & 12 \\ 0.5 & 4 \\ 0 & 0 \\ 0.5 & 2 \\ 10 & 18 \\ 1.5 & 4 \\ 1 & 0 \\ 3 & 2 \\ 18 & 10 \\ 5 & 4 \\ 0 & 0 \\ 2 & 0.5 \\ 10 & 10 \\ 4 & 1.5 \end{bmatrix} \begin{bmatrix} x_1 \\ x_1 \end{bmatrix} \le \begin{bmatrix} 18 \\ 20 \\ 30 \\ 22 \\ 24 \\ 26 \\ 40 \\ 28 \\ 42 \\ 44 \\ 60 \\ 46 \\ 27 \\ 29 \\ 40 \\ 31 \end{bmatrix}$$

By using a classical linear programming approach, we have an optimal solution:

$$x_1^* = 1.5179\,,\, x_2^* = 1.3790\,, \tag{7.2.9}$$

and corresponded fuzzy objective values:

$$\begin{cases} \tilde{f}_1^*\left(x_1^*, x_2^*\right) = 1.5179\tilde{c}_{11} + 1.3790\tilde{c}_{12} \\ \tilde{f}_2^*\left(x_1^*, x_2^*\right) = 1.5179\tilde{c}_{21} + 1.3790\tilde{c}_{22} \end{cases}. \tag{7.2.10}$$

Step 5: One more node is inserted into the interval $[0,1]$, we have $\lambda_0 = 0$, $\lambda_1 = 0.5$, and $\lambda_2 = 1$. Totally, 12 non-fuzzy objective functions and 24 non-fuzzy constraints are generated. Similarly, we have a new optimal solution

$$x_1^* = 1.5985\,,\, x_2^* = 1.1049\,, \tag{7.2.11}$$

and its corresponded optimal fuzzy objective values are

$$\begin{cases} \tilde{f}_1^*\left(x_1^*, x_2^*\right) = 1.5985\tilde{c}_{11} + 1.1049\tilde{c}_{12} \\ \tilde{f}_2^*\left(x_1^*, x_2^*\right) = 1.5985\tilde{c}_{21} + 1.1049\tilde{c}_{22} \end{cases}. \tag{7.2.12}$$

This step will be repeated for as many times as required. Table 7.1 shows the values of optimal solutions in first eight loops.

Table 7.1: The optimal solutions in first eight loops

Loop	x_1^*	x_2^*
1	1.51786	1.37897
2	1.59854	1.10480
3	1.61299	1.02958
4	1.61614	1.02795
5	1.61629	1.02788
6	1.61666	1.02634
7	1.61659	1.02638
8	1.61655	1.02640

Step 6: Before the method starts running, the tolerance ε needs to be preset. Different tolerance will cause the approximate-based method to terminate at different loops. From Table 7.1, we can find that if the tolerance $\varepsilon = 10^{-2}$, the method terminates at loop 5; if the tolerance $\varepsilon = 10^{-4}$, the method will terminate at loop 8.

Suppose we select the tolerance $\varepsilon = 10^{-4}$, then the final optimal solution for the example is

$$x_1^* = 1.6166,\ x_2^* = 1.0164 \tag{7.2.13}$$

and its corresponded optimal fuzzy objective values are

$$\begin{cases} \tilde{f}_1^*\left(x_1^*, x_2^*\right) = 1.6166\tilde{c}_{11} + 1.0164\tilde{c}_{12} \\ \tilde{f}_2^*\left(x_1^*, x_2^*\right) = 1.6166\tilde{c}_{21} + 1.0164\tilde{c}_{22} \end{cases}. \tag{7.2.14}$$

The membership functions of $\tilde{f}_1^*$ and $\tilde{f}_2^*$ (7.2.14) are shown in Fig. 7.2, respectively. The result shows that when $x_1^* = 1.6166$, $x_2^* = 1.0164$, the first objective's value is around from 4.2595 to 6.9025. It may also be acceptable for the value not fully into the interval within a threshold. Similarly, we can understand the meaning of the second objective's value interval.

The values of the weights of objective functions can directly influence the values of optimal solutions of an MOLP problem. Decision makers can set different weights for their objective functions based on their preference, experience, and judgment. In this example, w_1 and w_2 represent the weights of $\tilde{f}_1$ and $\tilde{f}_2$, respectively, and $w_1 = w_2 = 0.5$. When w_1 and w_2 are revised by decision makers, a new optimal solution will be generated. Table 7.2 summarises 11 optimal solutions in which each corresponds a set of specific weights.

From Table 7.2, when $w_1 = 1$ and $w_2 = 0$, the solution $(x_1^*, x_2^*) = (3.2260, 0.1932)$ only concerns the objective function $\tilde{f}_1^*$. While when $w_1 = 0$ and $w_2 = 1$, the solution $(x_1^*, x_2^*) = (0.0, 1.8506)$ more concerns objective function $\tilde{f}_2^*$. When w_1 decreases from 1 to 0 and w_2 increases from 0 to 1 simultaneously, the solution will move from $(3.2260, 0.1932)$ to $(0.0, 1.8506)$ gradually.

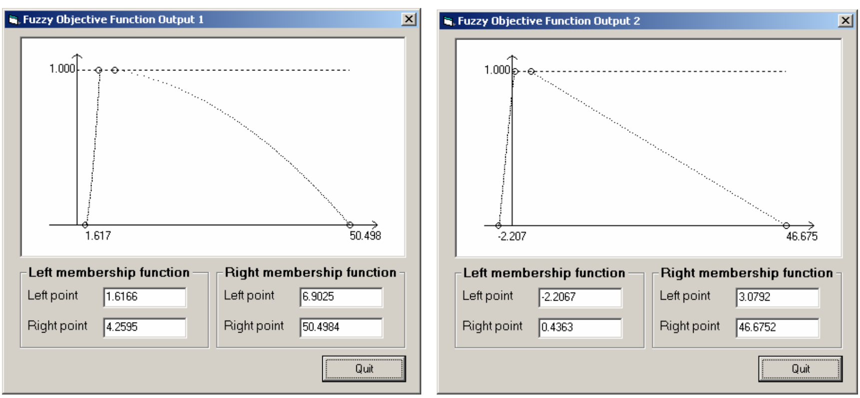


Fig. 7.2: Membership functions of $\tilde{f}_1^*$ and $\tilde{f}_2^*$ in the final solution

Table 7.2: The optimal solutions by revising the weights of objective functions

w_1	w_2	x_1^*	x_2^*	$\tilde{f}_1(x_1^*, x_2^*)$	$\tilde{f}_2(x_1^*, x_2^*)$
1	*0*	3.2260	0.1932	$3.2260\tilde{c}_{11} + 0.1932\tilde{c}_{12}$	$3.2260\tilde{c}_{21} + 0.1932\tilde{c}_{22}$
0.9	*0.1*	2.9141	0.3543	$2.9141\tilde{c}_{11} + 0.3543\tilde{c}_{12}$	$2.9141\tilde{c}_{21} + 0.3543\tilde{c}_{22}$
0.8	*0.2*	2.5954	0.5201	$2.5954\tilde{c}_{11} + 0.5201\tilde{c}_{12}$	$2.5954\tilde{c}_{21} + 0.5201\tilde{c}_{22}$
0.7	*0.3*	2.2721	0.6878	$2.2721\tilde{c}_{11} + 0.6878\tilde{c}_{12}$	$2.2721\tilde{c}_{21} + 0.6878\tilde{c}_{22}$
0.6	*0.4*	1.9454	0.8568	$1.9454\tilde{c}_{11} + 0.8568\tilde{c}_{12}$	$1.9454\tilde{c}_{21} + 0.8568\tilde{c}_{22}$
0.5	*0.5*	1.6166	1.0264	$1.6166\tilde{c}_{11} + 1.0264\tilde{c}_{12}$	$1.6166\tilde{c}_{21} + 1.0264\tilde{c}_{22}$
0.4	*0.6*	1.2872	1.1957	$1.2872\tilde{c}_{11} + 1.1957\tilde{c}_{12}$	$1.2872\tilde{c}_{21} + 1.1957\tilde{c}_{22}$
0.3	*0.7*	0.9589	1.3637	$0.9589\tilde{c}_{11} + 1.3637\tilde{c}_{12}$	$0.9589\tilde{c}_{21} + 1.3637\tilde{c}_{22}$
0.2	*0.8*	0.6337	1.5296	$0.6337\tilde{c}_{11} + 1.5296\tilde{c}_{12}$	$0.6337\tilde{c}_{21} + 1.5296\tilde{c}_{22}$
0.1	*0.9*	0.3134	1.6922	$0.3134\tilde{c}_{11} + 1.6922\tilde{c}_{12}$	$0.3134\tilde{c}_{21} + 1.6922\tilde{c}_{22}$
0	*1*	0	1.8506	$1.8506\tilde{c}_{12}$	$1.8506\tilde{c}_{22}$

7.3 Fuzzy MOLGP

Under some circumstances, decision makers may need to specify their goals for the objective functions, but it may be hard to provide an accurate value for each goal. In this section, we propose a fuzzy multi-objective linear goal programming (FMOLGP) method, which allows decision makers to provide their fuzzy goals for the fuzzy objectives in an FMOLP model. It then finds an optimal solution to reach the proposed fuzzy goals. Obviously, this method deals with two fuzzy issues: fuzzy parameters in a given FMOLP model and fuzzy goals provided by decision makers in the process of finding an optimal solution. A numeral example will further illustrate how this method deals with the two issues.

7.3.1 *Method description*

Considering an FMOLP problem, for the fuzzy multi-objective functions $\tilde{f}(x) = \left(\tilde{f}_1(x), \tilde{f}_2(x), \ldots, \tilde{f}_k(x)\right)^T$, decision makers can specify their fuzzy goals $\tilde{g} = (\tilde{g}_1, \tilde{g}_2, \ldots, \tilde{g}_k)^T$, which reflect the desired values of decision makers for the objective functions.

From the definitions of both FMOLP$_\alpha$ and MOLP$_{\alpha\lambda}$ problems in Chapter 6, when decision makers set up their fuzzy goals under a satisfactory degree α, an optimal solution, which corresponding optimal objective values are the nearest to the related fuzzy goals or better than that, is obtained by solving the following minimax programming problem:

$$
(\text{MOLP}_{\alpha\lambda m})\begin{cases} \min \max \begin{pmatrix} c_\lambda^L x - g_\lambda^L \\ c_\lambda^R x - g_\lambda^R \end{pmatrix}, \forall \lambda \in [\alpha, 1] \\ \text{s.t. } x \in X = \left\{x \in R^n \mid A_\lambda^L x \le b_\lambda^L, A_\lambda^R x \le b_\lambda^R, x \ge 0, \forall \lambda \in [\alpha, 1]\right\} \end{cases} \tag{7.3.1}
$$

where

$$
g_\lambda^L = \left[g_{1\lambda}^L, g_{2\lambda}^L, \cdots, g_{k\lambda}^L\right]^T, \; g_\lambda^R = \left[g_{1\lambda}^R, g_{2\lambda}^R, \cdots, g_{k\lambda}^R\right]^T,
$$

$$
c_\lambda^L = \begin{bmatrix} c_{11\lambda}^L & c_{12\lambda}^L & \cdots & c_{1n\lambda}^L \\ c_{21\lambda}^L & c_{22\lambda}^L & \cdots & c_{2n\lambda}^L \\ \vdots & \vdots & \ddots & \vdots \\ c_{k1\lambda}^L & c_{k2\lambda}^L & \cdots & c_{kn\lambda}^L \end{bmatrix}, \; c_\lambda^R = \begin{bmatrix} c_{11\lambda}^R & c_{12\lambda}^R & \cdots & c_{1n\lambda}^R \\ c_{21\lambda}^R & c_{22\lambda}^R & \cdots & c_{2n\lambda}^R \\ \vdots & \vdots & \ddots & \vdots \\ c_{k1\lambda}^R & c_{k2\lambda}^R & \cdots & c_{kn\lambda}^R \end{bmatrix},
$$

$$
A_\lambda^L = \begin{bmatrix} a_{11\lambda}^L & a_{12\lambda}^L & \cdots & a_{1n\lambda}^L \\ a_{21\lambda}^L & a_{22\lambda}^L & \cdots & a_{2n\lambda}^L \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1\lambda}^L & a_{m2\lambda}^L & \cdots & a_{mn\lambda}^L \end{bmatrix}, \; A_\lambda^R = \begin{bmatrix} a_{11\lambda}^R & a_{12\lambda}^R & \cdots & a_{1n\lambda}^R \\ a_{21\lambda}^R & a_{22\lambda}^R & \cdots & a_{2n\lambda}^R \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1\lambda}^R & a_{m2\lambda}^R & \cdots & a_{mn\lambda}^R \end{bmatrix},
$$

$$
b_\lambda^L = \left[b_{1\lambda}^L, b_{2\lambda}^L, \cdots, b_{m\lambda}^L\right]^T, \; b_\lambda^R = \left[b_{1\lambda}^R, b_{2\lambda}^R, \cdots, b_{m\lambda}^R\right]^T.
$$

We can see that the MOLP$_{\alpha\lambda m}$ problem (7.3.1) has an infinite number of objective functions and an infinite number of constraints. To solve the problem, we give the FMOLGP method, which can be described by the following steps.

Step 1: Give an initial satisfactory degree α $(0 \le \alpha \le 1)$, and the membership functions of $\tilde{c}$ for $\tilde{f}(x) = \tilde{c}x$, $\tilde{a}$ and $\tilde{b}$ for $\tilde{a}x \preceq_\alpha \tilde{b}$.

Step 2: Give weights $w_1, w_2, \ldots, w_k$ for $\tilde{f}_1, \tilde{f}_2, \cdots, \tilde{f}_k$, respectively, and $\sum_{i=1}^{k} w_i = 1$.

Step 3: Specify a set of fuzzy goals $\tilde{g} = (\tilde{g}_1, \tilde{g}_2, \ldots, \tilde{g}_k)^T$, which need to be achieved, for the objective functions.

Step 4: Let the interval $[\alpha, 1]$ be decomposed into l mean sub-intervals with $(l+1)$ nodes λ_i $(i = 0, \cdots, l)$, where $\alpha = \lambda_0 < \lambda_1 < \cdots < \lambda_l = 1$.

We denote:

$$(\text{MOLP}_{\alpha\lambda m})_l \begin{cases} \min \max \begin{pmatrix} c_{i\lambda_j}^L x - g_{i\lambda_{ji}}^L \\ c_{i\lambda_j}^R x - g_{i\lambda_{ji}}^R \end{pmatrix}, i = 1,2,\cdots,k,\ j = 1,2,\cdots,l, \\ \text{s.t. } x \in X^l \end{cases} \quad (7.3.2)$$

where $X^l = \bigcap_i^l X_{\lambda_i}$, $X_{\lambda_i} = \{x \in R^n \mid A_{\lambda_i}^L x \le b_{\lambda_i}^L, A_{\lambda_i}^R x \le b_{\lambda_i}^R, x \ge 0\}$, $\alpha = \lambda_0 < \ldots < \lambda_l = 1$.

Step 5: Set $l = 1$, solve the $(\text{MOLP}_{\alpha\lambda m})_l$ with the solution $(x)_l = (x_1, x_2, \ldots, x_n)_l$, which is subject to the constraint $x \in X^l$.

In this step, the interval $[\alpha, 1]$ is not split initially. So only $\lambda_0 = \alpha$ and $\lambda_1 = 1$ are considered. Then, each fuzzy objective function $\tilde{f}_i(x) = \tilde{c}_i x$ under the fuzzy goal $\tilde{g} = (\tilde{g}_1, \tilde{g}_2, \ldots, \tilde{g}_k)^T$ is converted into four non-fuzzy objective functions:

$$\begin{cases} c_{i\alpha}^L x - g_{i\alpha}^L \\ c_{i1}^L x - g_{i1}^L \\ c_{i1}^R x - g_{i1}^R \\ c_{i\alpha}^R x - g_{i\alpha}^R \end{cases}, \ i = 1, \ldots, k. \quad (7.3.3)$$

Similarly, each fuzzy constraint $\tilde{a}_s x \precsim_{\alpha} \tilde{b}_s$ in the FMOLP_α is converted into four non-fuzzy constraints, which are as follows:

$$\begin{pmatrix} a_{s\alpha}^L x \\ a_{s1}^L x \\ a_{s1}^R x \\ a_{s\alpha}^R x \end{pmatrix} \le \begin{pmatrix} b_{s\alpha}^L \\ b_{s1}^L \\ b_{s1}^R \\ b_{s\alpha}^R \end{pmatrix}, \ s = 1, \ldots, m \quad (7.3.4)$$

Hence, an MOLP problem with non-fuzzy objective functions (7.3.3) and non-fuzzy constraints (7.3.4) is formed to find a solution $(x)_l$.

Step 6: Solve $(\text{MOLP}_{\alpha\lambda m})_{2l}$ with the solution $(x)_{2l}$, which is subject to the constraint $x \in X^{2l}$.

The interval $[\alpha, 1]$ is further split in the step. We suppose there are $(l+1)$ nodes λ_i $(i = 0, 2, 4, \ldots, 2l)$ in the interval $[\alpha, 1]$, and l new nodes λ_i $(i = 1, 3, \ldots, 2l-1)$ are inserted. The relationship between the new inserted nodes and previous ones is:

$$\lambda_{2i+1} = \frac{\lambda_{2i} + \lambda_{2i+2}}{2}, \quad i = 0, 1, \ldots, l-1 \cdot \tag{7.3.5}$$

Therefore, each fuzzy objective function $\tilde{f}_i(x) = \tilde{c}_i x$ under its related fuzzy goal $\tilde{g} = (\tilde{g}_1, \tilde{g}_2, \ldots, \tilde{g}_k)^T$ is converted into $2*(2l+1)$ non-fuzzy objective functions, and the same for the constraints $\tilde{a}x \precsim_\alpha \tilde{b}$. The solution $(x)_{2l}$ is now based on the set of updated (including original) non-fuzzy objective functions and non-fuzzy constraints.

Step 7: Referring to the solutions $(x)_1$ and $(x)_{2l}$, if $\|(x)_{2l} - (x)_l\| < \varepsilon$, the final solution of $\text{MOLP}_{\alpha\lambda m}$ problem is $(x)_{2l}$. Otherwise, update l to $2l$ and go back to Step 6.

The FMOLGP method is shown in Fig. 7.3.

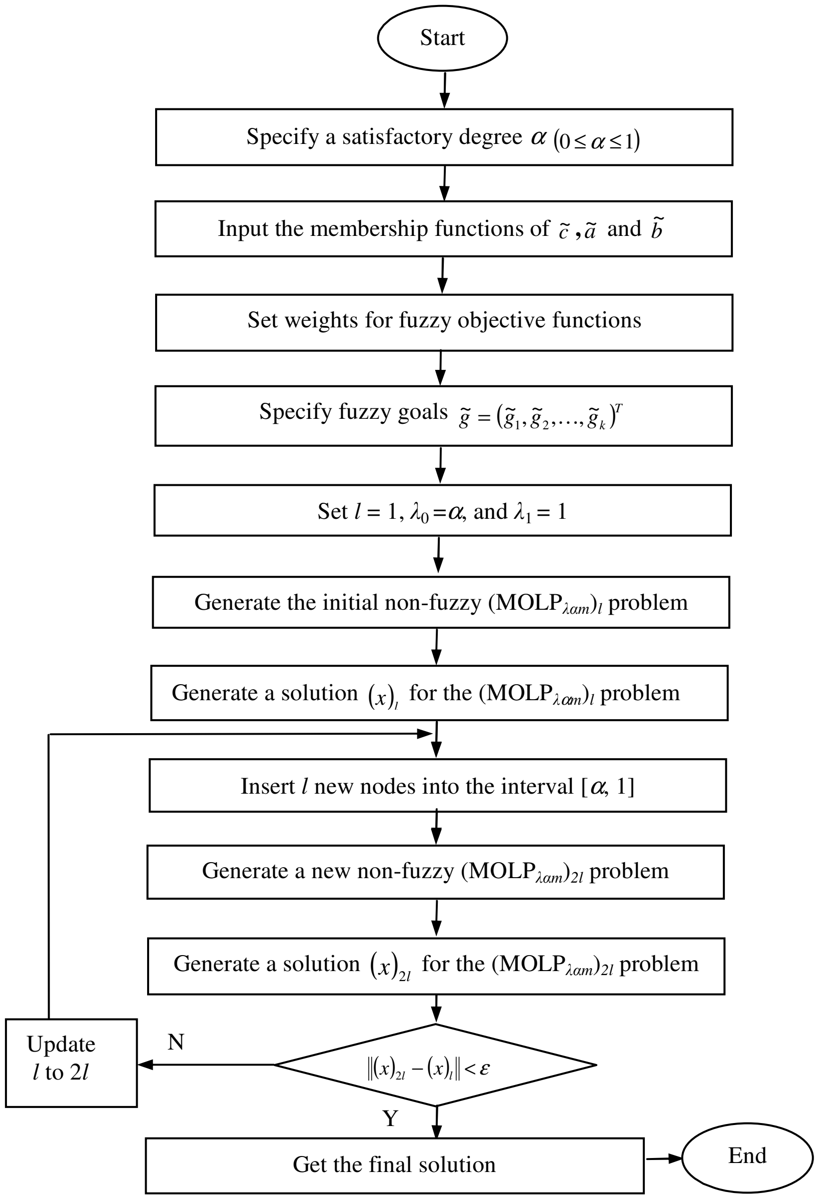


Fig. 7.3: Working process of the FMOLGP method

7.3.2 *A numeral example*

Consider a numeral FMOLP$_\alpha$ problem with two fuzzy objective functions and four fuzzy constraints as follows:

$$\max \tilde{f}(x) = \max \begin{pmatrix} \tilde{f}_1(x) \\ \tilde{f}_2(x) \end{pmatrix} = \max \begin{pmatrix} \tilde{c}_{11}x_1 + \tilde{c}_{12}x_2 \\ \tilde{c}_{21}x_1 + \tilde{c}_{22}x_2 \end{pmatrix} = \max \begin{pmatrix} \tilde{6}x_1 + \tilde{3}x_2 \\ -\tilde{3}x_1 + \tilde{6}x_2 \end{pmatrix} \quad (7.3.6)$$

$$\text{s.t.} \quad \begin{cases} \tilde{a}_{11}x_1 + \tilde{a}_{12}x_2 = -\tilde{1}x_1 + \tilde{3}x_2 \precsim \tilde{b}_1 = \tilde{21} \\ \tilde{a}_{21}x_1 + \tilde{a}_{22}x_2 = \tilde{1}x_1 + \tilde{3}x_2 \precsim \tilde{b}_2 = \tilde{27} \\ \tilde{a}_{31}x_1 + \tilde{a}_{32}x_2 = \tilde{4}x_1 + \tilde{3}x_2 \precsim \tilde{b}_3 = \tilde{45} \\ \tilde{a}_{41}x_1 + \tilde{a}_{42}x_2 = \tilde{3}x_1 + \tilde{1}x_2 \precsim \tilde{b}_4 = \tilde{30} \\ x_1 \ge 0; \quad x_2 \ge 0 \end{cases}$$

The membership functions of fuzzy parameters of the objective functions and constraints are set up as follows:

$$\mu_{\tilde{c}_{11}}(x) = \begin{cases} 0 & x < 5 \text{ or } 8 < x \\ (x^2 - 25)/11 & 5 \le x < 6 \\ 1 & x = 6 \\ (64 - x^2)/28 & 6 < x \le 8 \end{cases} \qquad \mu_{\tilde{c}_{12}}(x) = \begin{cases} 0 & x < 2 \text{ or } 5 < x \\ (x^2 - 4)/5 & 2 \le x < 3 \\ 1 & x = 3 \\ (25 - x^2)/16 & 3 < x \le 5 \end{cases}$$

$$\mu_{\tilde{c}_{21}}(x) = \begin{cases} 0 & x < -4 \text{ or } -1 < x \\ (16 - x^2)/7 & -4 \le x < -3 \\ 1 & x = -3 \\ (x^2 - 1)/8 & -3 < x \le -1 \end{cases} \qquad \mu_{\tilde{c}_{11}}(x) = \begin{cases} 0 & x < 5 \text{ or } 8 < x \\ (x^2 - 25)/11 & 5 \le x < 6 \\ 1 & x = 6 \\ (64 - x^2)/28 & 6 < x \le 8 \end{cases}$$

$$\mu_{\tilde{a}_{11}}(x) = \begin{cases} 0 & x < -2 \text{ or } -0.5 < x \\ (4 - x^2)/3 & -2 \le x < -1 \\ 1 & x = -1 \\ (x^2 - 0.25)/0.75 & -1 < x \le -0.5 \end{cases} \qquad \mu_{\tilde{a}_{12}}(x) = \begin{cases} 0 & x < 2 \text{ or } 5 < x \\ (x^2 - 4)/5 & 2 \le x < 3 \\ 1 & x = 3 \\ (25 - x^2)/16 & 3 < x \le 5 \end{cases}$$

$$\mu_{\tilde{a}_{21}}(x) = \begin{cases} 0 & x < 0.5 \text{ or } 2 < x \\ (x^2 - 0.25)/0.75 & 0.5 \le x < 1 \\ 1 & x = 1 \\ (4 - x^2)/3 & 1 < x \le 2 \end{cases} \qquad \mu_{\tilde{a}_{22}}(x) = \begin{cases} 0 & x < 2 \text{ or } 5 < x \\ (x^2 - 4)/5 & 2 \le x < 3 \\ 1 & x = 3 \\ (25 - x^2)/16 & 3 < x \le 5 \end{cases}$$

$$\mu_{\tilde{c}_{31}}(x)=\begin{cases}0 & x<3 \text{ or } 6<x\\ (x^2-9)/7 & 3\le x<4\\ 1 & x=4\\ (36-x^2)/20 & 4<x\le 6\end{cases} \qquad \mu_{\tilde{a}_{32}}(x)=\begin{cases}0 & x<2 \text{ or } 5<x\\ (x^2-4)/5 & 2\le x<3\\ 1 & x=3\\ (25-x^2)/16 & 3<x\le 5\end{cases}$$

$$\mu_{\tilde{a}_{41}}(x)=\begin{cases}0 & x<2 \text{ or } 5<x\\ (x^2-4)/5 & 2\le x<3\\ 1 & x=3\\ (25-x^2)/16 & 3<x\le 5\end{cases} \qquad \mu_{\tilde{a}_{42}}(x)=\begin{cases}0 & x<0.5 \text{ or } 2<x\\ (x^2-0.25)/0.75 & 0.5\le x<1\\ 1 & x=1\\ (4-x^2)/3 & 1<x\le 2\end{cases}$$

$$\mu_{\tilde{b}_1}(x)=\begin{cases}0 & x<19 \text{ or } 25<x\\ (x^2-361)/80 & 19\le x<21\\ 1 & x=21\\ (625-x^2)/184 & 21<x\le 25\end{cases} \qquad \mu_{\tilde{b}_2}(x)=\begin{cases}0 & x<25 \text{ or } 31<x\\ (x^2-625)/104 & 25\le x<27\\ 1 & x=27\\ (961-x^2)/232 & 27<x\le 31\end{cases}$$

$$\mu_{\tilde{b}_3}(x)=\begin{cases}0 & x<43 \text{ or } 49<x\\ (x^2-1849)/176 & 43\le x<45\\ 1 & x=45\\ (2401-x^2)/376 & 45<x\le 49\end{cases} \qquad \mu_{\tilde{b}_4}(x)=\begin{cases}0 & x<28 \text{ or } 34<x\\ (x^2-764)/116 & 28\le x<30\\ 1 & x=30\\ (1156-x^2)/256 & 30<x\le 34\end{cases}$$

We now show the process of finding an solution for the problem by using this method.

Step 1: We give an initial value of the satisfactory degree $\alpha = 0.2$, and input the membership functions of $\tilde{c}$ for objective functions $\tilde{f}(x) = \tilde{c}x$, $\tilde{a}$ and $\tilde{b}$ for constraints $\tilde{a}x \preceq \tilde{b}$. For example, the membership function of $\tilde{c}_{11}$ is given as shown in Fig. 7.4.

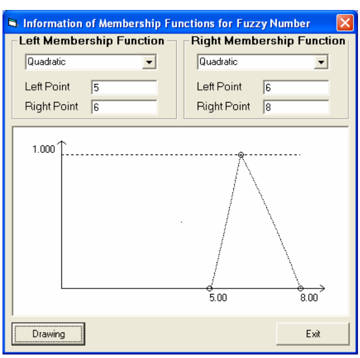


Fig. 7.4: Membership function of fuzzy parameter $\tilde{c}_{11}$

Step 2: Give equal weights w_1 and w_2 for objective functions $\tilde{f}_1$ and $\tilde{f}_2$, respectively, *i.e.*, $w_1 = w_2 = 0.5$

Step 3: Specify fuzzy goals $(\tilde{g}_1, \tilde{g}_2)$ by corresponded membership functions as follows:

$$\mu_{\tilde{g}_1}(x) = \begin{cases} 0 & x < 15 \text{ or } 30 < x \\ (x^2 - 225)/175 & 15 \le x < 20 \\ 1 & x = 20 \\ (900 - x^2)/500 & 20 < x \le 30 \end{cases} \tag{7.3.7}$$

$$\mu_{\tilde{g}_2}(x) = \begin{cases} 0 & x < 4 \text{ or } 15 < x \\ (x^2 - 16)/48 & 4 \le x < 8 \\ 1 & x = 8 \\ (225 - x^2)/161 & 8 < x \le 15 \end{cases} \tag{7.3.8}$$

The membership functions of the two fuzzy goals $(\tilde{g}_1, \tilde{g}_2)$ are shown in Fig. 7.5. The first fuzzy goal $\tilde{g}_1$ is around 20, and the second one $\tilde{g}_2$ is about 8.

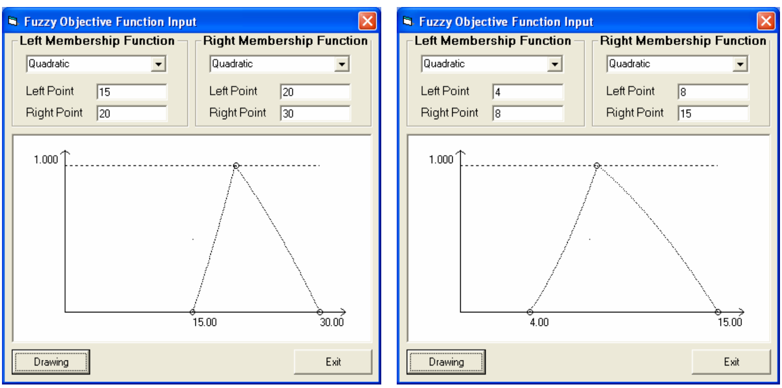


Fig. 7.5: Membership functions of two fuzzy goals $(\tilde{g}_1, \tilde{g}_2)$

Steps 4-7: Under the satisfactory degree $\alpha = 0.2$ and the fuzzy goals in (7.3.7) and (7.3.8), the FMOLP$_\alpha$ problem is converted into a non-fuzzy MOLP$_{\alpha\lambda m}$ problem as follows:

$$\min\ \max \begin{bmatrix} \sqrt{11\lambda+25} & \sqrt{5\lambda+4} \\ \sqrt{64-28\lambda} & \sqrt{25-16\lambda} \\ -\sqrt{16-7\lambda} & \sqrt{11\lambda+25} \\ -\sqrt{8\lambda+1} & \sqrt{64-28\lambda} \end{bmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} - \begin{pmatrix} \sqrt{175\lambda+225} \\ \sqrt{900-500\lambda} \\ \sqrt{48\lambda+16} \\ \sqrt{225-161\lambda} \end{pmatrix} \tag{7.3.9}$$

$$\text{s.t.} \begin{bmatrix} -\sqrt{4-3\lambda} & \sqrt{5\lambda+4} \\ -\sqrt{0.75\lambda+0.25} & \sqrt{25-16\lambda} \\ \sqrt{0.75\lambda+0.25} & \sqrt{5\lambda+4} \\ \sqrt{4-3\lambda} & \sqrt{25-16\lambda} \\ \sqrt{7\lambda+9} & \sqrt{5\lambda+4} \\ \sqrt{36-20\lambda} & \sqrt{25-16\lambda} \\ \sqrt{5\lambda+4} & \sqrt{0.75\lambda+0.25} \\ \sqrt{25-16\lambda} & \sqrt{4-3\lambda} \end{bmatrix} \begin{bmatrix} x_1 \\ x_1 \end{bmatrix} \le \begin{bmatrix} \sqrt{80\lambda+361} \\ \sqrt{625-184\lambda} \\ \sqrt{104\lambda+625} \\ \sqrt{961-232\lambda} \\ \sqrt{176\lambda+1849} \\ \sqrt{2401-376\lambda} \\ \sqrt{116\lambda+764} \\ \sqrt{1156-256\lambda} \end{bmatrix}$$

where $\forall\lambda \in [\alpha,1]$.

Referring to the MOLP$_{\alpha\lambda m}$ problem in (7.3.9), as initially the interval $[\alpha,1]$ is not split, and only $\lambda_0 = 0.2$ and $\lambda_1 = 1$ are considered, totally, 8 non-fuzzy objective functions and 16 non-fuzzy constraints are generated. From (7.3.9), the result of the conversion is as follows:

$$\max f(x) = \max \begin{bmatrix} \sqrt{27.2} & \sqrt{5} \\ 6 & 3 \\ \sqrt{58.4} & \sqrt{21.8} \\ 6 & 3 \\ -\sqrt{14.6} & \sqrt{27.2} \\ -3 & 6 \\ -\sqrt{2.6} & \sqrt{58.4} \\ -3 & 6 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \tag{7.3.10}$$

s.t.

$$\begin{bmatrix} -\sqrt{3.4} & \sqrt{5} \\ -1 & 3 \\ -\sqrt{0.4} & \sqrt{21.8} \\ -1 & 3 \\ \sqrt{0.4} & \sqrt{5} \\ 1 & 3 \\ \sqrt{3.4} & \sqrt{21.8} \\ 1 & 3 \\ \sqrt{10.4} & \sqrt{5} \\ 4 & 3 \\ \sqrt{32} & \sqrt{21.8} \\ 4 & 3 \\ \sqrt{5} & \sqrt{0.4} \\ 3 & 1 \\ \sqrt{21.8} & \sqrt{3.4} \\ 3 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_1 \end{bmatrix} \leq \begin{bmatrix} \sqrt{377} \\ 21 \\ \sqrt{588.2} \\ 21 \\ \sqrt{645.8} \\ 27 \\ \sqrt{914.6} \\ 27 \\ \sqrt{1884.2} \\ 45 \\ \sqrt{2325.8} \\ 45 \\ \sqrt{787.2} \\ 30 \\ \sqrt{1104.8} \\ 30 \end{bmatrix}$$

The interval $[\alpha, 1]$ is further split. Three nodes are considered in this step, they are $\lambda_0 = 0.2$, $\lambda_1 = 0.6$, and $\lambda_2 = 1$. Totally, 12 non-fuzzy objective functions and 24 non-fuzzy constraints are generated.

The process will be repeated until the difference between $(x)_l$ and $(x)_{2l}$ is within a preset tolerance.

Finally, we have an optimal solution

$$\begin{cases} x_1^* = 2.1455 \\ x_2^* = 2.5418 \end{cases}, \tag{7.3.11}$$

and its corresponded optimal fuzzy objective values

$$\begin{cases} \tilde{f}_1^*\left(x_1^*, x_2^*\right) = \tilde{f}_1^*\left(2.1455, 2.5415\right) = 2.1455\tilde{c}_{11} + 2.5415\tilde{c}_{12} \\ \tilde{f}_2^*\left(x_1^*, x_2^*\right) = \tilde{f}_2^*\left(2.1455, 2.5415\right) = 2.1455\tilde{c}_{21} + 2.5415\tilde{c}_{22} \end{cases}, \tag{7.3.12}$$

which are shown in Fig. 7.6, respectively. From Fig. 7.6, we can see that the first optimal fuzzy objective $\tilde{f}_1^*$ is about 20.4985, and the second one $\tilde{f}_2^*$ is about 8.8145.

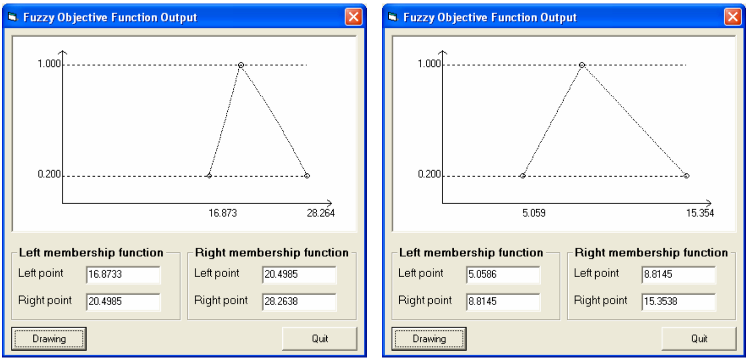


Fig. 7.6: Membership functions of $\tilde{f}_1^*(x_1^*, x_2^*)$ and $\tilde{f}_2^*(x_1^*, x_2^*)$

7.4 Interactive FMOLP

Many decision makers prefer an interactive approach to find an optimal solution for a decision problem as such an approach enables decision makers to directly engage in the problem solving process. In this section, we propose an interactive FMOLP (IFMOLP) method, which not only allows decision makers to give their fuzzy goals, but also allows them to continuously revise and adjust their fuzzy goals. In this way, decision makers can explore various optimal solutions under their goals, and then choose the most satisfactory one. We also supply a numeral example to illustrate how to use this method.

7.4.1 *Method description*

From the definitions of both FMOLP$_\alpha$ and MOLP$_{\alpha\lambda}$ problems, decision makers can set up their fuzzy goals $\tilde{g} = (\tilde{g}_1, \tilde{g}_2, \ldots, \tilde{g}_k)^T$ under a satisfactory degree α. Its corresponded optimal solution, which results in the objective values being the nearest to the fuzzy goals, is obtained by solving the following minimax problem:

$$
(\text{MOLP}_{\alpha\lambda m})\begin{cases}\min\ \max\begin{pmatrix} c_\lambda^L x - g_\lambda^L \\ c_\lambda^R x - g_\lambda^R \end{pmatrix}, \forall \lambda \in [\alpha,1] \\ \text{s.t.}\quad x \in X = \left\{x \in R^n \mid A_\lambda^L x \le b_\lambda^L, A_\lambda^R x \le b_\lambda^R, x \ge 0, \forall \lambda \in [\alpha,1]\right\}\end{cases} \tag{7.4.1}
$$

where

$$
g_\lambda^L = \left[g_{1\lambda}^L, g_{2\lambda}^L, \cdots, g_{k\lambda}^L\right]^T, \quad g_\lambda^R = \left[g_{1\lambda}^R, g_{2\lambda}^R, \cdots, g_{k\lambda}^R\right]^T,
$$

$$
c_\lambda^L = \begin{bmatrix} c_{11\lambda}^L & c_{12\lambda}^L & \cdots & c_{1n\lambda}^L \\ c_{21\lambda}^L & c_{22\lambda}^L & \cdots & c_{2n\lambda}^L \\ \vdots & \vdots & \ddots & \vdots \\ c_{k1\lambda}^L & c_{k2\lambda}^L & \cdots & c_{kn\lambda}^L \end{bmatrix}, \quad c_\lambda^R = \begin{bmatrix} c_{11\lambda}^R & c_{12\lambda}^R & \cdots & c_{1n\lambda}^R \\ c_{21\lambda}^R & c_{22\lambda}^R & \cdots & c_{2n\lambda}^R \\ \vdots & \vdots & \ddots & \vdots \\ c_{k1\lambda}^R & c_{k2\lambda}^R & \cdots & c_{kn\lambda}^R \end{bmatrix},
$$

$$
A_\lambda^L = \begin{bmatrix} a_{11\lambda}^L & a_{12\lambda}^L & \cdots & a_{1n\lambda}^L \\ a_{21\lambda}^L & a_{22\lambda}^L & \cdots & a_{2n\lambda}^L \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1\lambda}^L & a_{m2\lambda}^L & \cdots & a_{mn\lambda}^L \end{bmatrix}, \quad A_\lambda^R = \begin{bmatrix} a_{11\lambda}^R & a_{12\lambda}^R & \cdots & a_{1n\lambda}^R \\ a_{21\lambda}^R & a_{22\lambda}^R & \cdots & a_{2n\lambda}^R \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1\lambda}^R & a_{m2\lambda}^R & \cdots & a_{mn\lambda}^R \end{bmatrix}, \tag{7.4.2}
$$

$$
b_\lambda^L = \left[b_{1\lambda}^L, b_{2\lambda}^L, \cdots, b_{m\lambda}^L\right]^T, \quad b_\lambda^R = \left[b_{1\lambda}^R, b_{2\lambda}^R, \cdots, b_{m\lambda}^R\right]^T.
$$

Let the interval $[\alpha, 1]$ be decomposed into l mean sub-intervals with $(l+1)$ nodes $\lambda_i\ (i = 0, \cdots, l)$, which $\alpha = \lambda_0 < \lambda_1 < \cdots < \lambda_l = 1$.

We denote:

$$
(\text{MOLP}_{\alpha\lambda m})_l\begin{cases}\min\ \max\begin{pmatrix} c_{i\lambda_j}^L x - g_{i\lambda_j}^L \\ c_{i\lambda_j}^R x - g_{i\lambda_j}^R \end{pmatrix}, i = 1,2,\cdots,k,\ j = 1,2,\cdots,l, \\ \text{s.t.}\qquad x \in X^l\end{cases} \tag{7.4.3}
$$

where $X^l = \bigcap_i^l X_{\lambda_i}$, $X_{\lambda_i} = \left\{x \in R^n \mid A_{\lambda_i}^L x \le b_{\lambda_i}^L, A_{\lambda_i}^R x \le b_{\lambda_i}^R, x \ge 0\right\}, \lambda \in [\alpha, 1]$.

This method consists of 11 steps under two stages. Stage 1 aims to find an initial optimal solution for the problem. Stage 2 is an interactive process in which when decision makers specify a set of fuzzy goals for related objective functions, an optimal solution is generated. By revising fuzzy goals, this method will provide decision makers with a series of optimal solutions. Hence, decision makers can select the most suitable one on the basis of their preference, judgment, and experience.

The method is described as follows:

Stage 1: Initialisation

Step 1: Select an initial satisfactory degree α $(0 \le \alpha \le 1)$, give the membership function of $\tilde{c}$ for $\tilde{f}(x) = \tilde{c}x$, $\tilde{a}$ and $\tilde{b}$ for $\tilde{a}x \precsim_{\alpha} \tilde{b}$, and set weights for fuzzy objective functions by decision makers.

Step 2: Set $l = 1$, then solve

$$(\mathrm{MOLP}_{\alpha\lambda})_l \begin{cases} \max \begin{pmatrix} c_{i\lambda_j}^{L} x \\ c_{i\lambda_j}^{R} x \end{pmatrix}, i = 1, \cdots, k; \quad j = 0, 1, \cdots, l, \\ \text{s.t. } x \in X^{l} \end{cases} \tag{7.4.4}$$

with the solution $(x)_l$, where $(x)_l = (x_1, x_2, \ldots, x_n)_l$, and the solution obtained is subject to the constraint $x \in X^{l}$.

Step 3: Solve the $(\mathrm{MOLP}_{\alpha\lambda})_{2l}$ with the solution $(x)_{2l}$, subject to the constraint $x \in X^{2l}$.

The interval $[\alpha, 1]$ is further split. Suppose there are $(l+1)$ nodes λ_i $(i = 0, 2, 4, \ldots, 2l)$ in the interval, and l new nodes λ_i $(i = 1, 3, \ldots, 2l-1)$ are inserted. The relationship between the new nodes and previous ones is:

$$\lambda_{2i+1} = \frac{\lambda_{2i} + \lambda_{2i+2}}{2}, \quad i = 0, 1, \ldots, l-1 \cdot \tag{7.4.5}$$

Each of the fuzzy objective functions is converted into $2*(2l+1)$ non-fuzzy objective functions, and the same conversion happens for the constraints $\tilde{a}_i x \precsim_{\alpha} \tilde{b}_i$. The solution $(x)_{2l}$ is now based on the set of updated (including original) non-fuzzy objective functions and non-fuzzy constraints.

Step 4: If $\|(x)_{2l} - (x)_l\| < \varepsilon$, then $(x)_{2l}$ is the final solution of the $\mathrm{MOLP}_{\alpha\lambda}$ problem. Otherwise, update l to $2l$ and go back to Step 3.

Step 5: If the corresponded Pareto optimal solution x^{*} exists, go forward to Step 6. Otherwise, decision makers must go back to Step 1 to reassign a degree α (give a higher value for the degree α).

Step 6: If decision makers are satisfied with the Pareto optimal solution, the interactive process terminates. Otherwise, go to Stage 2.

Stage 2: Iteration

As decision makers are not satisfied with the obtained solution in the *Initialisation* stage (or the previous iteration phase), they specify their fuzzy goals (or revised current goals) for the fuzzy objective functions. A new compromise solution is then generated. This process will terminate when decision makers find their satisfactory solution.

Step 7: Give a set of new fuzzy goals or revise current fuzzy goals by decision makers. At the same time, a satisfactory degree α can be revised as well. The original decision problem is therefore covered into an $(\text{MOLP}_{\alpha\lambda m})_l$ problem.

Step 8: Set $l=1$, solve the $(\text{MOLP}_{\alpha\lambda m})_l$ with the solution $(x)_l$, which is subject to the constraint $x \in X^l$.

Let $\lambda_0 = \alpha$ and $\lambda_1 = 1$ in the interval $[\alpha, 1]$, each fuzzy objective function $\tilde{f}_i(x) = \tilde{c}_i x$ under the fuzzy goal $\tilde{g} = (\tilde{g}_1, \tilde{g}_2, \ldots, \tilde{g}_k)^T$ and related constraints are converted into non-fuzzy, as described in (7.3.3) and (7.3.4)

Step 9: Solve the $(\text{MOLP}_{\alpha\lambda m})_{2l}$ with the solution $(x)_{2l}$, which is subject to the constraint $x \in X^{2l}$.

Similar as Step 6 in 7.3.1, the interval $[\alpha, 1]$ is further split, and new nodes are inserted further. Fuzzy objective functions under related fuzzy goals and constraints are converted into non-fuzzy again. A new solution $(x)_{2l}$ is generated.

Step 10: If $\|(x)_{2l} - (x)_l\| < \varepsilon$, then $(x)_{2l}$ is the final solution of the $\text{MOLP}_{\alpha\lambda m}$ problem. Otherwise, update l to $2l$ and go back to Step 9.

Step 11: If decision makers are satisfied with the current Pareto optimal solution obtained in Step 10, the interactive process terminates.

The current optimal solution is the final satisfactory solution to decision makers. Otherwise, go back to Step 7.

We now give further explanations for this method:

- Definition 5.3.13 is about ranking two *n*-dimensional fuzzy numbers under a satisfactory degree α. This definition is the foundation for the comparison of fuzzy objective functions and left- and right-hand-side of fuzzy constraints in an FMOLP problem. In Step 5, if the Pareto optimal solution does not exist under a satisfactory degree α, by replacing this α with a higher value may derive a Pareto optimal solution.
- In Step 7, decision makers can improve their goals for some unsatisfactory objectives by sacrificing the goals of others. The new fuzzy goals can be given directly by a new fuzzy number vector or by increasing/decreasing the values of its corresponded objective functions in a current Pareto optimal solution.

Figure 7.7 shows the working process of the IFMOLP method.

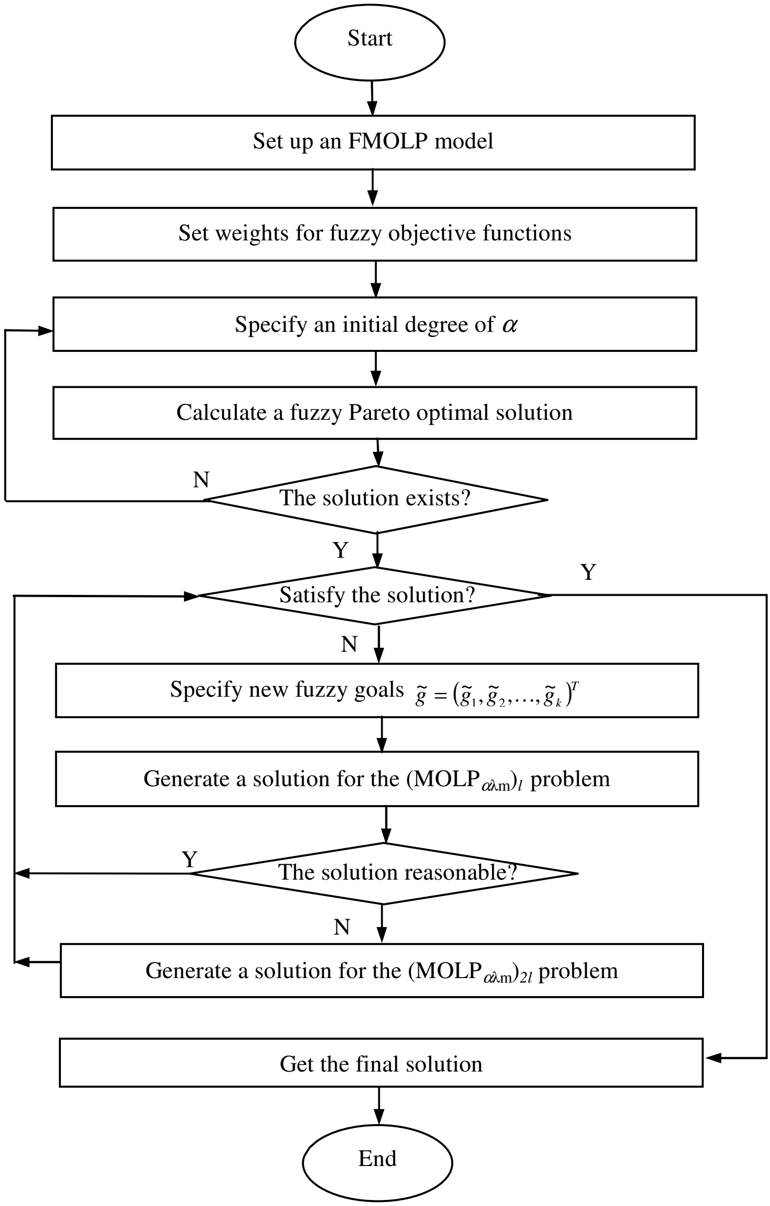


Fig. 7.7: Working process of the IFMOLP method

7.4.2 *A numeral example*

Consider a numeral FMOLP$_\alpha$ problem as follows:

$$\max\ \tilde{f}(x) = \max\begin{pmatrix}\tilde{f}_1(x)\\ \tilde{f}_2(x)\end{pmatrix} = \max\begin{pmatrix}\tilde{c}_{11}x_1 + \tilde{c}_{12}x_2\\ \tilde{c}_{21}x_1 + \tilde{c}_{22}x_2\end{pmatrix} = \max\begin{pmatrix}\tilde{4}x_1 + \tilde{2}x_2\\ \text{-}\,\tilde{2}x_1 + \tilde{4}x_2\end{pmatrix} \quad (7.4.6)$$

$$\text{s.t.}\quad \begin{cases}\tilde{a}_{11}x_1 + \tilde{a}_{12}x_2 = \text{-}\,\tilde{1}x_1 + \tilde{3}x_2 \precsim \tilde{b}_1 = \tilde{21}\\ \tilde{a}_{21}x_1 + \tilde{a}_{22}x_2 = \tilde{1}x_1 + \tilde{3}x_2 \precsim \tilde{b}_2 = \tilde{27}\\ \tilde{a}_{31}x_1 + \tilde{a}_{32}x_2 = \tilde{4}x_1 + \tilde{3}x_2 \precsim \tilde{b}_3 = \tilde{45}\\ \tilde{a}_{41}x_1 + \tilde{a}_{42}x_2 = \tilde{3}x_1 + \tilde{1}x_2 \precsim \tilde{b}_4 = \tilde{30}\\ x_1 \ge 0;\quad x_2 \ge 0\end{cases}$$

The membership functions of fuzzy parameters of the objective functions and constraints are set up as follows:

$$\mu_{\tilde{c}_{11}}(x) = \begin{cases}0 & x<3 \text{ or } 6<x\\ (x^2-9)/7 & 3\le x<4\\ 1 & x=4\\ (36-x^2)/20 & 4<x\le 6\end{cases}\qquad \mu_{\tilde{c}_{12}}(x) = \begin{cases}0 & x<1 \text{ or } 4<x\\ (x^2-1)/3 & 1\le x<2\\ 1 & x=2\\ (16-x^2)/12 & 2<x\le 4\end{cases}$$

$$\mu_{\tilde{c}_{21}}(x) = \begin{cases}0 & x<-2.5 \text{ or } -1<x\\ (6.25-x^2)/2.25 & -2.5\le x<-2\\ 1 & x=-2\\ (x^2-1)/3 & -2<x\le -1\end{cases}\qquad \mu_{\tilde{c}_{22}}(x) = \begin{cases}0 & x<3 \text{ or } 6<x\\ (x^2-9)/7 & 3\le x<4\\ 1 & x=4\\ (36-x^2)/20 & 4<x\le 6\end{cases}$$

$$\mu_{\tilde{a}_{11}}(x) = \begin{cases}0 & x<-2 \text{ or } -0.5<x\\ (4-x^2)/3 & -2\le x<-1\\ 1 & x=-1\\ (x^2-0.25)/0.75 & -1<x\le -0.5\end{cases}\qquad \mu_{\tilde{a}_{12}}(x) = \begin{cases}0 & x<2 \text{ or } 5<x\\ (x^2-4)/5 & 2\le x<3\\ 1 & x=3\\ (25-x^2)/16 & 3<x\le 5\end{cases}$$

$$\mu_{\tilde{a}_{21}}(x) = \begin{cases}0 & x<0.5 \text{ or } 2<x\\ (x^2-0.25)/0.75 & 0.5\le x<1\\ 1 & x=1\\ (4-x^2)/3 & 1<x\le 2\end{cases}\qquad \mu_{\tilde{a}_{22}}(x) = \begin{cases}0 & x<2 \text{ or } 5<x\\ (x^2-4)/5 & 2\le x<3\\ 1 & x=3\\ (25-x^2)/16 & 3<x\le 5\end{cases}$$

$$\mu_{\tilde{c}_{31}}(x) = \begin{cases}0 & x<3 \text{ or } 6<x\\ (x^2-9)/7 & 3\le x<4\\ 1 & x=4\\ (36-x^2)/20 & 4<x\le 6\end{cases}\qquad \mu_{\tilde{a}_{32}}(x) = \begin{cases}0 & x<2 \text{ or } 5<x\\ (x^2-4)/5 & 2\le x<3\\ 1 & x=3\\ (25-x^2)/16 & 3<x\le 5\end{cases}$$

$$\mu_{\tilde{a}_{41}}(x)=\begin{cases}0 & x<2 \text{ or } 5<x\\ (x^2-4)/5 & 2\le x<3\\ 1 & x=3\\ (25-x^2)/16 & 3<x\le 5\end{cases}\qquad \mu_{\tilde{a}_{42}}(x)=\begin{cases}0 & x<0.5 \text{ or } 2<x\\ (x^2-0.25)/0.75 & 0.5\le x<1\\ 1 & x=1\\ (4-x^2)/3 & 1<x\le 2\end{cases}$$

$$\mu_{\tilde{b}_1}(x)=\begin{cases}0 & x<20 \text{ or } 23<x\\ (x^2-400)/41 & 20\le x<21\\ 1 & x=21\\ (529-x^3)/88 & 21<x\le 23\end{cases}\qquad \mu_{\tilde{b}_2}(x)=\begin{cases}0 & x<26 \text{ or } 29<x\\ (x^2-676)/53 & 26\le x<27\\ 1 & x=27\\ (841-x^2)/112 & 27<x\le 29\end{cases}$$

$$\mu_{\tilde{b}_3}(x)=\begin{cases}0 & x<44 \text{ or } 47<x\\ (x^2-1936)/89 & 44\le x<45\\ 1 & x=.45\\ (2209-x^2)/184 & 45<x\le 47\end{cases}\qquad \mu_{\tilde{b}_4}(x)=\begin{cases}0 & x<29 \text{ or } 32<x\\ (x^2-841)/59 & 29\le x<30\\ 1 & x=.30\\ (1024-x^2)/124 & 30<x\le 32\end{cases}$$

Stage 1: Initialisation

Step 1: Input membership functions of $\tilde{c}$ for objective functions $\tilde{f}(x)=\tilde{c}x$, $\tilde{a}$ and $\tilde{b}$ for constraints $\tilde{a}x \precsim_{\alpha} \tilde{b}$. For example, the membership function of $\tilde{c}_{11}$ is given as shown in Fig. 7.8. We set an initial satisfactory degree α as 0.2. We use default values for the weights of objective functions.

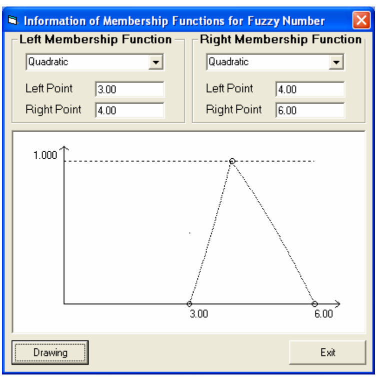


Fig. 7.8: Membership function of a fuzzy parameter $\tilde{c}_{11}$

Steps 2-4: Under the degree $\alpha = 0.2$, we calculate the Pareto optimal solution. Associated with the FMOLP$_\alpha$ problem in the example, a corresponded MOLP$_{\alpha}$ problem is listed:

$$\max \begin{bmatrix} \sqrt{9\lambda+9} & \sqrt{3\lambda+1} \\ \sqrt{36-20\lambda} & \sqrt{16-12\lambda} \\ \sqrt{6.25-2.25\lambda} & \sqrt{9\lambda+9} \\ \sqrt{3\lambda+1} & \sqrt{36-20\lambda} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \tag{7.4.7}$$

st.

$$\begin{bmatrix} \sqrt{4-3\lambda} & \sqrt{5\lambda+4} \\ \sqrt{0.75\lambda+0.25} & \sqrt{25-16\lambda} \\ \sqrt{0.75\lambda+0.25} & \sqrt{5\lambda+4} \\ \sqrt{4-3\lambda} & \sqrt{25-16\lambda} \\ \sqrt{9\lambda+9} & \sqrt{5\lambda+4} \\ \sqrt{36-20\lambda} & \sqrt{25-16\lambda} \\ \sqrt{5\lambda+4} & \sqrt{0.75\lambda+0.25} \\ \sqrt{25-16\lambda} & \sqrt{4-3\lambda} \end{bmatrix} \begin{bmatrix} x_1 \\ x_1 \end{bmatrix} \leq \begin{bmatrix} \sqrt{41\lambda+400} \\ \sqrt{529-88\lambda} \\ \sqrt{53\lambda+676} \\ \sqrt{841-112\lambda} \\ \sqrt{89\lambda+1936} \\ \sqrt{2209-184\lambda} \\ \sqrt{59\lambda+841} \\ \sqrt{1024-124\lambda} \end{bmatrix},$$

where $\forall \lambda \in [\alpha, 1]$.

Refer to the MOLP$_{\alpha\lambda}$ problem, initially, $\lambda_0 = 0.2$ and $\lambda_1 = 1$, then totally, 8 non-fuzzy objective functions and 16 non-fuzzy constraints are generated. The result is listed as follows:

$$\max \begin{bmatrix} \sqrt{10.8} & \sqrt{1.6} \\ \sqrt{18} & 2 \\ \sqrt{32} & \sqrt{13.6} \\ 4 & 2 \\ \sqrt{5.8} & \sqrt{10.8} \\ 2 & \sqrt{18} \\ \sqrt{1.6} & \sqrt{32} \\ 2 & 4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \tag{7.4.8}$$

s.t.

$$\begin{bmatrix} \sqrt{3.4} & \sqrt{5} \\ 1 & 3 \\ \sqrt{0.4} & \sqrt{21.8} \\ 1 & 3 \\ \sqrt{0.4} & \sqrt{5} \\ 1 & 3 \\ \sqrt{3.4} & \sqrt{21.8} \\ 1 & 3 \\ \sqrt{10.8} & \sqrt{5} \\ \sqrt{18} & 3 \\ \sqrt{32} & \sqrt{21.8} \\ 4 & 3 \\ \sqrt{5} & \sqrt{0.4} \\ 3 & 1 \\ \sqrt{21.8} & \sqrt{3.4} \\ 3 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_1 \end{bmatrix} \le \begin{bmatrix} \sqrt{408.2} \\ 21 \\ \sqrt{501.6} \\ 21 \\ \sqrt{686.6} \\ 27 \\ \sqrt{818.6} \\ 27 \\ \sqrt{1953.8} \\ 45 \\ \sqrt{2245.8} \\ 45 \\ \sqrt{852.8} \\ 30 \\ \sqrt{999.2} \\ 30 \end{bmatrix}.$$

The interval [α, 1] is further split. We then have

$$x_1^* = 1.9115 \ , \ x_2^* = 5.1023 \ ,$$

and two optimal objective values (see Fig. 7.9)

$$\tilde{f}_1^*\left(x_1^*, x_2^*\right) = \tilde{f}_1^*(1.9115{,}5.1023) = 1.9115\tilde{c}_{11} + 5.1023\tilde{c}_{12}\,,$$
$$\tilde{f}_2^*\left(x_1^*, x_2^*\right) = \tilde{f}_2^*(1.9115{,}5.1023) = 1.9115\tilde{c}_{21} + 5.1023\tilde{c}_{22}\,.$$

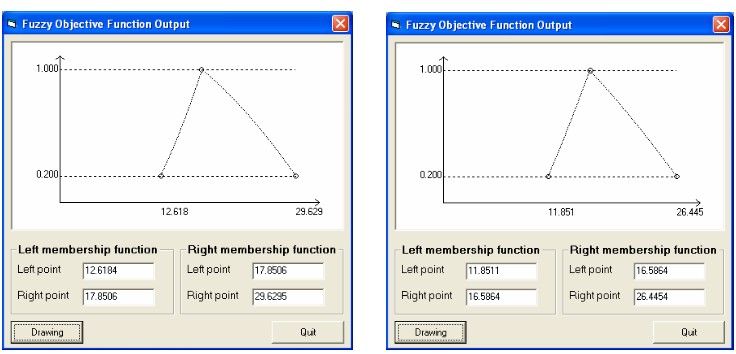


Fig. 7.9: Membership functions for $\tilde{f}_1^*\left(x_1^*, x_2^*\right)$ and $\tilde{f}_2^*\left(x_1^*, x_2^*\right)$ in Stage 1

Steps 5-6: Suppose decision makers are not satisfied with the initial Pareto optimal solution, the interactive process will start.

Stage 2: Iterations

Iteration No. 1:

Step 7: Based on the Pareto optimal solution obtained in Stage 1, decision makers specify new fuzzy goals $(\tilde{g}_1, \tilde{g}_2)$ by increasing 30% on the first objective function $\tilde{f}_1^*(x_1^*, x_2^*) = \tilde{f}_1^*(1.9115, 5.1023)$ and decreasing 25% on the second one $\tilde{f}_2^*(x_1^*, x_2^*) = \tilde{f}_2^*(1.9115, 5.1023)$, that is,

$$(\tilde{g}_1, \tilde{g}_2) = \left(1.3 * \tilde{f}_1^*(x_1^*, x_2^*),\ 0.75 * \tilde{f}_2^*(x_1^*, x_2^*)\right). \tag{7.4.9}$$

Steps 8-10: Calculate the fuzzy Pareto optimal solution based on the new fuzzy goals $(\tilde{g}_1, \tilde{g}_2)$ and the satisfactory degree $\alpha = 0.2$.

Under the new fuzzy goals, the FMOLP$_\alpha$ problem is converted into a non-fuzzy MOLP$_{\alpha\lambda m}$ problem as follows:

$$\min\ \max \begin{bmatrix} \sqrt{9\lambda+9} & \sqrt{3\lambda+1} \\ \sqrt{36-20\lambda} & \sqrt{16-12\lambda} \\ \sqrt{6.25-2.25\lambda} & \sqrt{9\lambda+9} \\ \sqrt{3\lambda+1} & \sqrt{36-20\lambda} \end{bmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} - \begin{pmatrix} 2.485\sqrt{9\lambda+9} + 6.633\sqrt{3\lambda+1} \\ 2.485\sqrt{36-20\lambda} + 6.633\sqrt{16-12\lambda} \\ 1.529\sqrt{6.25-2.25\lambda} + 4.082\sqrt{9\lambda+9} \\ 1.529\sqrt{3\lambda+1} + 4.082\sqrt{36-20\lambda} \end{pmatrix} \tag{7.4.10}$$

$$\text{s.t.} \begin{bmatrix} \sqrt{4-3\lambda} & \sqrt{5\lambda+4} \\ \sqrt{0.75\lambda+0.25} & \sqrt{25-16\lambda} \\ \sqrt{0.75\lambda+0.25} & \sqrt{5\lambda+4} \\ \sqrt{4-3\lambda} & \sqrt{25-16\lambda} \\ \sqrt{9\lambda+9} & \sqrt{5\lambda+4} \\ \sqrt{36-20\lambda} & \sqrt{25-16\lambda} \\ \sqrt{5\lambda+4} & \sqrt{0.75\lambda+0.25} \\ \sqrt{25-16\lambda} & \sqrt{4-3\lambda} \end{bmatrix} \begin{bmatrix} x_1 \\ x_1 \end{bmatrix} \le \begin{bmatrix} \sqrt{41\lambda+400} \\ \sqrt{529-88\lambda} \\ \sqrt{53\lambda+676} \\ \sqrt{841-112\lambda} \\ \sqrt{89\lambda+1936} \\ \sqrt{2209-184\lambda} \\ \sqrt{59\lambda+841} \\ \sqrt{1024-124\lambda} \end{bmatrix},$$

where $\forall\lambda \in [\alpha, 1]$.

We obtain

$$x_1^* = 3.0486\ ,\ x_2^* = 4.9239\ ,$$

and two optimal fuzzy objective values are

$$\tilde{f}_1^*\left(x_1^*, x_2^*\right) = \tilde{f}_1^*(3.0486, 4.9239) = 3.0486\tilde{c}_{11} + 4.9239\tilde{c}_{12},$$
$$\tilde{f}_2^*\left(x_1^*, x_2^*\right) = \tilde{f}_2^*(3.0486, 4.9239) = 3.0486\tilde{c}_{21} + 4.9239\tilde{c}_{22},$$

as shown in Fig. 7.10.

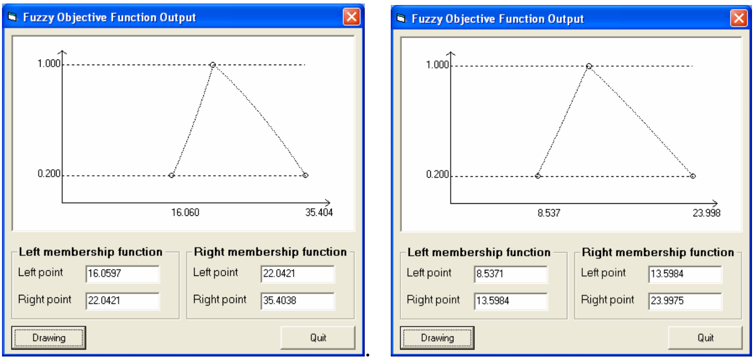


Fig. 7.10: Membership functions for $\tilde{f}_1^*\left(x_1^*, x_2^*\right)$ and $\tilde{f}_2^*\left(x_1^*, x_2^*\right)$ in Iteration No. 1

Step 11: Suppose decision makers do not satisfy the fuzzy Pareto optimal solution, the interactive process will proceed, that is, starting the second iteration.

Iteration No. 2:

Step 7: At this iteration, suppose decision makers specify new fuzzy goals $(\tilde{g}_1, \tilde{g}_2)$ by the corresponding membership functions as follows (see Fig. 7.11):

$$\mu_{\tilde{g}_1}(x) = \begin{cases} 0 & x < 14 \text{ or } 37 < x \\ \left(x^2 - 196\right)/245 & 14 \le x < 21 \\ 1 & x = 21 \\ \left(1369 - x^2\right)/928 & 21 < x \le 37 \end{cases}, \qquad (7.4.11)$$

$$\mu_{\tilde{g}_2}(x) = \begin{cases} 0 & x < 6.5 \text{ or } 25 < x \\ \left(x^2 - 42.25\right)/114 & 6.5 \le x < 12.5 \\ 1 & x = 12.5 \\ \left(625 - x^2\right)/468.75 & 12.5 < x \le 25 \end{cases}. \qquad (7.4.12)$$

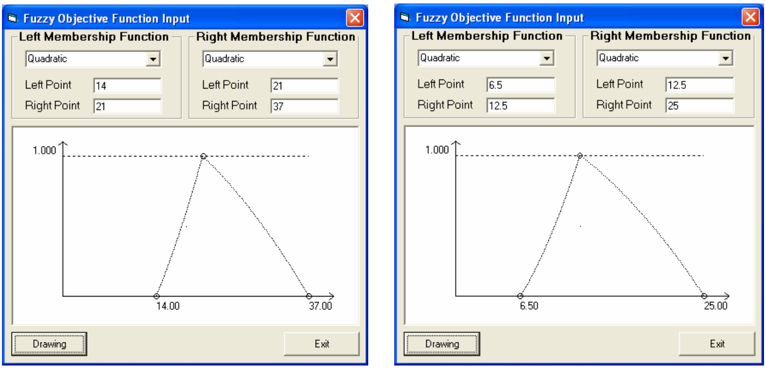


Fig. 7.11: Membership functions of the new fuzzy goals in Iteration No. 2

Steps 8-10: Calculate the fuzzy Pareto optimal solution based on the new fuzzy goals $(\tilde{g}_1, \tilde{g}_2)$ and keep the degree $\alpha = 0.2$.

Under the fuzzy goals, the FMOLP$_\alpha$ problem is converted into the non-fuzzy MOLP$_{\alpha\lambda m}$ problem as follows:

$$\min\ \max \begin{bmatrix} \sqrt{9\lambda+9} & \sqrt{3\lambda+1} \\ \sqrt{36-20\lambda} & \sqrt{16-12\lambda} \\ \sqrt{6.25-2.25\lambda} & \sqrt{9\lambda+9} \\ \sqrt{3\lambda+1} & \sqrt{36-20\lambda} \end{bmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} - \begin{pmatrix} \sqrt{245\lambda+196} \\ \sqrt{1369-928\lambda} \\ \sqrt{114\lambda+42.25} \\ \sqrt{625-468.75\lambda} \end{pmatrix} \tag{7.4.13}$$

s.t.

$$\begin{bmatrix} \sqrt{4-3\lambda} & \sqrt{5\lambda+4} \\ \sqrt{0.75\lambda+0.25} & \sqrt{25-16\lambda} \\ \sqrt{0.75\lambda+0.25} & \sqrt{5\lambda+4} \\ \sqrt{4-3\lambda} & \sqrt{25-16\lambda} \\ \sqrt{9\lambda+9} & \sqrt{5\lambda+4} \\ \sqrt{36-20\lambda} & \sqrt{25-16\lambda} \\ \sqrt{5\lambda+4} & \sqrt{0.75\lambda+0.25} \\ \sqrt{25-16\lambda} & \sqrt{4-3\lambda} \end{bmatrix} \begin{bmatrix} x_1 \\ x_1 \end{bmatrix} \le \begin{bmatrix} \sqrt{41\lambda+400} \\ \sqrt{529-88\lambda} \\ \sqrt{53\lambda+676} \\ \sqrt{841-112\lambda} \\ \sqrt{89\lambda+1936} \\ \sqrt{2209-184\lambda} \\ \sqrt{59\lambda+841} \\ \sqrt{1024-124\lambda} \end{bmatrix},$$

where $\forall \lambda \in [\alpha, 1]$.

We have

$$x_1^* = 2.8992\ ,\ x_2^* = 4.9829\ ,$$

and two optimal objective values are (see Fig. 7.12)

$$\tilde{f}_1^*(x_1^*, x_2^*) = \tilde{f}_1^*(2.8992, 4.9829) = 2.8992\tilde{c}_{11} + 4.9829\tilde{c}_{12},$$
$$\tilde{f}_2^*(x_1^*, x_2^*) = \tilde{f}_2^*(2.8992, 4.9829) = 2.8992\tilde{c}_{21} + 4.9829\tilde{c}_{22}.$$

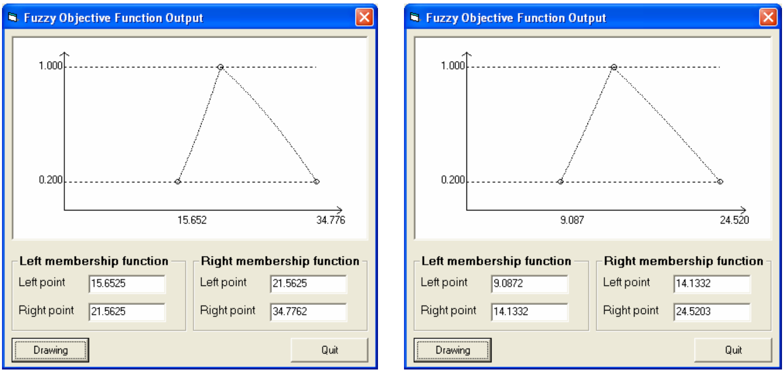


Fig. 7.12: Membership functions for $\tilde{f}_1^*(x_1^*, x_2^*)$ and $\tilde{f}_2^*(x_1^*, x_2^*)$ in Iteration No. 3

Step 11: Now decision makers are satisfied with the solution obtained in Step 10, the interactive process thus terminates. The final solution of the FMOLP problem is, $x_1^* = 2.8992$, , $x_2^* = 4.9829$, the first objective's value is around 21.5625, and the second's is around 14.1332.

$$\begin{cases} x_1^* = 2.8992 \\ x_2^* = 4.9829 \end{cases}, \tag{7.4.14}$$

$$\begin{cases} \tilde{f}_1^*(x_1^*, x_2^*) = \tilde{f}_1^*(2.8992, 4.9829) = 2.8992\tilde{c}_{11} + 4.9829\tilde{c}_{12} \\ \tilde{f}_2^*(x_1^*, x_2^*) = \tilde{f}_2^*(2.8992, 4.9829) = 2.8992\tilde{c}_{21} + 4.9829\tilde{c}_{22} \end{cases}. \tag{7.4.15}$$

7.5 Summary

The developed FMODM methods extend MODM decision analysis functions from crisp to imprecise scope. This chapter gives three methods to solve the FMOLP problems. Several points are indicated here to help readers effectively use these methods.

- These three methods deal with a general FMOLP problem with fuzzy parameters appearing in either objective functions or constraints or both. They are still applicable to deal with non-fuzzy parameters as a real number is as a special case of a fuzzy number. Similarly, a goal with a real number is also as a special case of a fuzzy goal.

- These three methods all allow decision makers to use any form of membership functions for describing fuzzy parameters in objective functions or constraints, and also for expressing decision makers' fuzzy goals. When decision makers do not have a clear idea to choose a suitable form of membership functions, they can try different forms or use a default form provided by the FMODSS software. This feature offers decision makers a higher confidence in using the methods to solve practical problems.
- Obviously, some FMOLP methods are more suitable than others for some particular decision makers in some particular decision problems. For example, managers might have enough expertise knowledge of FMOLP models and their fuzzy goals for objectives in an FMOLP problem. Particularly, they prefer to explore possible optimal solutions through monitoring their fuzzy goals. In such a case, the IFMOLP method is the most suitable one for them. Other decision makers who have expertise of the FMOLP model but have no idea in giving goals for objective functions in their decision problems, the FMOLP method will be the best.
- However, the selection of the most suitable one from a number of available FMODM methods is difficult to accomplish by general decision makers, because it needs some expertise and experience to understand specific features of these methods. Table 7.3 shows the main characteristics of the three methods in order to advise users choosing a suitable one for a particular decision.

Table 7.3: Main characteristics of the three methods

Methods / *Char.*	*Scalarisation*	*Fuzzy goal*	*Interaction*
Degree α	*	*	*
Weight	*	*	*
Fuzzy goal		*	*
Revising goal			*

Chapter 8

Fuzzy Multi-Objective DSS

We now present a fuzzy multi-objective DSS that implements the three methods proposed in Chapter 7 for solving fuzzy multi-objective decision problems. We first describe the configuration, the interface, the model-base, and the method-base of the system. We then give two case-based examples to demonstrate the FMOLP problem solving procedure.

8.1 System Configuration

As a specific type of DSS, a fuzzy multi-objective DSS (FMODSS) aims to help decision makers gather the knowledge about the FMOLP problem itself so as to make a better-informed decision, and encourage decision makers to explore the support tools in an iterative fashion for further defining and refining the nature of the problem.

With the aid of the FMODSS, decision makers are able to fully control the decision making process and can obtain possible solutions to their problems. The friendly windows-based user interface of this system enables decision makers to take advantage of the capabilities of the system in making real-time decisions.

The user interface of the FMODSS has the typical form of window-based software. It takes advantage of the graphical capabilities of Windows environment enabling users (decision makers or decision analysts) to exploit fully the capabilities of the system.

The FMODSS consists of four major software components: (1) input-and-display component, (2) model management component, (3) method management component, and (4) data management component. It also has three bases: (a) database, (b) FMOLP method-base, and (c) model-base. These bases are linked to their corresponding management components respectively. Fig. 8.1 shows the structure of the FMODSS.

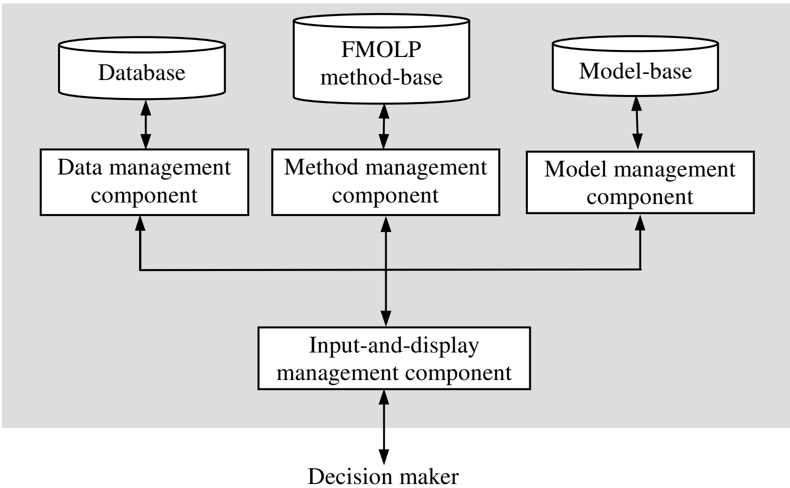


Fig. 8.1: The structure of the FMODSS

During the decision making process, the system needs decision makers' inputs to the FMOLP model and the selected decision method, and interprets output from the method continually throughout the interactive process. Thus, these inputs and outputs must be formatted in such a way that they are intuitive and easy for decision makers to use. Following the FMOLP model and its input-and-display component, some typical data, such as fuzzy parameters of the FMOLP model, weights, and satisfactory degrees *etc*., need to be input from users for setting up models and other initial data for the system.

From Fig. 8.1, all data within the system, such as parameters, alternative definitions and values, intermediate and/or final results, even the data from the external sources, will be stored in the database by the data management component.

Importantly, the model management component is functionally able to define and structure a fuzzy multi-objective decision problem, and generate a decision making model based on data inputs. Generally, it is combined with the data management component and provides facilities for the definition, storage, retrieval and execution of a wide range of models. It also gives decision makers to specify or build entirely new models by using a model-building facility associated with an input-and-display component.

Depending on the nature of decision makers to an FMOLP problem, different methods are thus contained in the method-base for the method management component to access for searching the optimal solution. Decision makers can select the most suitable method for solving their decision problems. To harness the potential of these methods effectively, the system is flexible enough to let new or revised methods be introduced if desired.

8.2 System Interface

An FMODSS is designed and developed as a prototype essentially applied for solving FMOLP problems. It involves different kinds of interfaces such as windows, menus, dialog boxes, icons, and forms that are able to assist decision makers for modelling, understanding, analysing, and solving their problems. There are five menus that form the functions of the system interface. They are *File* menu, *Method* menu, *Model* menu, *Result* menu, and *Help* menu. These pull-down menus together with their respective windows perform all kinds of decision support activities.

Among five items in the File menu (see Fig. 8.2), New FMOLP Model, Open FMOLP Model, and Save FMOLP Model are for dealing with FMOLP models. Three items, Fuzzy Multi-Objective Linear Programming (FMOLP), Fuzzy Multi-Objective Linear Goal Programming (FMOLGP), and Interactive Fuzzy Multi-Objective Linear Programming (IFMOLP), are included in the *Method* menu. The Model menu is used for displaying the current model that will be solved in the latter procedure by using some suitable methods. One item, which is FMOLP Model, is in the Model menu. Similarly, the item, FMOLP Result, is included in the Result menu for showing the optimal solution for the current FMOLP problem.

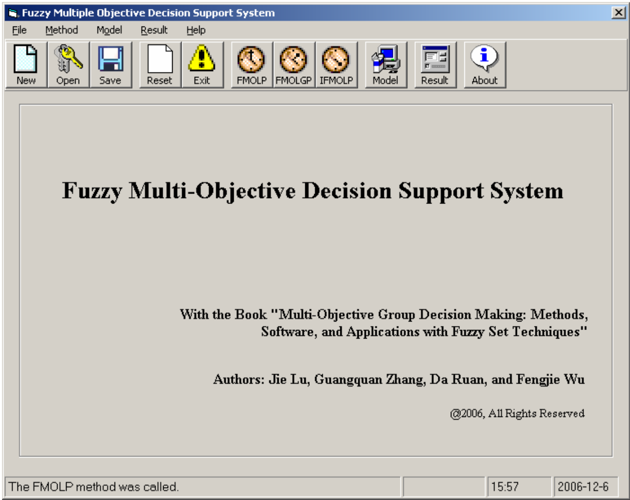


Fig. 8.2: Main interface of the FMODSS

8.3 A Model-Base and Model Management

A model-base is set up for storing users' application models in the FMODSS. Each model is prepared in a file format. And these models in the model-base are connected with the database and data management component for retrieving and storing the related modelling data of the problems.

A model management component combined with the model-base defines, develops, and maintains decision models for computing efficient solutions. This component inputs a new model, opens an existing model stored in the model-base, or stores the current model to the model-base for the further use or modification. Generally, the model management component is connected with the database and the data management component.

By clicking the item of *New FMOLP Model* in the *File* menu, we can start a procedure for setting up a new model for an FMOLP problem. Based on the FMOLP model described in Chapter 6, the following common data are needed for creating the model.

- The numbers of decision variables, fuzzy objective functions, and fuzzy constraints, respectively.
- The names of decision variables, fuzzy objective functions, and fuzzy constraints, respectively.
- The parameters of fuzzy objective functions, the max/min for individual fuzzy objective function as shown in Fig. 8.3.
- The parameters of fuzzy constraints and the relation signs of individual fuzzy constraint as shown in Fig. 8.3.

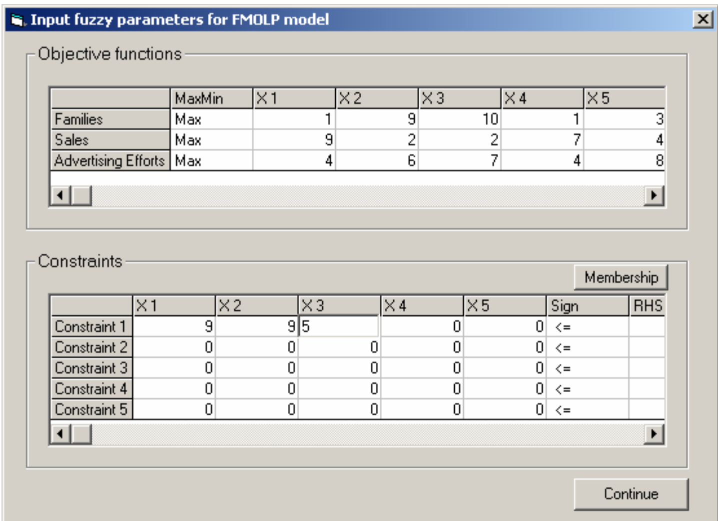


Fig. 8.3: Input fuzzy objective functions and fuzzy constraints

As the parameters of fuzzy objective functions and fuzzy constraints and fuzzy goals are represented by fuzzy numbers, a Dialog Box as showed in Fig. 8.4 is designed specially for entering these fuzzy numbers. Referring to a fuzzy number to be entered, the forms of left continuous increasing function and right continuous decreasing function of a fuzzy number can be selected as linear, quadratic, cubic, exponential, logarithmic, other piecewise forms from the dropdown lists, and four end-points of left and right function of fuzzy numbers are entered in the textboxes simultaneously. Fig. 8.5 shows the general information about an FMOLP problem to be solved.

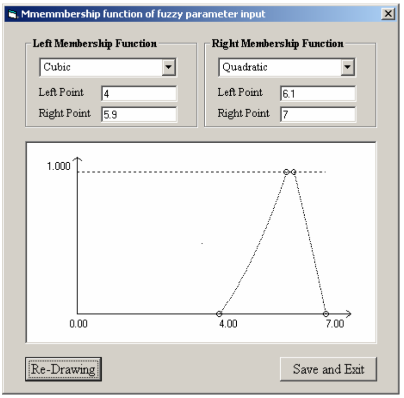


Fig. 8.4: Input membership function of a fuzzy number

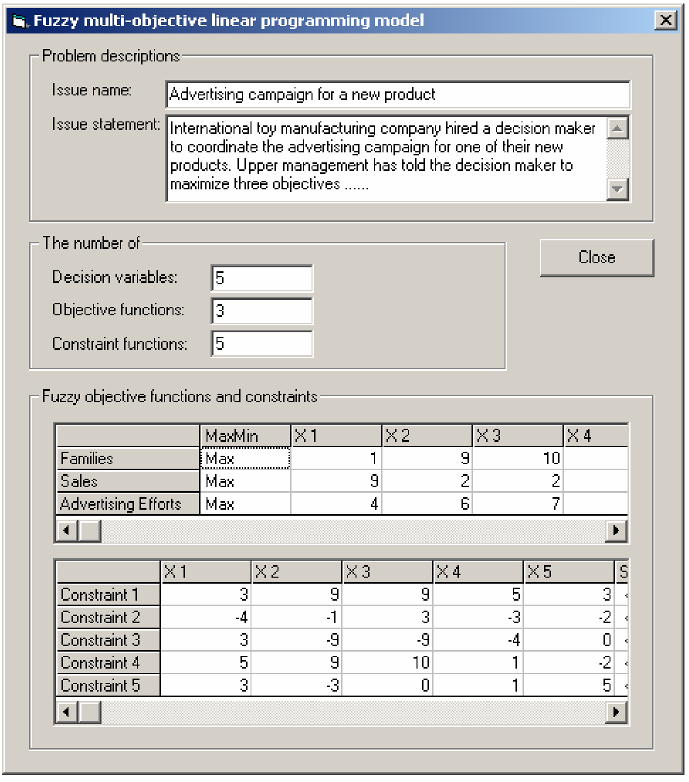


Fig. 8.5: General information about an FMOLP problem

8.4 A Method-Base and Solution Process

Recall the three methods proposed in Chapter 7: FMOLP method, FMOLGP method, and IFMOLP method. They are now implemented and stored in the method-base. Different methods contained in the method-base can be accessed for the method management component and for searching optimal solutions of FMOLP problems.

8.4.1 *Fuzzy MOLP*

By clicking the item of *FMOLP* in the *Method* menu, Fig. 8.6 shows windows in which different weights for fuzzy objective functions can be entered in *FlexGrid 1*, and the degree of all membership functions of the fuzzy numbers can also be set by the slider as well. Currently in the window, the degree is 0.15. When the degree is set to 1, the original fuzzy problem is converted to a crisp problem, and the values of objective functions will be non-fuzzy numbers.

Following the FMOLP method, click Button *Run*, a solution of the problem including decision variables and fuzzy objective functions will be shown in *FlexGrid 2* and *FlexGrid 3*, respectively. Here, the output of decision variables as shown in *FlexGrid 2* is $x_1^* = 58.27$, $x_2^* = 52.56$, $x_3^* = 0.0$, $x_4^* = 4.66$, $x_5^* = 36.86$. To display membership functions of fuzzy objective functions output, click the corresponding grids in *FlexGrid 3* and Button *membership* one by one, new windows will be displayed similarly as Fig. 8.7 sequentially.

8.4.2 *Fuzzy MOLGP*

Similar to the FMOLP method in Section 8.4.1, by clicking the item of *FMOLGP* in the *Method* menu, Fig. 8.8 shows the window, in which the

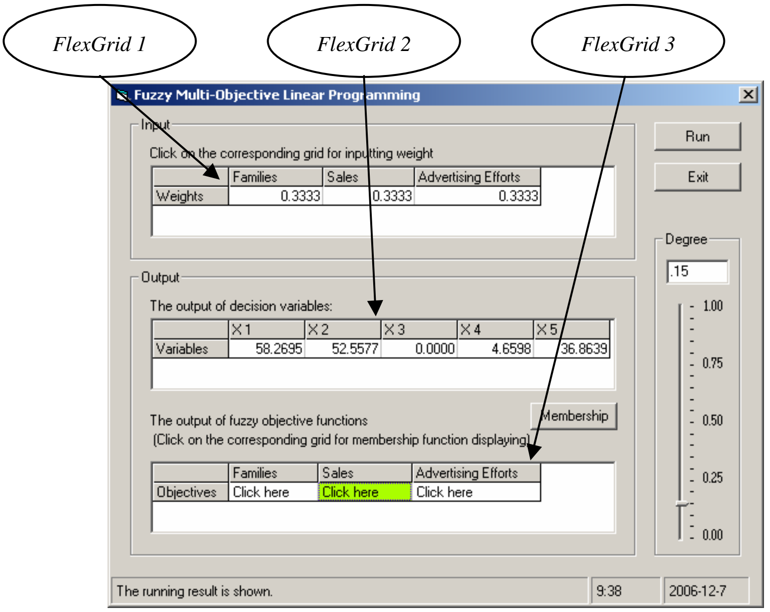


Fig. 8.6: Solving FMOLP problems with the FMOLP method

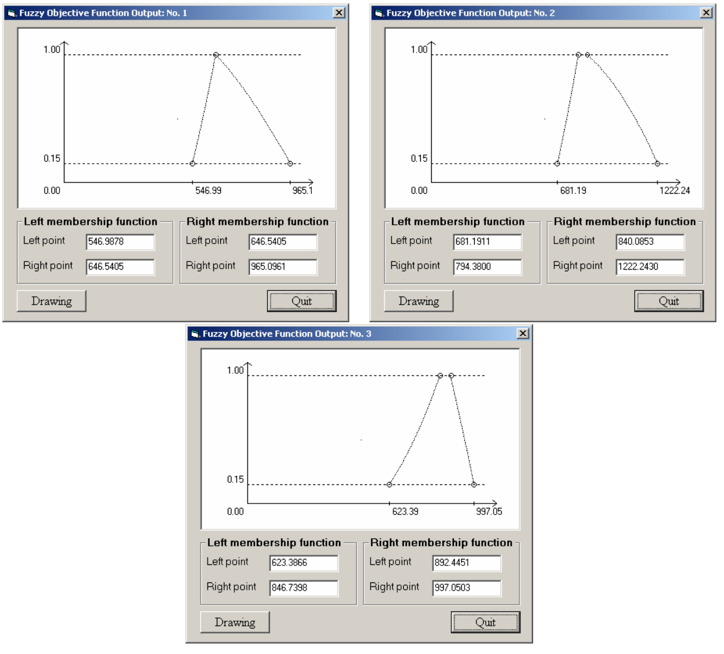


Fig. 8.7: Membership functions of fuzzy objective functions

initial fuzzy goals can be entered in *FlexGrid 2*. To input membership functions of fuzzy goals that are represented by fuzzy numbers, click the corresponding grid in *FlexGrid 2* and Button *Membership*, new windows will be shown in Fig. 8.9.

For example, a fuzzy goal is entered in Fig. 8.9, both of the left and right membership functions of the fuzzy goal are set as quadratic. The four-end points for left and right membership functions are 650, 700, 700, and 740, respectively. The diagram in Fig. 8.9 shows the shape of the membership function for the fuzzy goal.

After having input fuzzy goals and setting the degree α, press Button *Run*, the solution will be supplied in *FlexGrid 3* and *FlexGrid 4*.

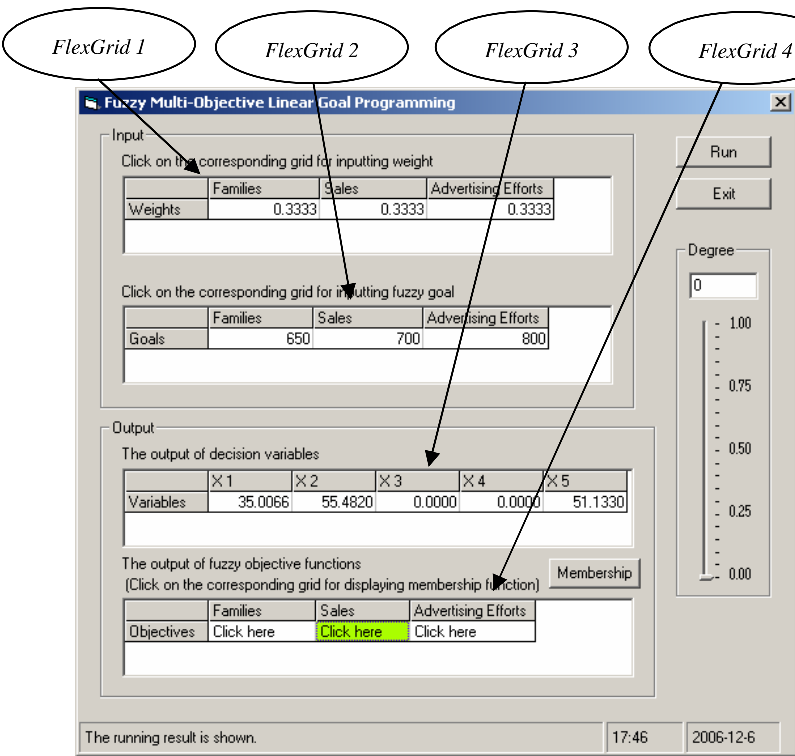


Fig. 8.8: Solving FMOLP problems with the FMOLGP method

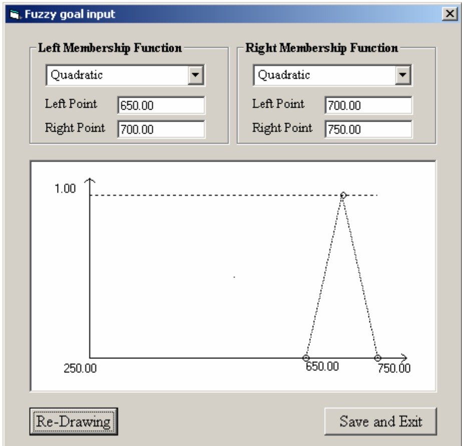


Fig. 8.9: Input a fuzzy goal

8.4.3 *Interactive FMOLP*

Following our discussions in Chapter 7, we just outline the working process of using the IFMOLP method in the FMODSS. The steps here are little different from the steps listed in Section 7.4 as here we work on a software. Original 11 steps are reduced to eight steps here.

Stage 1: Initialisation

This stage is to set up an FMOLP model and generate an initial optimal solution to the model.

Step 1: Set up an FMOLP model and input membership functions of fuzzy parameters of the model.

Step 2: Ask decision makers to select a satisfactory degree α $(0 \le \alpha \le 1)$ and individual weights for fuzzy objective functions.

Step 3: Solve the FMOLP problem under the current degree α and weights.

Step 4: If the Pareto optimal solution including optimal decision variables x^* and fuzzy objective functions $\tilde{f}\left(x^*\right)$ exists in Step 3, go to the next step. Otherwise, go back to Step 2 to reassign the degree α and solve the FMOLP problem again.

Step 5: Ask decision makers whether the initial solution in Step 3 is satisfied. If so, the whole interactive process stops, and the initial solution is to be the final satisfactory solution. Otherwise, go to Stage 2.

Stage 2: Iterations

At this stage, the interactive process will proceed. At each iteration phase, decision makers are supplied with the solution obtained at Initialisation stage or the previous phase. If not satisfied with the current solution, decision makers are asked to specify their fuzzy goals, and then a new compromise solution will be generated until decision makers stop the iterative procedure.

Step 6: Specify new fuzzy goals $\tilde{g} = (\tilde{g}_1, \tilde{g}_2, \ldots, \tilde{g}_k)^T$ for fuzzy objective functions based on the current solution and a new degree α if needed. Decision makers will have to make the compromise among the fuzzy objectives. An improvement for one or more of the fuzzy objectives will result in the sacrifices of other fuzzy objectives.

Step 7: Calculate a compromise solution based on the current fuzzy goals of objective functions specified in Step 6 and the degree α.

Step 8: If decision makers are satisfied with the solution calculated in Step 7, the whole interactive process stops. The current compromise solution is the final satisfactory solution of the FMOLP problem. Otherwise, go back to Step 6 for more iteration.

Figure 8.10 shows the working process of the IFMOLP method.

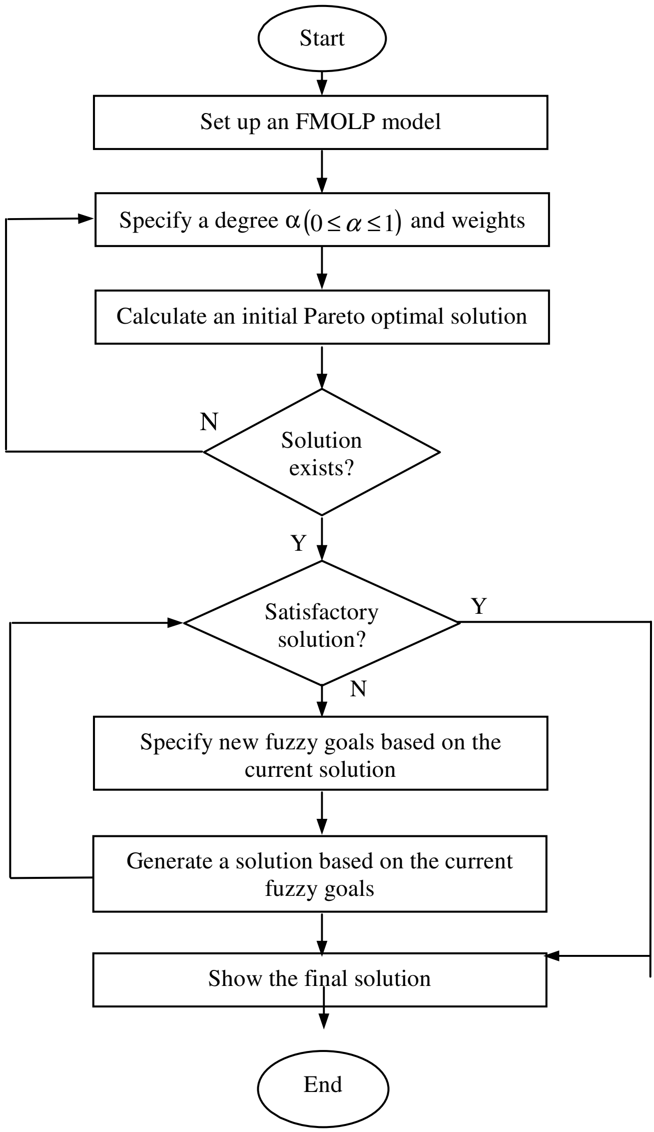


Fig. 8.10: Working process of the IFMOLP method in the FMODSS

In the FMODSS, windows are designed to facilitate decision makers to gather the knowledge about the FMOLP problem to be solved and make a better decision with the IFMOLP method. During the solution process, decision makers can specify fuzzy goals to be achieved by two ways. One is to increase or decrease the previous individual fuzzy objective function solution by percentage in the row '*By %*'. The other is by entering the new fuzzy goals in the row '*By value*'. A new solution at the current trial will be generated. The solution for each trial during the interactive process is recorded and listed in the *historical records* frame.

8.5 Case-Based Examples

To show the programme of the FMODSS, two case-based examples are formulated as FMOLP models and solved in this section.

Example 1: Production planning

As presented in Section 6.1, a manufacturing company has a production planning problem. It has six machine types used to produce three products. Decision makers have three objectives of maximising profits, quality, and worker satisfaction. With the imprecise values listed in Table 6.1, this problem is described by an FMOLP model as follows:

$$\max \begin{pmatrix} \tilde{f}_1(x) \\ \tilde{f}_2(x) \\ \tilde{f}_3(x) \end{pmatrix} = \max \begin{pmatrix} \tilde{50}\, x_1 + \tilde{100}\, x_2 + \tilde{17.5}\, x_3 \\ \tilde{92}\, x_1 + \tilde{75}\, x_2 + \tilde{50}\, x_3 \\ \tilde{25}\, x_1 + \tilde{100}\, x_2 + \tilde{75}\, x_3 \end{pmatrix} \tag{8.5.1}$$

$$\text{s.t.} \begin{cases} \tilde{g}_1(x) = \tilde{12}\, x_1 + \tilde{17}\, x_2 \precsim \tilde{1400} \\ \tilde{g}_2(x) = \tilde{3}\, x_1 + \tilde{9}\, x_2 + \tilde{8}\, x_3 \precsim \tilde{1000} \\ \tilde{g}_3(x) = \tilde{10}\, x_1 + \tilde{13}\, x_2 + \tilde{15}\, x_3 \precsim \tilde{1750} \\ \tilde{g}_4(x) = \tilde{6}\, x_1 + \tilde{16}\, x_3 \precsim \tilde{1325} \\ \tilde{g}_5(x) = \tilde{12}\, x_2 + \tilde{7}\, x_3 \precsim \tilde{900} \\ \tilde{g}_6(x) = \tilde{9.5}\, x_1 + \tilde{9.5}\, x_2 + \tilde{4}\, x_3 \precsim \tilde{1075} \\ x_1, x_2, x_3 \geq 0 \end{cases}$$

In this model, all parameters of objective functions and constraints are represented in triangular fuzzy numbers. The FMOLP model (8.5.1) is built into the system, and the result is shown in Fig. 8.11.

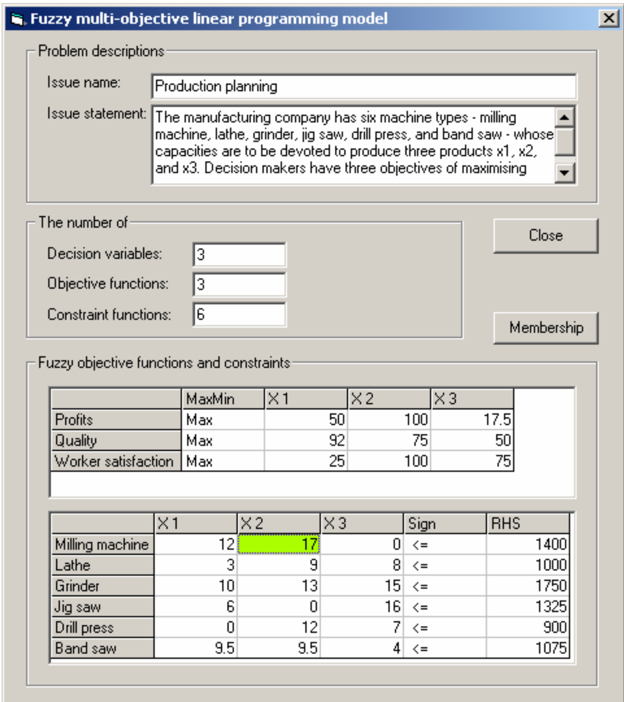


Fig. 8.11: The FMOLP model of Example 1 in the FMODSS

Here, we use the FMOLP method to solve the problem. As shown in Fig. 8.12, the output of decision variables are

$$x_1^* = 68.85, \quad x_2^* = 25.42, \quad x_3^* = 44.68. \tag{8.5.2}$$

and the fuzzy objective functions are

$$\begin{cases} \tilde{f}_1^*\left(x_1^*, x_2^*, x_3^*\right) = 68.85\tilde{c}_{11} + 25.42\tilde{c}_{12} + 44.68\tilde{c}_{13} \\ \tilde{f}_2^*\left(x_1^*, x_2^*, x_3^*\right) = 68.85\tilde{c}_{21} + 25.42\tilde{c}_{22} + 44.68\tilde{c}_{23} \\ \tilde{f}_2^*\left(x_1^*, x_2^*, x_3^*\right) = 68.85\tilde{c}_{31} + 25.42\tilde{c}_{32} + 44.68\tilde{c}_{33} \end{cases} . \tag{8.5.3}$$

The membership functions of $\tilde{f}_1^*$, $\tilde{f}_2^*$, and $\tilde{f}_3^*$ in (8.5.3) are shown in Fig. 8.13, respectively.

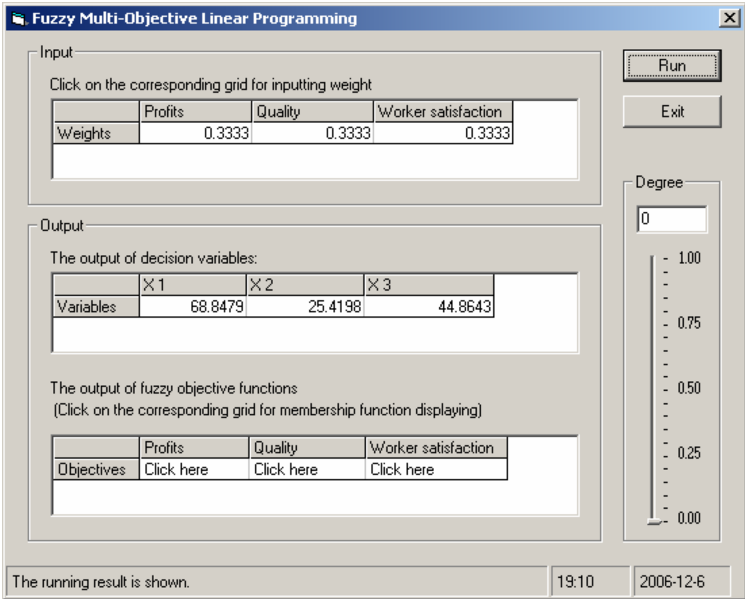


Fig. 8.12: Solving the FMOLP problem (Example 1) by the FMOLP method

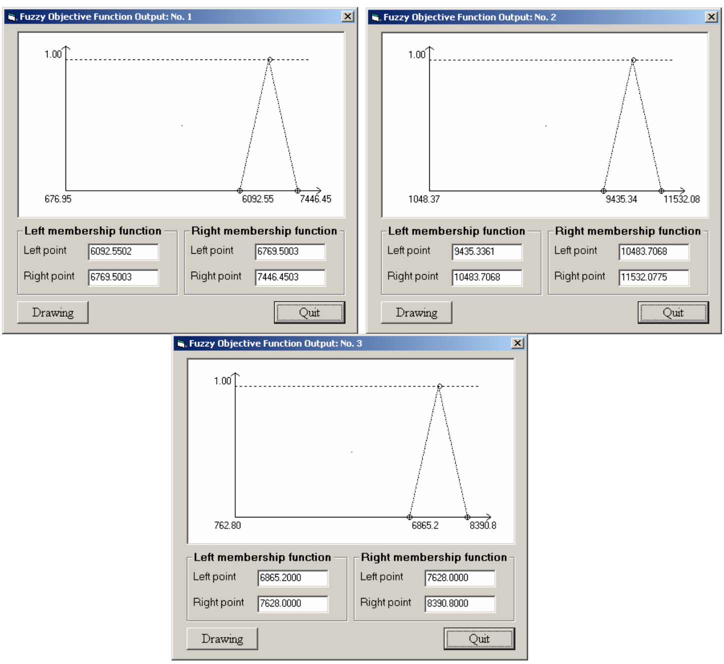


Fig. 8.13: Membership functions of $\tilde{f}_1^*$, $\tilde{f}_2^*$, and $\tilde{f}_3^*$ in Example 1

Example 2: Marketing decision

A marketing decision problem in an international toy manufacturing company is reformulated as an FMOLP model (8.5.4). The FMOLP model consists of simultaneous maximisation of three fuzzy objective functions subjective to five fuzzy constraints involving five decision variables. Three objectives are determined as follows:

- Potential purchase families reached;
- Potential unit sales;
- Benefit/cost of advertising efforts.

The FMOLP problem is modelled as follows.

$$\text{Max}\begin{pmatrix} \tilde{f}_1(x) \\ \tilde{f}_2(x) \\ \tilde{f}_3(x) \end{pmatrix} = \text{Max}\begin{pmatrix} \tilde{c}_{11}x_1 + \tilde{c}_{12}x_2 + \tilde{c}_{13}x_3 + \tilde{c}_{14}x_4 + \tilde{c}_{15}x_5 \\ \tilde{c}_{21}x_1 + \tilde{c}_{22}x_2 + \tilde{c}_{23}x_3 + \tilde{c}_{24}x_4 + \tilde{c}_{25}x_5 \\ \tilde{c}_{31}x_1 + \tilde{c}_{32}x_2 + \tilde{c}_{33}x_3 + \tilde{c}_{44}x_4 + \tilde{c}_{55}x_5 \end{pmatrix} \tag{8.5.4}$$

$$= \text{Max}\begin{pmatrix} \tilde{1}x_1 + \tilde{9}x_2 + \widetilde{10}x_3 + \tilde{1}x_4 + \tilde{3}x_5 \\ \tilde{9}x_1 + \tilde{2}x_2 + \tilde{2}x_3 + \tilde{7}x_4 + \tilde{4}x_5 \\ \tilde{4}x_1 + \tilde{6}x_2 + \tilde{7}x_3 + \tilde{4}x_4 + \tilde{8}x_5 \end{pmatrix}$$

$$\text{s.t.}\begin{cases} \tilde{a}_{11}x_1 + \tilde{a}_{12}x_2 + \tilde{a}_{13}x_3 + \tilde{a}_{14}x_4 + \tilde{a}_{15}x_5 = \tilde{3}x_1 + \tilde{9}x_2 + \tilde{9}x_3 + \tilde{5}x_4 + \tilde{3}x_5 \preceq \tilde{b}_1 = \widetilde{1039} \\ \tilde{a}_{21}x_1 + \tilde{a}_{22}x_2 + \tilde{a}_{23}x_3 + \tilde{a}_{24}x_4 + \tilde{a}_{25}x_5 = -\tilde{4}x_1 - \tilde{1}x_2 + \tilde{3}x_3 - \tilde{3}x_4 - \tilde{2}x_5 \preceq \tilde{b}_2 = \widetilde{94} \\ \tilde{a}_{31}x_1 + \tilde{a}_{32}x_2 + \tilde{a}_{33}x_3 + \tilde{a}_{34}x_4 + \tilde{a}_{35}x_5 = \tilde{3}x_1 - \tilde{9}x_2 - \tilde{9}x_3 - \tilde{4}x_4 - \tilde{0}x_5 \preceq \tilde{b}_3 = \widetilde{61} \\ \tilde{a}_{41}x_1 + \tilde{a}_{42}x_2 + \tilde{a}_{43}x_3 + \tilde{a}_{44}x_4 + \tilde{a}_{45}x_5 = \tilde{5}x_1 + \tilde{9}x_2 + \widetilde{10}x_3 - \tilde{1}x_4 - \tilde{2}x_5 \preceq \tilde{b}_4 = \widetilde{924} \\ \tilde{a}_{51}x_1 + \tilde{a}_{52}x_2 + \tilde{a}_{53}x_3 + \tilde{a}_{54}x_4 + \tilde{a}_{55}x_5 = \tilde{3}x_1 - \tilde{3}x_2 + \tilde{0}x_3 + \tilde{1}x_4 + \tilde{5}x_5 \preceq \tilde{b}_5 = \widetilde{420} \\ x_1 \geq 0;\quad x_2 \geq 0;\quad x_3 \geq 0;\quad x_4 \geq 0;\quad x_5 \geq 0 \end{cases}$$

In this model, the unified form for all membership functions of the parameters of the objective functions and constraints is as follows:

$$\mu_{\tilde{a}}(x) = \begin{cases} 0 & x < a \text{ or } d < x \\ (x^2 - a^2)/(b^2 - a^2) & a \leq x < b \\ 1 & b \leq x \leq c \\ (d^2 - x^2)/(d^2 - c^2) & c < x \leq d \end{cases} \tag{8.5.5}$$

For simplicity, we only represent the above form of membership function as a quadruple pair (*a, b, c, d*). Then, for the FMOLP model (8.5.4), all membership functions of fuzzy parameters of the objective functions and constraints are to be represented in the quadruple pair form and listed in Tables 8.1, 8.2, and 8.3, respectively.

Table 8.1: Membership functions of fuzzy objective functions' parameters

$\tilde{c}_{ij}$	1	2	3	4	5
1	(0.5, 1, 1, 2.5)	(8, 9, 9, 12)	(9, 10, 10, 13)	(0.5, 1, 1, 2.5)	(2, 3, 3, 6)
2	(8, 8.9, 9.2, 12)	(1, 1.9, 2.2, 5)	(1, 1.9, 2.2, 5)	(6, 6.9, 7.2, 10)	(3, 3.9, 4.2, 7)
3	(2, 3.9, 4.2, 5)	(4, 5.9, 6.2, 7)	(5, 6.9, 7.2, 8)	(2, 3.9, 4.2, 5)	(6, 7.9, 8.2, 9)

Table 8.2: Membership functions of fuzzy constraints' parameters

$\tilde{a}_{ij}$	1	2	3	4	5
1	(2, 3,3, 5)	(8,9, 9, 11)	(8, 9, 9, 11)	(4, 5, 5, 7)	(2,3, 3, 5)
2	(-6,-4.1,-3.9,-3)	(-2, -1.1, -0.9, -0.5)	(2, 2.9, 3.1, 5)	(-5, -3.1, -2.9, -2)	(-4, -2.1, -1.9, -1)
3	(2, 2.9, 3.1, 5)	(-11, -9.1, -8.9, -8)	(-11, -9.1, -8.9, -8)	(-6, -4.1, -3.9, -3)	(0, 0, 0, 0)
4	(4, 4.9, 5.1, 7)	(8, 8.9, 9.1, 11)	(9, 9.9, 10.1, 12)	(0.5, 0.9, 1.1, 2)	(-4, -2.1, -1.9, -1)
5	(2, 2.9, 3.1, 5)	(-5, -3.1, -2.9, -2)	(0, 0, 0, 0)	(0.5, 0.9, 1.1, 2)	(4, 4.9, 5.1, 7)

Table 8.3: Membership functions of fuzzy right-hand-side's parameters

$\tilde{b}_i$	1
1	(1038, 1038.9, 1039.1, 1041)
2	(93, 93.9, 94.1, 96)
3	(60, 60.9, 61.1, 63)
4	(923, 923.9, 924.1, 926)
5	(419, 419.9, 420.1, 422)

By the main steps of the IFMOLP method in Section 8.4.3, the procedure of solving the problem by using the FMODSS is as follows:

Stage 1: Initialisation

Step 1: Initially, the FMOLP model of the problem (8.5.4) is input into the system. The result is shown in Fig. 8.5.

Step 2: After having finished establishing the FMOLP model, decision makers will switch to windows as shown in Fig. 8.14 to solve the problem. Suppose the satisfactory degree α is set to 0.25, and each weight for three fuzzy objective functions is all equally set to 0.333.

Step 3: Click Button *Initiate*, an initial solution to the FMOLP model is generated. The decision variables are

$$x_1^* = 60.45\,,\ x_2^* = 53.43\,,\ x_3^* = 0\,, x_4^* = 5.09\,, x_5^* = 38.77 \qquad (8.5.6)$$

as displayed in the *Output* frame in Fig. 8.14, and the fuzzy objective functions are

$$\begin{cases} \tilde{f}_1^*\left(x_1^*, x_2^*, x_3^*, x_4^*, x_5^*\right) = 60.45\tilde{c}_{11} + 53.43\tilde{c}_{12} + 5.09\tilde{c}_{14} + 38.77\tilde{c}_{15} \\ \tilde{f}_2^*\left(x_1^*, x_2^*, x_3^*, x_4^*, x_5^*\right) = 60.45\tilde{c}_{21} + 53.43\tilde{c}_{22} + 5.09\tilde{c}_{24} + 38.77\tilde{c}_{25} \\ \tilde{f}_2^*\left(x_1^*, x_2^*, x_3^*, x_4^*, x_5^*\right) = 60.45\tilde{c}_{31} + 53.43\tilde{c}_{32} + 5.09\tilde{c}_{34} + 38.77\tilde{c}_{35} \end{cases} \qquad (8.5.7)$$

By clicking the corresponding grids one by one in the row '*Objectives*' in the *Output* frame in Fig. 8.14, the membership functions of fuzzy objective functions $\tilde{f}_1^*, \tilde{f}_2^*$, and $\tilde{f}_3^*$ in (8.5.7) are shown in Fig. 8.15, and the initial solution is logged and listed in the first row '*Trial 1*'. At this stage, the $\tilde{f}_1^*$, $\tilde{f}_2^*$, and $\tilde{f}_3^*$ are about 662.6828, 825.7988, and 877.0659, respectively.

Step 4: Since the Pareto optimal solution with the optimal decision variables x^* and fuzzy objective functions $\tilde{f}\left(x^*\right)$ exists, the procedure will move to the next step.

Step 5: Suppose decision makers are not satisfied with the initial solution in Step 3, then the interactive process will continue to Stage 2.

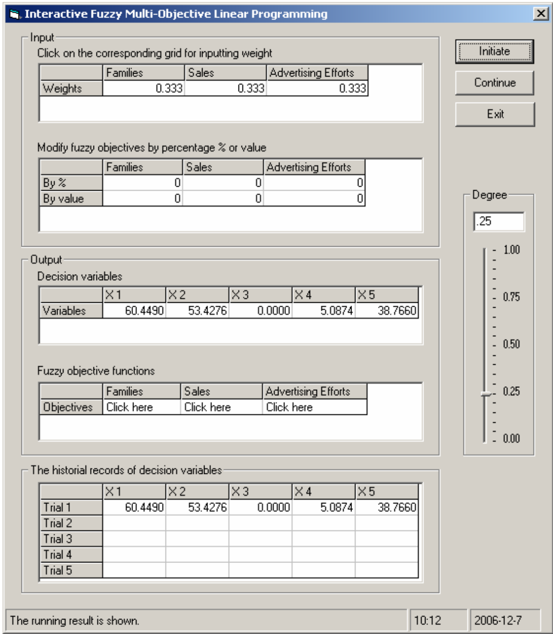


Fig. 8.14: Main window shown when the initial solution obtained

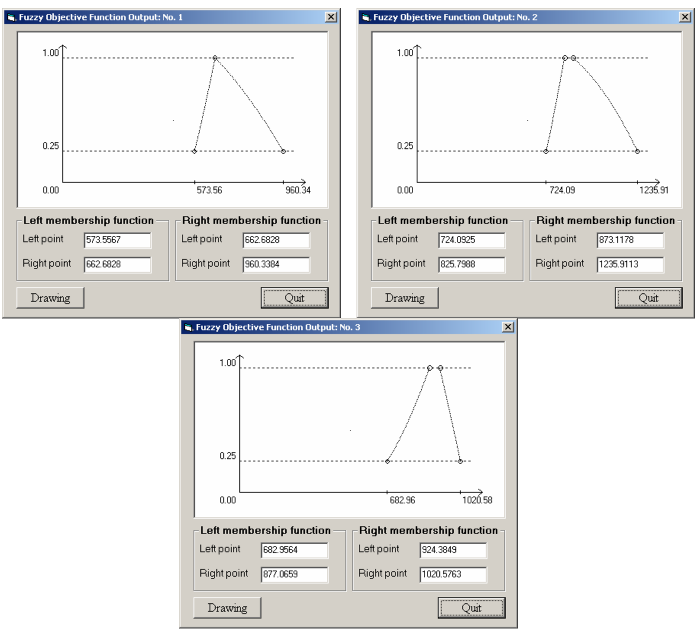


Fig. 8.15: Membership functions of $\tilde{f}_1^*$, $\tilde{f}_2^*$, and $\tilde{f}_3^*$ at Trial 1

Stage 2: Iterations

Iteration No 1:

Step 6: In this step, decision makers specify new fuzzy goals for the fuzzy objective functions to be achieved. Suppose these new fuzzy goals are assigned by decreasing the first and third fuzzy objective functions by 5% as the first and third fuzzy goals, respectively, and increasing the second fuzzy objective function by 5% as the second fuzzy goal based on the initial solution at Stage 1. That is,

$$(\tilde{g}_1, \tilde{g}_2, \tilde{g}_3) = \left(0.95 * \tilde{f}_1^*\left(x_1^*, x_2^*, x_3^*\right),\ 1.05 * \tilde{f}_2^*\left(x_1^*, x_2^*, x_3^*\right),\ 0.95 * \tilde{f}_3^*\left(x_1^*, x_2^*, x_3^*\right)\right) \quad (8.5.8)$$

By clicking the corresponding grid in the row '*By %*' in the *Input* frame in Fig. 8.16, the increasing and decreasing numbers are filled one by one in the textboxes.

Step 7: Click Button *Continue*, the new solution based on the fuzzy goals (8.5.8) is generated. Consequently, the decision variables are

$$x_1^* = 63.11,\ x_2^* = 49.76,\ x_3^* = 0.0,\ x_4^* = 12.73,\ x_5^* = 33.54, \quad (8.5.9)$$

and the fuzzy objective functions are

$$\begin{cases} \tilde{f}_1^*\left(x_1^*, x_2^*, x_3^*, x_4^*, x_5^*\right) = 63.11\tilde{c}_{11} + 49.76\tilde{c}_{12} + 12.73\tilde{c}_{14} + 33.54\tilde{c}_{15} \\ \tilde{f}_2^*\left(x_1^*, x_2^*, x_3^*, x_4^*, x_5^*\right) = 63.11\tilde{c}_{21} + 49.76\tilde{c}_{22} + 12.73\tilde{c}_{24} + 33.54\tilde{c}_{25} \\ \tilde{f}_2^*\left(x_1^*, x_2^*, x_3^*, x_4^*, x_5^*\right) = 63.11\tilde{c}_{31} + 49.76\tilde{c}_{32} + 12.73\tilde{c}_{34} + 33.54\tilde{c}_{35} \end{cases}. \quad (8.5.10)$$

The membership functions of $\tilde{f}_1^*$, $\tilde{f}_2^*$, and $\tilde{f}_3^*$ (8.5.10) are displayed in Fig. 8.17. The new solution is also logged and listed in the second row '*Trial 2*' in Fig. 8.16. At this iteration, the $\tilde{f}_1^*$, $\tilde{f}_2^*$, and $\tilde{f}_3^*$ are about 625.5503, 873.7452, and 855.1154, respectively. Comparing the fuzzy optimal objective functions $\tilde{f}_1^*$, $\tilde{f}_2^*$ and $\tilde{f}_3^*$ in (8.5.7) with the ones in (8.5.10), $\tilde{f}_1^*$ and $\tilde{f}_3^*$ got some decrement, and $\tilde{f}_2^*$ obtained some increment. That is the purpose of the fuzzy goals (8.5.8) at this iteration.

Step 8: Suppose decision makers are not satisfied with the solution in Step 7, the interactive process will carry on to the next iteration and go back to Step 6.

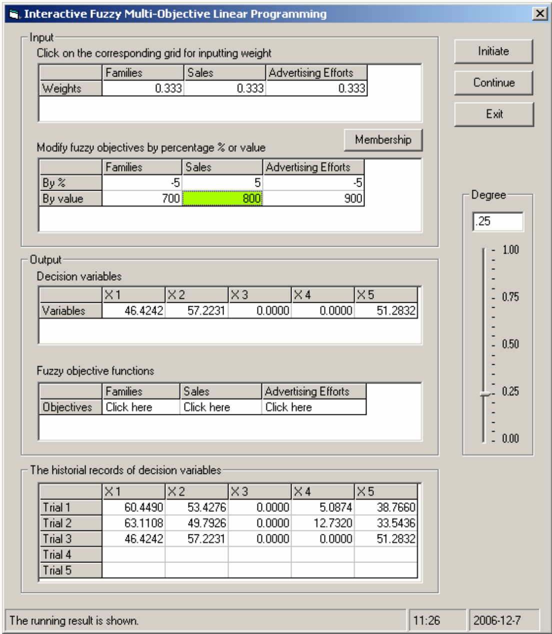


Fig. 8.16: Main window with an IFMOLP method for solving an FMOLP problem

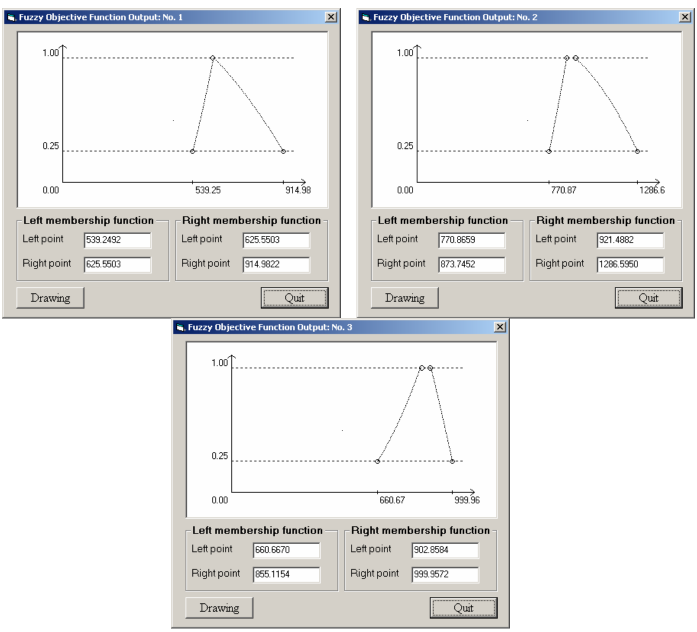


Fig. 8.17: Membership functions of $\tilde{f}_1^*$, $\tilde{f}_2^*$ and $\tilde{f}_3^*$ at Trial 2

Iteration No 2:

Step 6: Now, suppose decision makers set some new fuzzy goals as follows:

$$(\tilde{g}_1, \tilde{g}_2, \tilde{g}_3) = \left(\tilde{700}, \tilde{800}, \tilde{900}\right), \tag{8.5.11}$$

and the membership functions in quadruple pair format are listed as:

$$\begin{cases} u_{g_1} = (630, 700, 700, 770) \\ u_{g_2} = (720, 800, 800, 880) \\ u_{g_3} = (810, 900, 900, 990) \end{cases}. \tag{8.5.12}$$

By clicking on the corresponding grids in the row '*By value*' in the *Input* frame and Button *Membership* in Fig. 8.16, another Dialog Box similar to Fig. 8.9 will pop up. And the membership functions u_{g_1}, u_{g_2}, and u_{g_3} in (8.5.12) can be input in this Dialog Box sequentially.

Step 7: Clicking Button *Continue*, a compromise solution to the FMOLP problem based on the fuzzy goals in (8.5.11) and (8.5.12) is generated. The decision variables are

$$x_1^* = 46.42,\ x_2^* = 57.22,\ x_3^* = 0.0,\ x_4^* = 0.0,\ x_5^* = 51.28, \tag{8.5.13}$$

and the fuzzy objective functions are

$$\begin{cases} \tilde{f}_1^*(x_1^*, x_2^*, x_3^*, x_4^*, x_5^*) = 51.37\tilde{c}_{11} + 56.67\tilde{c}_{12} + 47.64\tilde{c}_{15} \\ \tilde{f}_2^*(x_1^*, x_2^*, x_3^*, x_4^*, x_5^*) = 51.37\tilde{c}_{21} + 56.67\tilde{c}_{22} + 47.64\tilde{c}_{25} \\ \tilde{f}_2^*(x_1^*, x_2^*, x_3^*, x_4^*, x_5^*) = 51.37\tilde{c}_{31} + 56.67\tilde{c}_{32} + 47.64\tilde{c}_{35} \end{cases}. \tag{8.5.14}$$

The membership functions of $\tilde{f}_1^*$, $\tilde{f}_2^*$, and $\tilde{f}_3^*$ in (8.5.14) are shown in Fig. 8.18, respectively. The $\tilde{f}_1^*$, $\tilde{f}_2^*$, and $\tilde{f}_3^*$ are about 715.3107, 721.8266, and 923.8416, respectively. The solution is also logged and listed in the row '*Trial 3*' in Fig. 8.16.

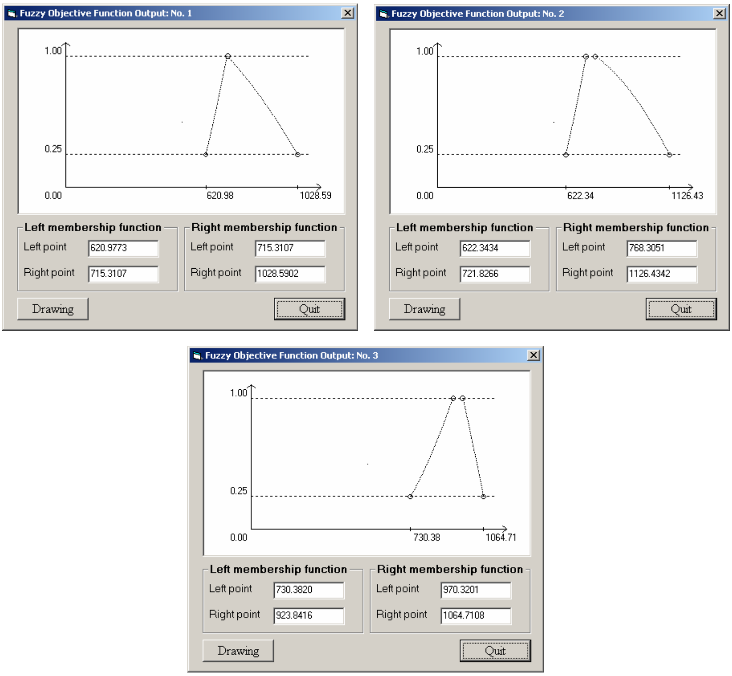


Fig. 8.18: Membership functions of $\tilde{f}_1^*$, $\tilde{f}_2^*$, and $\tilde{f}_3^*$ in Trial 3

Step 8: Suppose decision makers are now satisfied with the solution in Step 7, the whole interactive process stops, and the current solution is the final satisfactory solution of the FMOLP problem.

During the interactive process with the IFMOLP method, decision makers may have some different satisfactory degree α $(0 \le \alpha \le 1)$. With a different degree α, fuzzy parameters of the FMOLP model and fuzzy goals will take some value in different ranges, and for the solutions, decision variables will also be different and the fuzzy objection functions will be in different ranges as well. Tables 4.4, 4.5 and 4.6 list solutions at different stages with different satisfactory degrees α $(0 \le \alpha \le 1)$. With the optimal decision variables x_1^*, x_2^*, x_3^*, x_4^* and x_5^*, the fuzzy objective functions $\tilde{f}_1^*$, $\tilde{f}_2^*$ and $\tilde{f}_3^*$ are obtained by

$$\begin{cases} \tilde{f}_1^*\left(x_1^*, x_2^*, x_3^*, x_4^*, x_5^*\right) = \tilde{c}_{11}x_1^* + \tilde{c}_{12}x_2^* + \tilde{c}_{13}x_3^* + \tilde{c}_{14}x_4^* + \tilde{c}_{15}x_5^* \\ \tilde{f}_2^*\left(x_1^*, x_2^*, x_3^*, x_4^*, x_5^*\right) = \tilde{c}_{21}x_1^* + \tilde{c}_{22}x_2^* + \tilde{c}_{23}x_3^* + \tilde{c}_{24}x_4^* + \tilde{c}_{25}x_5^* \\ \tilde{f}_2^*\left(x_1^*, x_2^*, x_3^*, x_4^*, x_5^*\right) = \tilde{c}_{31}x_1^* + \tilde{c}_{32}x_2^* + \tilde{c}_{33}x_3^* + \tilde{c}_{34}x_4^* + \tilde{c}_{35}x_5^* \end{cases} \tag{8.5.15}$$

Table 8.4: Solutions at Initialisation Stage with some different satisfactory degree α

α	x_1^*	x_2^*	x_3^*	x_4^*	x_5^*
1.0	93.83	59.75	0.0	9.56	57.25
0.9	86.51	58.51	0.0	10.25	53.36
0.8	79.24	58.34	0.0	5.94	52.98
0.7	74.49	58.13	0.0	6.08	49.81
0.6	70.52	56.96	0.0	6.10	46.91
0.5	67.16	55.83	0.0	6.01	44.26
0.4	64.22	54.83	0.0	5.68	41.92
0.3	61.63	53.88	0.0	5.29	39.77
0.2	59.33	52.98	0.0	4.87	37.79
0.1	35.60	41.92	0.0	33.05	40.08
0.0	34.54	41.25	0.0	31.83	38.31

Table 8.5: Solutions at Iteration 1 with some different satisfactory degree α

α	x_1^*	x_2^*	x_3^*	x_4^*	x_5^*
1.0	96.89	55.51	0.0	19.14	50.85
0.9	89.38	54.43	0.0	19.45	47.32
0.8	81.99	55.36	0.0	14.85	47.10
0.7	77.13	54.27	0.0	14.67	44.14
0.6	74.49	58.16	0.0	14.36	41.42
0.5	69.71	52.17	0.0	13.96	38.94
0.4	66.73	51.26	0.0	13.33	36.77
0.3	64.14	50.41	0.0	12.65	34.71
0.2	61.84	49.60	0.0	11.95	32.94
0.1	60.56	47.76	0.0	13.46	29.70
0.0	42.44	39.54	0.0	33.64	31.65

Table 8.6: Solutions at Iteration 2 with some different satisfactory degree α

α	x_1^*	x_2^*	x_3^*	x_4^*	x_5^*
1.0	66.01	0.00	55.11	0.00	27.61
0.9	63.82	0.00	53.72	0.00	33.53
0.8	60.95	11.62	41.53	0.00	42.53
0.7	79.33	55.70	0.00	0.00	47.02
0.6	51.65	20.41	41.95	12.57	42.00
0.5	56.81	59.53	0.0	0.00	54.38
0.4	55.32	58.26	0.0	0.00	50.88
0.3	52.86	57.16	0.0	0.00	48.49
0.2	49.73	56.21	0.0	0.00	46.97
0.1	46.11	55.38	0.00	0.00	46.12
0	42.12	54.66	0.00	0.00	45.82

8.6 Summary

A fuzzy multi-objective DSS takes into account how to reach a solution when multiple objectives and fuzzy parameters are involved in the decision problem. We have developed an FMODSS based on the methods given in Chapter 7. The FMODSS helps decision makers to solve a multi-objective decision problem in practice. The FMODSS contains three methods each of which has particular features to support FMODM. This structure improves the usefulness of the system by different requirements and preferences of decision makers in their decision problems. Readers are recommended to use the case-based example given in Section 4 with the FMODSS in the attached CD to learn more the use of the system.

This page intentionally left blank

Part III

Fuzzy Group Decision Making

This page intentionally left blank

Chapter 9

Fuzzy MCDM

In most real world contexts, an MCDM (MADM) problem at tactical and strategic levels often involves fuzziness in its criteria (attributes) and decision makers' judgments. This kind of decision problems is called *fuzzy multi-criteria decision making* (FMCDM). We first give a case-based example to illustrate what is an FMCDM problem, and then present a general FMCDM model. We will discuss two FMCDM methods, fuzzy TOPSIS and fuzzy AHP, and then present a hybrid FMCDM method that has been implemented into a DSS, FMCDSS.

9.1 A Problem

Fuzzy MCDM technique has been one of the fastest growing areas in decision making and operations research during the last two decades. A major reason behind the development of FMCDM is due to the large number of criteria that decision makers are expected to incorporate in their actions and the difficulty of expressing decision makers' opinions by crisp values in practice. A typical FMCDM problem is performance evaluation.

A university plans to give an award to an academic who has the highest performance among all applicants. This issue involves multiple aspects. Each aspect has multiple evaluation criteria, and these criteria have different important degrees. All applicants of the university can be seen as alternatives. Since the judgments from the assessment committee of the university are usually vague rather than crisp, and hence can only be described by linguistic terms. It is a typical FMCDM problem.

In general, an academic's performance can be evaluated from three main aspects: *teaching*, *research*, and *service*. Each aspect contains a set of criteria and some criteria may also involve some sub-criteria. Fig. 9.1 gives a hierarchy of performance evaluation. Totally, 13 criteria are listed in the hierarchy.

The main criteria in the *teaching* aspect include course development (new subject design and existing subject update), teaching method research (innovative teaching method development and related grants, reports, and publications), teaching load (undergraduate subject teaching load, graduate subject teaching load, teaching material preparations, online teaching systems, and projects supervision), student performance (industry training and job finding), and student evaluation results (satisfaction on teaching contents, teaching methods, teaching attitude, assignments, and examination).

The main criteria on the *research* aspect include the number of research grants (international, national and internal), the amount of money funded in these grants, the number of publications (such as books, book chapters, journal papers, and conference papers), the quality of publications (such as journal quality index, citations to published materials), and the number of completion of research students.

The main criteria on the *service* aspect include service to the university (faculty and university committee members, leadership/participation in administration, and leadership/participation on management functions, and early career academic staff members), service to the professional society (referee or editor of scientific journals, invited speakers and guest lectures, member of national or international professional associations, organisation of conferences, and editorships), service to the community and related service performance (technical consultation, recommendation letters and sponsorship of visitors, and providing technical assistance to public policy analysis for local, state, national, and international governmental agencies).

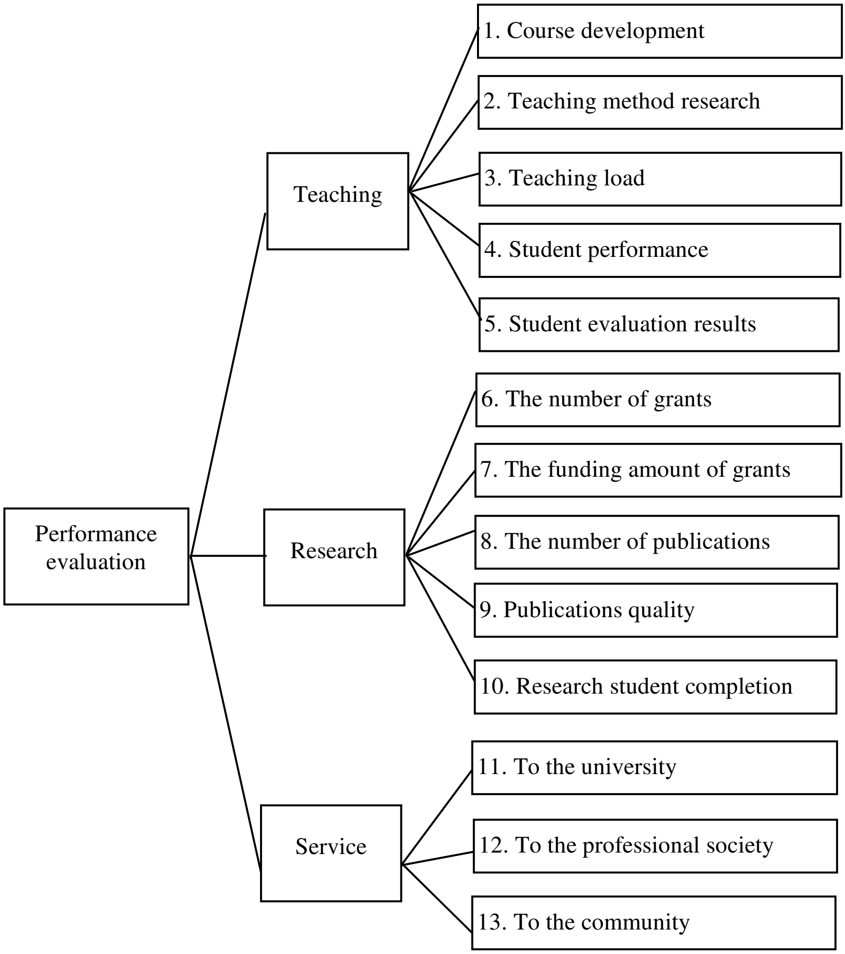


Fig. 9.1: Hierarchy of criteria for academic performance evaluation

Through the hierarchy of criteria from Fig. 9.1, the committee is able to assess all applicants' performance. To determine the importance degree of each criterion with respect to the goal, a set of linguistic terms may be used by the committee members to express their opinions to each applicant's performance. These linguistic terms are then represented by fuzzy numbers for achieving a final result. Table 9.1 lists some common used linguistic terms, described by triangle fuzzy numbers, for scoring weights of the three aspects and the 13 criteria in Fig. 9.1. Table 9.2 shows the needed linguistic terms and related triangle fuzzy numbers for assessing each applicant by the committee.

Table 9.1: Linguistic terms and related triangle fuzzy numbers for describing the weights

The importance degrees	*Membership functions*
Absolutely unimportant	(0, 0, 1/6)
Unimportant	(0, 1/6, 1/3)
Less important	(1/6, 1/3, 1/2)
Important	(1/3, 1/2, 2/3)
More important	(1/2, 2/3, 5/6)
Strongly important	(2/3, 5/6, 1)
Absolutely important	(5/6, 1, 1)

Table 9.2: Linguistic terms and related triangle fuzzy numbers for scoring

The scores	*Membership functions*
Lowest	(0, 0, 1/6)
Very low	(0, 1/6, 1/3)
Low	(1/6, 1/3, 1/2)
Medium	(1/3, 1/2, 2/3)
High	(1/2, 2/3, 5/6)
Very High	(2/3, 5/6, 1)
Highest	(5/6, 1, 1)

After having the importance degrees of criteria and all scores for applicants, the committee can use an FMCDM method to show who has the highest score among all applicants.

9.2 Models

As fuzziness may appear in different aspects and in different forms of an MCDM problem, FMCDM has been characterised in several ways. In principle, FMCDM constitutes the models of MCDM.

Mathematically, as described in Section 2.4, a typical MCDM (here and in the following context represents MADM) problem can be modelled as follows:

$$\text{(MCDM)} \begin{cases} \text{Select}: A_1, A_2, \ldots, A_m \\ \text{s.t.}: C_1, \mathrm{C}_2, \ldots, \mathrm{C}_n \end{cases} \tag{9.2.1}$$

where $A = (A_1, A_2, \ldots, A_m)$ denotes m alternatives, $C = (C_1, C_2, \ldots, C_n)$ represents n criteria. The *select* here is normally based on maximising a multi-attribute value (or utility) function elicited from the stakeholders. The model can be described in a matrix format:

$$D = \begin{array}{c} \\ A_1 \\ A_2 \\ \vdots \\ A_m \end{array} \begin{array}{c} \begin{array}{cccc} C_1 & C_2 & \cdots & C_n \end{array} \\ \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{12} & x_{22} & \cdots & x_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mn} \end{bmatrix} \end{array} \tag{9.2.2}$$

$$W = [w_1 \; w_2 \ldots w_n]$$

where $A_1, A_2, \ldots, A_m$ are alternatives from which decision makers choose; $C_1, C_2, \ldots, C_n$ are criteria with which alternative performances are measured; x_{ij}, $i = 1, \ldots, m$, $j = 1, \ldots, n$, is the rating of alternative A_i with respective to criterion C_j; and w_j is the weight of criterion C_j.

Basically, there are two issues involved in the MCDM model.

(1) The rating of alternative A_i with respect to criterion C_j given by decision makers expresses their judgments and preferences. These judgments and preferences are often described by linguistic terms, which are a kind of fuzzy values. That is, x_{ij} ($i = 1, \ldots, m$, $j = 1, \ldots, n$) can be fuzzy numbers.

(2) When we utilise weights to assess the relative importance of these multiple criteria, the weight for each criterion C_j may also be

described by linguistic terms. That is, w ($j=1,\ldots,n$) can be fuzzy numbers.

Hence, an FMCDM problem can be modelled to achieve Formula 9.1 in a matrix format as follows:

$$\tilde{D} = \begin{array}{c} \\ A_1 \\ A_2 \\ \vdots \\ A_m \end{array} \begin{array}{c} \begin{array}{cccc} C_1 & C_2 & \cdots & C_n \end{array} \\ \begin{bmatrix} \tilde{x}_{11} & \tilde{x}_{12} & \cdots & \tilde{x}_{1n} \\ \tilde{x}_{12} & \tilde{x}_{22} & \cdots & \tilde{x}_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{x}_{m1} & \tilde{x}_{m2} & \cdots & \tilde{x}_{mn} \end{bmatrix} \end{array} \tag{9.2.3}$$

$$\tilde{W} = [\tilde{w}_1 \ \tilde{w}_2 \ \ldots \tilde{w}_n]$$

where $\tilde{x}_{ij}$ $\forall i, j$ and $\tilde{w}_j$, $j=1,\ldots,n$ can be linguistic variables that are described by any form of fuzzy numbers. For example, in triangular fuzzy numbers, $\tilde{x}_{ij} = (a_{ij}, b_{ij}, c_{ij})$ and $\tilde{w}_j = (w_{j1}, w_{j2}, w_{j3})$.

However, there are many types of FMCDM models. The first category contains a number of ways to find a ranking: degree of optimality, Hamming distance, comparison function, fuzzy mean and spread, proportion to the ideal, left and right scores, centroid index, area measurement, and linguistic ranking methods. The second category is built around methods that utilise various ways to assess the relative importance of multiple criteria: fuzzy simple additive weighting methods, analytic hierarchy process, fuzzy conjunctive/disjunctive methods, fuzzy outranking methods, and maximin methods. We will discuss some typical methods in the following sections.

9.3 Fuzzy TOPSIS

From Chapter 2, TOPSIS deals with an m-alternatives MCDM problem as an m-points geometric system in an n-dimensional space.

Referring to the fuzzy decision matrix $\tilde{D}$ (9.2.3), the fuzzy TOPSIS method can be implemented by the following steps (Chen and Hwang, 1992):

Step 1: Calculate the normalised fuzzy decision matrix $\tilde{R}$ as

$$\tilde{R} = \left[\tilde{r}_{ij}\right]_{m \times n} \tag{9.3.1}$$

where

$$\tilde{r}_{ij} = \left(\frac{a_{ij}}{c_j^*}, \frac{b_{ij}}{c_j^*}, \frac{c_{ij}}{c_j^*} \right), \ j \in B;$$

$$\tilde{r}_{ij} = \left(\frac{a_j^-}{c_{ij}}, \frac{a_j^-}{b_{ij}}, \frac{a_j^-}{a_{ij}} \right), \ j \in C;$$

$$c_j^* = \max_i c_{ij} \quad \text{if } j \in B;$$

$$a_j^- = \min_i a_{ij} \quad \text{if } j \in C,$$

and B and C are the set of *benefit criteria* and the set of *cost criteria*, respectively (Chen and Hwang, 1992).

Step 2: Calculate the weighted normalised fuzzy decision matrix $\tilde{V}$ as

$$\tilde{V} = \left[\tilde{v}_{ij}\right]_{m \times n} \tag{9.3.2}$$

where $\tilde{v}_{ij} = \tilde{r}_{ij} \cdot \tilde{w}_j$.

Step 3: Identify the fuzzy positive-ideal solution (FPIS, $\tilde{A}^*$) and the fuzzy negative-ideal solution (FNIS, $\tilde{A}^-$) as

$$\tilde{A}^* = \left(\tilde{v}_1^*, \tilde{v}_2^*, \ldots, \tilde{v}_n^*\right), \tag{9.3.3}$$

$$\tilde{A}^- = \left(\tilde{v}_1^-, \tilde{v}_2^-, \ldots, \tilde{v}_n^1\right) \tag{9.3.4}$$

where $\tilde{v}_j^* = (1,1,1)$ and $\tilde{v}_j^- = (0,0,0)$, $j = 1,2,\ldots,n$.

Step 4: Calculate the distances of each alternative from $\tilde{A}^*$ and $\tilde{A}^-$ as

$$d_i^* = \sum_{j=1}^{n} d\left(\tilde{v}_{ij}, \tilde{v}_j^*\right), \quad i = 1,2,\ldots,m, \tag{9.3.5}$$

$$d_i^- = \sum_{j=1}^{n} d\left(\tilde{v}_{ij}, \tilde{v}_j^-\right), \quad i = 1,2,\ldots,m, \tag{9.3.6}$$

where $d(\cdot,\cdot)$ is the distance measurement between two fuzzy numbers.

Step 5: Calculate the closeness coefficient of each alternative as

$$CC_i = \frac{d_i^-}{d_i^* + d_i^-}, \quad i = 1,2,\ldots,m. \tag{9.3.7}$$

Step 6: Rank alternatives according to the values of CC_i in descending order and choose an alternative with the maximum CC_i.

9.4 Fuzzy AHP

Fuzzy analytic hierarchy process (AHP) is a direct extension of Saaty's AHP method (1980). Referring to the AHP method in Chapter 2, in this fuzzy AHP, the elements in the reciprocal matrices are represented by fuzzy numbers.

The fuzzy AHP method has the following steps:

Step 1: Determine the relative importance of the decision criteria. By a pairwise comparison, the matrix $\tilde{R}$, containing fuzzy estimates for the relative significance of each pair of factors, is constructed.

$$\tilde{R} = \begin{array}{cc} & \begin{matrix} C_1 & C_2 & \cdots & C_n \end{matrix} \\ \begin{matrix} C_1 \\ C_2 \\ \vdots \\ C_n \end{matrix} & \begin{bmatrix} \tilde{r}_{11} & \tilde{r}_{12} & \cdots & \tilde{r}_{1n} \\ \tilde{r}_{12} & \tilde{r}_{22} & \cdots & \tilde{r}_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{r}_{n1} & \tilde{r}_{n2} & \cdots & \tilde{r}_{nn} \end{bmatrix} \end{array} \tag{9.4.1}$$

Step 2: Calculate fuzzy estimates for the weights or priorities of the decision criteria based on the matrix $\tilde{R}$ (9.4.1).

Step 3: Make pairwise comparisons of alternatives under each of the criteria separately. Then, n matrices $\left(\tilde{R}^1, \tilde{R}^2, \ldots, \tilde{R}^n\right)$, each of which contains fuzzy estimates for the relative significance of each pair of alternatives, is constructed.

$$\tilde{R}^i = \begin{array}{cc} & \begin{matrix} A_1 & A_2 & \cdots & A_m \end{matrix} \\ \begin{matrix} A_1 \\ A_2 \\ \vdots \\ A_m \end{matrix} & \begin{bmatrix} \tilde{r}^i_{11} & \tilde{r}^i_{12} & \cdots & \tilde{r}^i_{1m} \\ \tilde{r}^i_{21} & \tilde{r}^i_{22} & \cdots & \tilde{r}^i_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{r}^i_{1m} & \tilde{r}^i_{2m} & \cdots & \tilde{r}^i_{mm} \end{bmatrix} \end{array} \quad i = 1, \ldots, n \tag{9.4.2}$$

Step 4: Calculate fuzzy estimates for the weight of each alternative under each criterion separately, based on the matrices $\left(\tilde{R}^1, \tilde{R}^2, \ldots, \tilde{R}^n\right)$ (9.4.2).

Step 5: Obtain a final score for each alternative by adding the weights per alternative (obtained in Step 4) multiplied by the weights of the corresponding criteria (obtained in Step 2).

9.5 A Hybrid Method

To define positive and negative ideal solutions is an advantage of the TOPSIS method, and to make a consistence check is an advantage of AHP. A hybrid FMCDM method is proposed by integrating the two features to deal with a hierarchy decision problem. Particularly, in this hybrid method, fuzzy numbers can be described in any form to handle linguistic terms and other uncertain values. The method is designed by the following nine steps.

Step 1: Set up weights for all aspects and related criteria

Referring to a set of aspects $F = (F_1, F_2, \ldots, F_n)$, let $WF = (WF_1, WF_2, \ldots, WF_n)$ be the weights of these aspects, where $WF_i \in$ {*Absolutely unimportant, Unimportant, Less important, Important, More important, Strongly important, Absolutely important*}, as shown in Table 9.1, for example, and are described by fuzzy numbers $\tilde{a}_1, \tilde{a}_2, \ldots, \tilde{a}_n$.

For an aspect F_i, let $C_i = \{C_{i1}, C_{i2}, \cdots, C_{it_i}\}$, $i = 1,2,\ldots,n$ be a set of the selected criteria with respect to the aspect F_i. Let $WC_i = \{WC_{i1}, WC_{i2}, \cdots, WC_{it_i}\}$, $i = 1,2,\ldots,n$, be the weights for the set of criteria, where WC_{ij} will be signed a value from the same linguistic term list as WF_i above, for example, and are described by fuzzy numbers $\tilde{c}_1, \tilde{c}_2, \ldots, \tilde{c}_t$. For the example given in Fig. 9.1, '*Teaching*' is an aspect of performance, five criteria to evaluate it are '*Course development*,' '*Teaching method research*,' '*Teaching load*,' '*Student performance*,' and '*Student evaluation results*.'

Step 2: Finalise these aspects and criteria by some rules

For example, a criterion can be ignored when

- it has a very low weight;
- the degree of its weight is much less than others; or
- its related sub-criteria are the subset of another criterion.

Step 3: Set up the relevance degree of each alternative on each criterion

Let $A=(A_1, A_2, \ldots, A_m)$ be a set of alternatives, $AC_i^k=\{AC_{i1}^k, AC_{i2}^k, \cdots, AC_{it_i}^k\}$ be the relevance degree of alternative A_k on criterion C_i, $i=1,2,\ldots,n$, $k=1,2,\ldots,m$, where $AC_{ij}^k \in$ {*Lowest*, *Very low, Low, Medium, High, Very high, Highest*}, as shown in Table 9.2, for example, and are described by fuzzy numbers $\tilde{b}_1, \tilde{b}_2, \ldots, \tilde{b}_k$. Table 9.3 further describes the relationships among these aspects, criteria, alternatives, their weights, and decision makers' evaluation values (scores).

Table 9.3: The relationships among the aspects, criteria, alternatives, their weights, and evaluation values

				A_1	...	A_m
F_1	WF_1	C_{11}	WC_{11}	AC_{11}^1	$\ldots$	AC_{11}^m
		$\ldots$	$\ldots$	$\ldots$	$\ddots$	$\ldots$
		C_{1t_1}	WC_{1t_1}	$AC_{1t_1}^1$	$\ldots$	$AC_{1t_1}^m$
$\ldots$	$\ldots$	$\ldots$	$\ldots$	$\ldots$	$\ddots$	$\ldots$
F_n	WF_n	C_{n1}	WC_{n1}	AC_{n1}^1	$\ldots$	AC_{n1}^m
		$\ldots$	$\ldots$	$\ldots$	$\ddots$	$\ldots$
		C_{nt_n}	WC_{nt_n}	$AC_{nt_n}^1$	$\ldots$	$AC_{nt_n}^m$

Step 4: Normalise the weights for criteria

The weights for the criteria $WC_i=\{WC_{i1}, WC_{i2}, \cdots, WC_{it_i}\}$, $i=1,2,\ldots,n$ are normalised and denoted as

$$WC_{ij}^* = \frac{WC_{ij}}{\sum_{j=1}^{t_i} WC_{ij\,0}^{\,R}}, \quad \text{for } j=1,2,\cdots,t_i, i=1,2,\cdots,n. \tag{9.5.1}$$

where the C_{ij0}^{R} is the right end of 0-cutset (Chapter 5).

Step 5: Calculate the relevance degrees

The relevance degree FA_i^k of the alternatives A_k on the aspect F_i, $i=1,2,\ldots,n,\ k=1,2,\ldots,m,$ are calculated by using $FA_i^k = WC_i^* \times AC_i^k = \sum_{j=1}^{t_i} WC_{ij}^* \times AC_{ij}^k,\ i=1,2,\ldots,n,\ k=1,2,\ldots,m.$

Step 6: Normalise the relevance degrees

The relevance degrees FA_i^k of the alternatives A_k on the aspect F_i , $i=1,2,\ldots,n$, $k=1,2,\ldots,m$ are normalised based on $FA^k = \{FA_1^k, FA_2^k, \cdots, FA_n^k\},\ k=1,2,\ldots,m$.

$$\overline{FA}_i^k = \frac{FA_i^k}{\sum_{i=1}^{n} FA_{i\,0}^{k\,R}}, \quad \text{for } i=1,2,\cdots,n, k=1,2,\cdots,m. \tag{9.5.2}$$

Step 7: Calculate the alternatives relevance degrees

The relevance degree S_k of the alternatives A_k on the aspects F, $k=1,2,\ldots,m$ is calculated by using $S_k = \overline{FA}^k \times WF = \sum_{i=1}^{n} \overline{FA}_i^k \times WF_i$ $k=1,2,\ldots,m.$ Here, S_k is still a fuzzy number.

Step 8: Calculate the positive and negative distances

The results $S_k,\ k=1,2,\cdots,m$ are normalised as positive fuzzy numbers, and their ranges belong to the closed interval [0, 1]. We define fuzzy positive-ideal solution (FPIS, S^*) and fuzzy negative-ideal solution (FNIS, S^-) as:

$$S^* = 1 \quad \text{and} \quad S^- = 0.$$

The distance between each S_k and S^* is called a *positive distance*, and the distance between S_k and S^- is called a *negative distance*. The two kinds of distances are calculated respectively by

$$d_k^* = d(S_k, S^*) \quad \text{and} \quad d_k^- = d(S_k, S^-), \quad k=1,2,\cdots,m, \tag{9.5.3}$$

where

$$d(\tilde{a},\tilde{b}) = \left(\int_0^1 \frac{1}{2}[(a_\lambda^L - b_\lambda^L)^2 + (a_\lambda^R - b_\lambda^R)^2] d\lambda \right)^{\frac{1}{2}} \tag{9.5.4}$$

is the distance measure between two fuzzy numbers $\tilde{a}$ and $\tilde{b}$.

Step 9: Get the satisfactory solution

A closeness coefficient is defined to determine the ranking order of alternatives once the d_k^* and d_k^- of each alternative A_k, $k = 1,2,\ldots,m$ are obtained. The closeness coefficient of each alternative is calculated as:

$$D_k = \frac{1}{2}\left(d_k^- + (1 - d_k^*)\right), \quad k = 1, 2, \cdots, m. \tag{9.5.5}$$

The alternative A_k with the largest D_k, $\max\{D_1,\ldots,D_m\}$, is the best solution for the decision problem.

9.6 Case-Based Examples

This hybrid FMCDM method has been implemented in a fuzzy multi-criteria DSS (FMCDSS). Here, we give two examples to demonstrate the use of the system.

Example 1: Buying a car

Chris wants to buy a car. He has three alternatives in his mind: *Toyota*, *Audi*, and *Ford*. He also has two aspects to consider for the selection: *Cost* and *Capacity*, and has more concern on *Cost* over *Capacity*. For the *Cost*, he has three criteria: *purchase price*, *mileage*, and *service cost* (repair frequency and average cost per time). For the *Capacity*, he has two criteria: *safety* and *comfort*.

Firstly: he sets up the problem as shown in Fig. 9.2: two evaluation aspects and three alternatives.

Secondly, he inputs the name of the three alternatives: *Toyota*, *Audi*, and *Ford* (Fig. 9.3) and all criteria: *purchase price*, *mileage*, *service cost*, *safety*, and *comfort* (Fig. 9.4).

Thirdly (corresponding to Steps 1 and 2 in the hybrid FMCDM method), he chooses the weights for all aspects and criteria (Fig. 9.5).

For example, the aspect, *Cost*, is '*more important*,' and the criterion, *Price*, is '*strongly important*.'

Fourthly (corresponding to Step 3), he sets up the relevant degree (a score with a linguistic term) of each alternative on each criterion as shown in Fig. 9.6. We can see that *Toyota* has a good *purchase price*, so he puts a '*very high*' satisfactory degree on the *Cost*.

Finally (corresponding to Steps 4 to 9), he obtains the result for the problem as shown in Fig. 9.7. *Toyota* is selected as the best one for his situation.

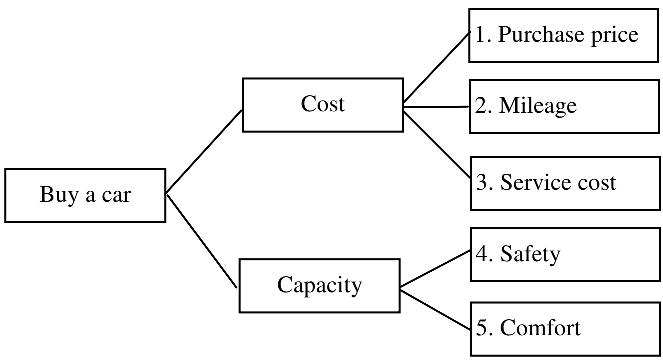


Fig. 9.2: Hierarchy of criteria for 'Buy a car'

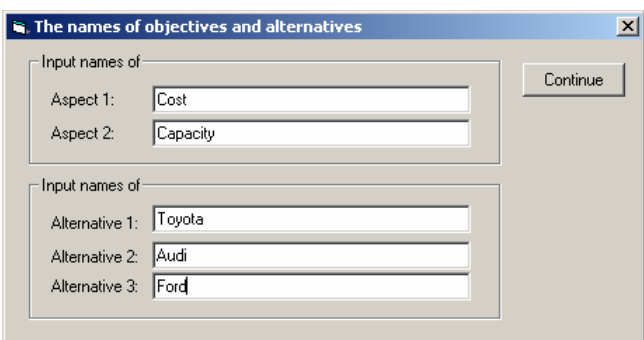


Fig. 9.3: Input aspects and alternatives

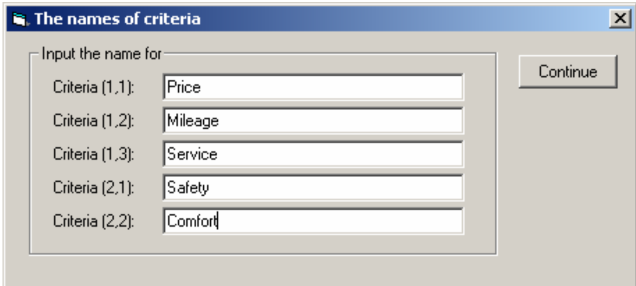


Fig. 9.4: Input criteria

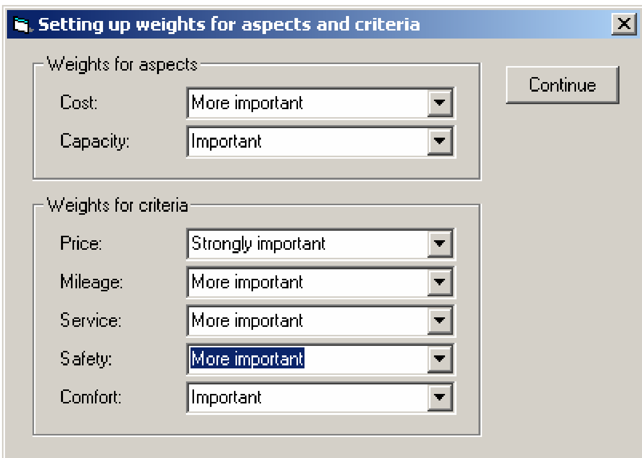


Fig. 9.5: Choosing the weights for aspects and criteria

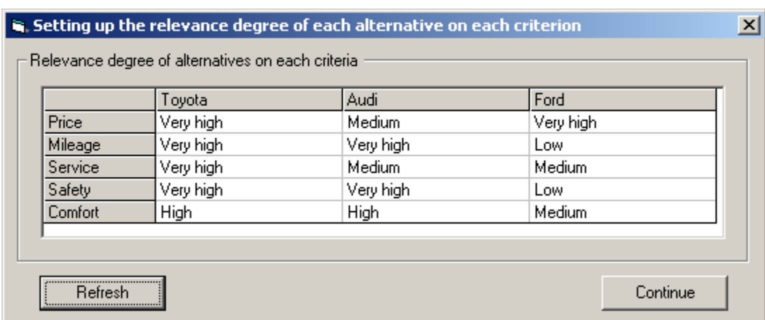


Fig. 9.6: Set up the relevant degree of each alternative on each criterion

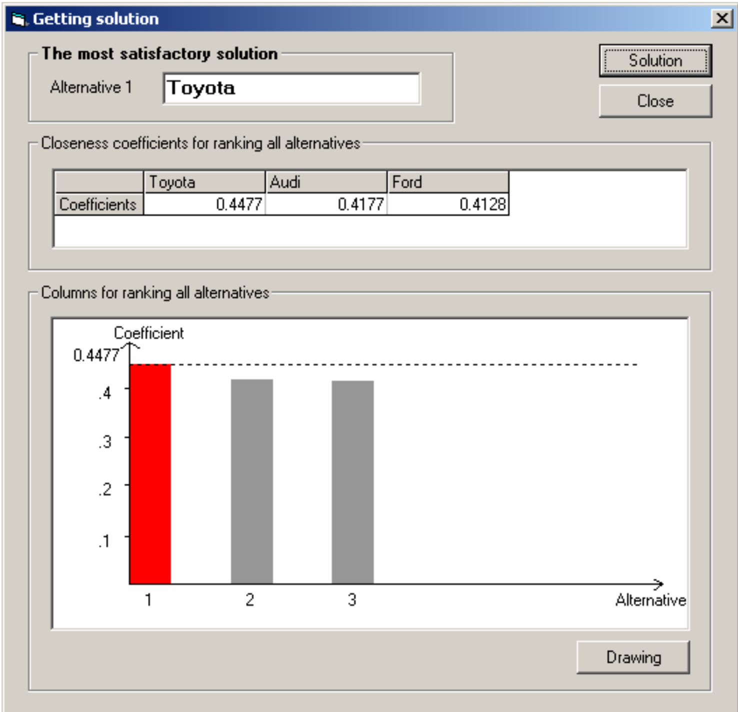


Fig. 9.7: Showing the final result

Example 2: Performance evaluation

We use the example given in Section 9.1. Suppose there are three academic staff as applicants, and the academic performance evaluation criteria are as the ones listed in Fig. 9.1.

Firstly, the committee sets up the FMCDM problem, which has three alternatives, three evaluation aspects, and totally 13 criteria.

Secondly (corresponding to Steps 1 and 2 in the hybrid FMCDM method), they choose weights for the three aspects and each aspect's related evaluation criteria as shown in Fig. 9.8. We can see that '*Teaching*' has a '*strongly important*' weight, and '*Research*' is '*more important*.'

Thirdly (corresponding to Step 3), they set up the relevant degree of each alternative on each criterion as shown in Fig. 9.9. For example, *Applicant 2* has received a '*high*' score on '*student performance*.'

Finally (corresponding to Step 4 to Step 9), the committee obtains the result for the evaluation problem (Fig. 9.10). The result shows that *Applicant 2* has the highest score (0.4964), and can thus obtain the award.

The two examples show how to use the proposed FMCDM method and the FMCDSS to solve some evaluation decision problems.

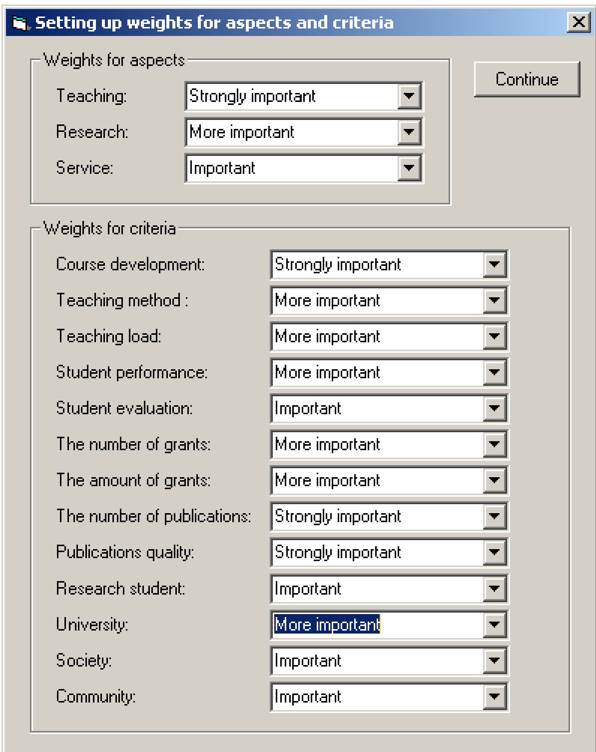


Fig. 9.8: Choose weights for aspects and each aspect's related evaluation criteria

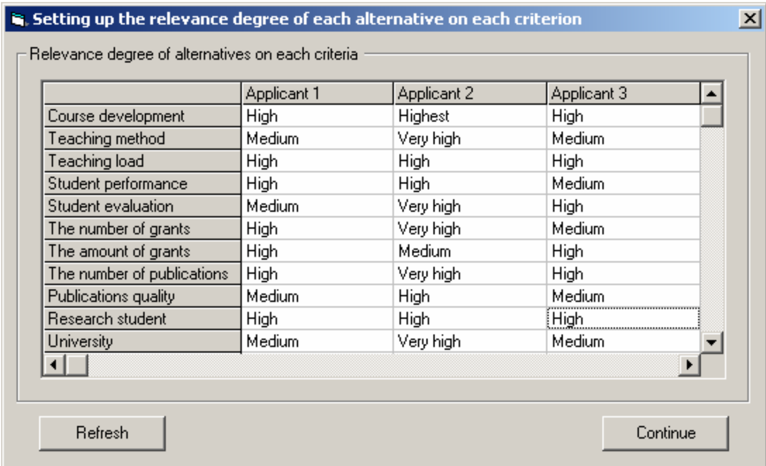


Fig. 9.9: Set up the relevant degree of each criterion on each alternative

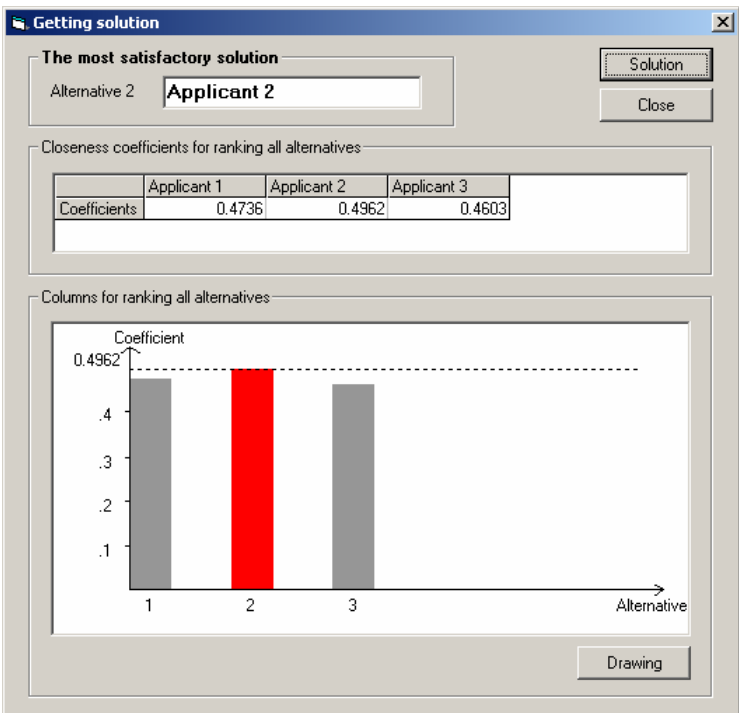


Fig. 9.10: The final result for the performance evaluation

9.7 Summary

Many decision problems involve a complex situation in which some qualitative criteria are with in a hierarchy and must be considered simultaneously. The judgments from decision makers are often in vague rather than in crisp numbers. It is more suitable to express their preferences in criteria and their judgments for alternatives by linguistic terms (fuzzy numbers) instead of crisp numbers. Fuzzy AHP, fuzzy TOPSIS, and hybrid FMCDM methods are presented in this chapter, and two case-based examples are given to illustrate how to use these FMCDM methods to many real world problems. Chapter 16 will further illustrate a real world application of the hybrid FMCDM method.

Chapter 10

Fuzzy Group Decision Making

Group decision making takes into account how people work together in reaching a decision. Uncertain factors often appear in a group decision process. After giving a rational-political group decision model, we first identify three main uncertain factors involved in a group decision-making process: decision makers' roles, preferences for alternatives, and judgments for assessment-criteria. We then present an intelligent fuzzy multi-criteria group decision-making (FMCGDM) method to deal with the three uncertain factors and generate a group satisfactory decision. The solution is in the most acceptable degree of the group. Inference rules are particularly introduced into the method for checking the consistence of individual preferences. Finally, we illustrate the proposed group decision-making method by a case-based example.

10.1 The Rational-Political Model

A group satisfactory solution is the one that is the most acceptable by the group of individuals as a whole. Since the impact of the group decisions (the selection of the satisfactory solution) affects organisational performance, it is crucial to make the group decision-making process as efficient and effective as possible. Three factors may influence the assessment of utility of alternatives and the deriving of the group satisfactory solution.

The first one is an *individual's role* (*weight*) in the ranking and selection of the satisfactory solutions. There may be a group leader or leaders who play more important roles in a particular group decision-making process. Although decision makers try to influence other members to adopt their viewpoint, powerful members will sway strongly decision making than other members. Group members thus have different *weights* in a group decision-making process, and the situation should be reflected on the generation process of the group satisfactory solution.

The second factor is an *individual's preference* for alternatives. Group members may not know all information relate to a decision problem or may not consider all relevant information to the decision problem. Also, they may have different understanding for the same information, different experiences in the area of current decision problems, and different preferences for different alternatives. The different preferences of group members may have impact directly on the deriving of the group satisfactory solution.

The third factor is *criteria* for assessing these alternatives. Assessment-criteria are usually determined through discussions within decision groups. Goals or priorities of decision objectives are often as assessment-criteria for MODM problems. In a real situation, different group members may have different viewpoints in assessment-criteria for a decision problem because of workload, time and inexperience at assessing a problem all affect determining assessment-criteria. Different members may often have different judgments in comparing the importance between a pair of assessment-criteria, for instance, which criterion is more important than another. Obviously, what assessment-criteria are used and how priority of each assessment-criterion is processed will directly influence the ranking of these alternatives and selection of the group satisfactory solution.

Based on our discussion about group decision-making models in Chapter 3, here we present a rational-political model for group decision making to support the achievement of group consensus in an uncertain environment by considering the three uncertain factors.

The rational-political model is consensus rule-based and takes advantage of both rational and political models of group decision making. By inheriting the optimisation property of the rational model, it shows a sequential approach to make a group decision and to get the best solution for the group decision. By considering the political model, it allows decision makers to have inconsistent assessment, incomplete information and inaccurate opinions for alternatives. The model deals with the three identified uncertain factors together based on the use of linguistic assessments: decision makers' weights in reaching a satisfactory solution, decision makers' preferences for alternatives, and decision makers' judgments for solution assessment-criteria.

As shown in Fig. 10.1, the model is assumed that a decision problem is defined, requirements are determined, and objectives are established. Group members will propose alternatives for the decision problem, and then rank these alternatives to select N of them. A set of assessment-criteria for assessing or ranking these alternatives will be nominated by these group members or generated through running a suitable model operated by them. Finally, T criteria will be used. Group members are awarded or assigned weights before or at the beginning of the decision-making process. Although group members may have different experiences, opinions and information at hand for the decision problem, they must participate in the group aggregation process to ensure that the disparate individuals come to share the same decision objectives. These group members will be required to give their individual judgments for the priority of the proposed assessment-criteria and preferences for alternatives under these assessment-criteria by linguistic terms. As a result, it allows incorporating more human consistency in group decision making.

To apply this model in developing a practical group decision-making method, we need define related linguistic consensus degrees and linguistic distances acting on the three uncertain factors. The consensus degrees will indicate how far a group of individuals is from the maximum consensus, and the linguistic distances will indicate how far each individual is from current consensus labels over the preferences. These will be discussed in following sections.

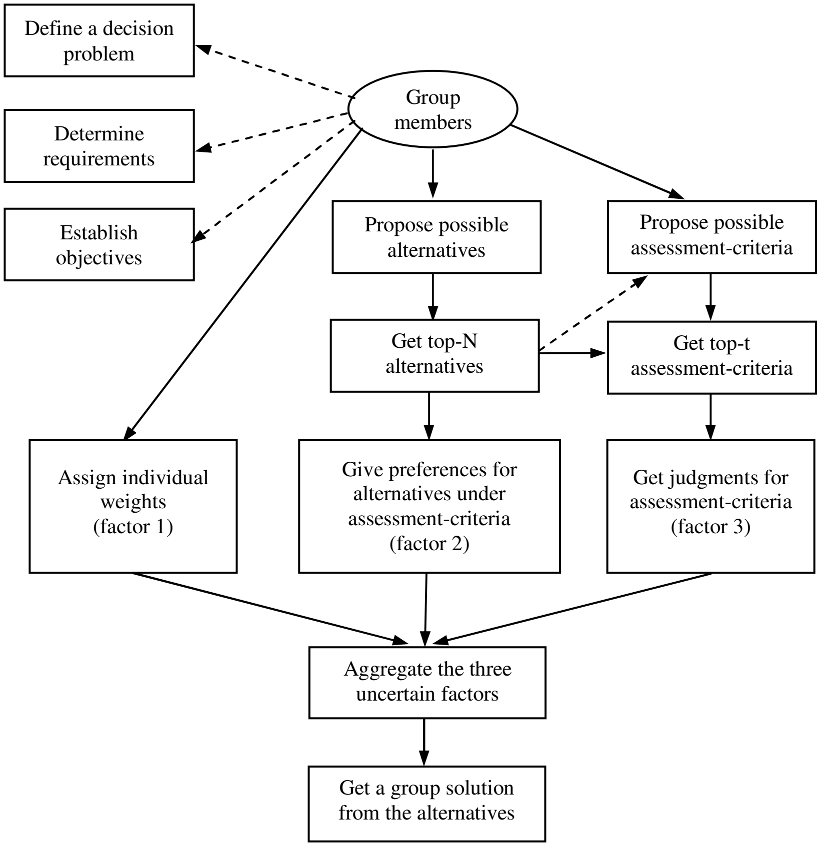


Fig. 10.1: The rational-political group decision-making model

10.2 Uncertain Factors

Any individual role in a decision process, a preference for alternatives, and a judgment for assessment-criteria are often expressed by linguistic terms. For example, an individual role can be described by using linguistic terms '*normal*,' '*more important*,' or '*most important*.' Similarly, to express decision makers' preference for an alternative, linguistic term such as '*low*' and '*high*' could be also used. Similarly, to express decision makers' judgment for comparison of a pair of assessment-criteria, '*equally important*' or '*A is more important than B*' could be used as well. However, precise mathematical approaches are not efficient enough to tackle such uncertain variables and derive a satisfactory solution. Since these linguistic terms reflect the uncertainty, inaccuracy and fuzziness of decision makers, fuzzy numbers and fuzzy operations can be directly applied to deal with them.

Much research has been conducted in the area of group decision-making under the application of fuzzy set theory and fuzzy decision-making theory. Some of them have also applied the concept of linguistic variables to handle linguistic terms and approximate reasoning in a group decision-making problem. Several typical fuzzy group decision-making methods have been developed and focused respectively on the three uncertain factors identified above. Some researches have been carried out in describing the uncertainty of individual preferences for alternatives and aggregating these fuzzy individual preferences into a group consensus decision. The uncertainty of individual roles, or individual weights, in attempting to reach a group satisfactory solution has been discussed in the literature of this area. Also, some comprehensive researches including the applications of fuzzy decision-making methods, comparison between some methods and survey-based approach analysis, have been reported in literature.

The result presented in this chapter firstly extends the decision-making method to deal with all these three uncertain factors mentioned in Section 10.1 together as they may exist in group decision-making simultaneously. Secondly, it allows these uncertain factors to be described by linguistic terms with fuzzy numbers. The third one is that it adds intelligent checking for logical consistence of individual decision makers' preferences. Each individual's preferences should not be self-conflict and the information provided by decision makers should be consistent. To avoid inconsistency-causing errors, intelligence-based inference should be functioned in a group decision-making process.

10.3 An Intelligent FMCGDM Method

In this section, we propose an intelligent fuzzy multi-criteria group decision-making (FMCGDM) method.

Let $P = \{P_1, P_2, \ldots, P_n\}$, $n \geq 2$, be a given finite set of decision makers to select a satisfactory solution from alternatives or identify a number of important issues with raking for the decision problem. The proposed method consists of ten steps within three stages:

Stage 1: Alternatives, assessment-criteria, and individual weights generation

Step 1: When a decision problem is proposed in a group, each member can raise one or several possible strategies or alternatives. Let $S^{\#} = \{S_1^{p_1}, S_2^{p_1}, \ldots S_{m_{p1}}^{p_1}, \ldots\ldots S_1^{p_n}, S_2^{p_n}, \ldots\ldots S_{m_{pn}}^{p_n}\}$, where $s_j^{p_i}$ is the jth alternative for a decision problem raised by group member p_i'. Through a discussion and summarisation, $S = \{S_1, S_2, \ldots, S_m\}$, $m \geq 2$ is selected from $S^{\#}$ as alternatives for the decision problem.

Step 2: If the decision problem is a multi-objective problem, the objectives can be as assessment-criteria. In a general situation, each group member P_k ($k = 1, 2, \ldots, n$) can propose a_k assessment-criteria $(C_1^k, C_2^k, \cdots, C_{a_k}^k)$ for ranking and assessing these alternatives. All members' assessment-criteria are put into a criterion pool and top-t criteria, $C = \{C_1, C_2, \ldots, C_t\}$, are chosen as assessment-criteria for the decision problem in the group.

Step 3: As group members play different roles in an organisation and hence have different degrees of influence for the selection of the group satisfactory solution. That means the relative importance of each decision maker may not equal in a decision group. Some members are more powerful than the others for a specific decision problem. Therefore, in the method, each member is assigned with a weight that is described by a linguistic term $\tilde{v}_k, k = 1, 2, \cdots, n$. These terms are determined through discussions in the group or assigned by a higher management level (say, the leader) before or at the beginning of the decision process. For example, P_k is assigned with '*important*' and P_l '*more important*.' Possible linguistic terms used in the factor are shown in Table 10.1.

Table 10.1: Linguistic terms for describing weights of decision makers

Linguistic terms	*Fuzzy numbers*
Normal	c_1
Important	c_2
More important	c_3
Most important	c_4

Stage 2: Individual preference generation

Step 4: Each decision maker P_k (k = 1, 2, ..., n) is required to express their opinion for assessment-criteria by a pairwise comparison of the relative importance of these criteria of fuzzy AHP method.

An initial pairwise comparison matrix $E = [\tilde{e}_{ij}^{k}]_{t\times t}$ is firstly established, where $\tilde{e}_{ij}^{k}$ represents the quantified judgments on pairs of assessment-criteria C_i and C_j (i, j =1, 2, …, t, $i \neq j$). The comparison scale belongs to a set of linguistic terms that contain various degrees of preferences required by decision makers P_k (k = 1, 2, ..., n), or take a value '*'. The linguistic terms are shown in Table 10.2. Character '*' represents that decision makers P_k (k = 1, 2, ..., n) do not know or cannot compare the relative importance of assessment-criteria C_i and C_j.

Table 10.2: Linguistic terms for the comparison of assessment-criteria

Linguistic terms	Fuzzy numbers
Absolutely less important	a_1
Much less important	a_2
Less important	a_3
Equally important	a_4
More important	a_5
Much more important	a_6
Absolutely more important	a_7

By using the following various linguistic inference rules, the inconsistence of each pairwise comparison matrix $E = [\tilde{e}_{ij}^k]_{t\times t}$ is corrected:

Positive-Transitive rule: If $\tilde{e}_{ij}^k = a_s$ $(s = 4, 5, 6, 7)$ and $\tilde{e}_{jm}^k = a_t$ $(t = 4, 5, 6, 7)$, then $\tilde{e}_{im}^k = a_{\max(s,t)}$. For example, if C_i is as '*equally important*' as C_j (s = 4), and C_j is '*much more important*' than C_m (t = 6), then C_i is '*much more important*' than C_m.

Negative-Transitive rule: If $\tilde{e}_{ij}^k = a_s$ $(s = 3, 2, 1)$ and $\tilde{e}_{jm}^k = a_t$ $(t = 3, 2, 1)$, then $\tilde{e}_{im}^k = a_{\min(s,t)}$. For example, C_i is '*absolutely less important*' than C_j (s = 1), C_j is '*less important*' than C_m (t = 3), then C_i is '*absolutely less important*' than C_m.

De-In-Uncertainty rule: If $\tilde{e}_{ij}^k = a_s$ $(s = 4, 5, 6, 7)$, $\tilde{e}_{jm}^k = a_t$ $(t = 3, 2, 1,)$ or '*,' then $\tilde{e}_{im}^k = a_i$ for any $t \le i \le s$ or '*.' For example, C_i is '*more important*' than C_j (s = 5) and C_j is '*much less important*' than C_m (t = 2), then C_i can have any relationship between '*much less important*' and '*more important,*' such as '*equally important* (i = 4)' or '*,' with C_m.

In-De-Uncertainty rule: If $\tilde{e}_{ij}^k = a_s$, $(s = 3, 2, 1)$ or '*,' and $\tilde{e}_{jm}^k = a_t$ $(t = 4, 5, 6, 7)$, then $\tilde{e}_{im}^k = a_i$ for any $s \le i \le t$ or '*.' For example, C_i is '*less important*' than C_j (s = 3) and C_j is '*much more important*' than C_m (t = 6), then C_i can have any relationship between '*less important*' and '*much more important,*' such as '*equally important* (i = 4)' or '*,' with C_m.

Consistent weights w_i^k $(i = 1, 2, \cdots, t)$ for every assessment-criterion can be determined by calculating the geometric mean of each row of the matrix $[\tilde{e}_{ij}^k]_{t\times t}$ where e_{ij}^k $(j = 1, 2, \cdots, i_k)$ is not '*,' and then the resulting fuzzy numbers are normalised and denoted as $\tilde{w}_1^k, \tilde{w}_2^k, \cdots, \tilde{w}_t^k$, where $\tilde{w}_i^k \in F_T^*(R)$ and

$$\tilde{w}_i^k = \frac{w_i^k}{\sum_{i=1}^t w_{i\,0}^{k\,R}}, \quad \text{for } i = 1, 2, \cdots, t;\ k = 1, 2, \cdots, n. \tag{10.3.1}$$

Step 5: Against every selection criterion C_j (j = 1, 2, ..., t), a belief level can be introduced to express the possibility of selecting a solution S_i under the criterion C_j for decision makers P_k. The belief level b_{ij}^k $(i = 1, 2, \cdots, t,\ j = 1, 2, \cdots, m, k = 1,2,\ldots,n)$ belongs to a set of linguistic terms that contain various degrees of preferences required by decision makers P_k $(k = 1,2,\ldots,n)$ under the jth assessment-criterion $(j = 1, 2, \cdots, m)$. The linguistic terms for variable '*preference*' are shown in Table 10.3. Notation '**' can be used to represent that decision makers P_k do not know or could not give a belief level for expressing the preference for a solution S_i under the criterion C_j.

Table 10.3: Linguistic terms for preference belief levels for alternatives

Linguistic terms	Fuzzy numbers
Lowest	b_1
Very Low	b_2
Low	b_3
Medium	b_4
High	b_5
Very high	b_6
Highest	b_7

Step 6: Belief level matrix $\left(b_{ij}^k\right)(k = 1, 2, \cdots, n)$ is aggregated into belief vectors $\left(\overline{b}_j^k\right)$ (j = 1, 2, ..., m, k = 1, 2, ..., n).

$$\overline{b}_j^k = \widetilde{w}_{j_1}^k * b_{jj_1}^k + \widetilde{w}_{j_2}^k * b_{jj_2}^k + \cdots + \widetilde{w}_{j_s}^k * b_{jj_s}^k, \qquad (10.3.2)$$

where $b_{jj_i}^k$ $(i = 1, 2, \cdots, s)$ is not '**.' Based on belief vectors $\left(\overline{b}_j^k\right)$, decision makers P_k (k = 1, 2, ..., n) can make an overall judgment on the alternatives, an individual assessment vector. All individual selection vectors can compose a group of selection matrixes $\left(\overline{b}_j^k\right)$.

Stage 3: *Group aggregation*

Step 7: Each member P_k has been assigned with a weight that is described by a linguistic term $\tilde{v}_k$, $k = 1, 2, \cdots, n$ as shown in Table 10.1. A weight vector is obtained:

$$V = \{\tilde{v}_k, k = 1, 2, \cdots, n\}.$$

The normalised weight of decision makers P_k (k = 1, 2, ..., n) is denoted as

$$\tilde{v}_k^* = \frac{\tilde{v}_k}{\sum_{i=1}^{n} v_{i0}^{R}}, \quad \text{for } k = 1, 2, \cdots, n. \tag{10.3.3}$$

Step 8: Considering the normalised weights of all group members, we can construct a weighted normalised fuzzy decision vector

$$(\tilde{r}_1, \tilde{r}_2, \cdots, \tilde{r}_m) = (\tilde{v}_1^*, \tilde{v}_2^*, \cdots, \tilde{v}_n^*) \begin{pmatrix} \bar{b}_1^1 & \bar{b}_2^1 & \cdots & \bar{b}_m^1 \\ \bar{b}_1^2 & \bar{b}_2^2 & \cdots & \bar{b}_m^2 \\ \vdots & \vdots & \ddots & \vdots \\ \bar{b}_1^n & \bar{b}_2^n & \cdots & \bar{b}_m^n \end{pmatrix}, \tag{10.3.4}$$

where $\tilde{r}_j = \sum_{k=1}^{n} \tilde{v}_k^* \bar{b}_j^k$.

Step 9: In the weighted normalised fuzzy decision vector the elements $\tilde{v}_j$, $j = 1, 2, \cdots, m$, are normalised as positive fuzzy numbers and their ranges belong to the closed interval [0, 1]. We can then define a fuzzy positive-ideal solution (FPIS, r^*) and a fuzzy negative-ideal solution (FNIS, r^-) as:

$$r^* = 1 \quad \text{and} \quad r^- = 0.$$

The positive and negative solution whose distances between each $\tilde{r}_j$ and r^*, $\tilde{r}_j$ and r^- can be calculated as:

$$d_j^* = d(\tilde{r}_j, r^*) \quad \text{and} \quad d_j^- = d(\tilde{r}_j, r^-), \quad j = 1, 2, \cdots, m, \tag{10.3.5}$$

where d(., .) is the distance measurement between two fuzzy numbers.

Step 10: A closeness coefficient is defined to determine the ranking order of all alternatives once the d_j^* and d_j^- of each $\boldsymbol{S}_j$ (j = 1, 2, ..., m) are obtained. The closeness coefficient of each alternative is calculated as:

$$CC_j = \frac{1}{2}\left(d_j^- + (1 - d_j^*)\right), \quad j = 1, 2, \cdots, m. \tag{10.3.6}$$

The alternative S_j that corresponds to *Max* (CC_j, j=1, 2, …, m) is the best satisfactory solution of the decision group, and the top N issues (alternatives) that correspond to the top N higher raking CC_j are the critical issues to consider for the decision problem.

10.4 A Case-Based Example

Strategic planning must include an assessment of the organisation's situation. It stresses the importance of focusing on the future within the context of an ever-changing environment. A key component of an organisation's situation assessment is the evaluation of effectiveness and efficiency of its strategies. This evaluation will provide data about whether to continue or discontinue each program or strategy, maintain it at its existing level, expand or change its direction, market it aggressively, and so on. Most business strategy assessments focus on both outcome and process. Outcome evaluation looks at whether a project achived its planned results. Process evaluation looks at internal project management, both staff performance and the extent to which the project in successfully implemented. The strategy assessment can be based on a set of criteria that involve quatitative and/or qialitative data. Quantitive data consists of fact-based information such as record review, describtive statisitics, and examinations results. It is more easily collected but less easily disputed because it translates experience into quanlitifiable data that can be counted, compared, and measured. Qualitative data consists of what people think the programs based on observations informal feedback, surveys. Skills at assessing business situation and then being proactive in responding to that situation (*i.e.*, strategic planning) determines how to effectively identify critical issues, deal with their situation and achive business goals.

As a consequence, the situation assessment outlines the process of identifying the issues, gathering decision makers' perceptions needed to make an explicit evaluation of organisational strategies, and analysing the impact of the strategy on clients and other business aspects through various assessments. Often a group of people participate in an organisational situation assessment with their personal opinions and information. At the conclusion of the situation assessment, strategic planners (decision group members) will have quality information about which are the most critical issues the organisation needs to deal with in the strategic planning process. We will apply the intelligent FMCGDM method to such a business situation assessment.

Suppose an executive group consists of three members P_1, P_2 and P_3 to participate assessing their company's situation through identifying critical and urgent key issues for the company's business development. The three members come from three functional departments of the company and have collected related environment information respectively, but their weights are same. Their weights, preferences for raised alternatives and judgements for proposed assessment criteria can be described by linguistic terms, as shown in Tables 10.4, 10.5, and 10.6, respectively.

Table 10.4: Linguistic terms and related fuzzy numbers for weights of decision makers

Linguistic terms	Fuzzy numbers
Normal	$\bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{16\lambda+9}}{10}, \frac{\sqrt{49-24\lambda}}{10}]$
Important	$\bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{24\lambda+25}}{10}, \frac{\sqrt{81-32\lambda}}{10}]$
More important	$\bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{32\lambda+49}}{10}, \frac{\sqrt{100-19\lambda}}{10}]$
Most important	$\bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{19\lambda+81}}{10}, 1]$

Table 10.5: Linguistic terms and related fuzzy numbers for comparison scales of criteria

Linguistic terms	Fuzzy numbers
Absolutely less important	$\bigcup_{\lambda\in[0,1]} \lambda[0, \frac{\sqrt{1-\lambda}}{10}]$
Much less important	$\bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{\lambda}}{10}, \frac{\sqrt{9-8\lambda}}{10}]$
Less important	$\bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{8\lambda+1}}{10}, \frac{\sqrt{25-16\lambda}}{10}]$
Equally important	$\bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{16\lambda+9}}{10}, \frac{\sqrt{49-24\lambda}}{10}]$
More important	$\bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{24\lambda+25}}{10}, \frac{\sqrt{81-32\lambda}}{10}]$
Much more important	$\bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{32\lambda+49}}{10}, \frac{\sqrt{100-19\lambda}}{10}]$
Absolutely more important	$\bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{19\lambda+81}}{10}, 1]$

Table 10.6: Linguistic terms and related fuzzy numbers for belief levels of preferences

Linguistic terms	Fuzzy numbers
Lowest	$\bigcup_{\lambda\in[0,1]} \lambda[0, \frac{\sqrt{1-\lambda}}{10}]$
Very low	$\bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{\lambda}}{10}, \frac{\sqrt{9-8\lambda}}{10}]$
Low	$\bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{8\lambda+1}}{10}, \frac{\sqrt{25-16\lambda}}{10}]$
Medium	$\bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{16\lambda+9}}{10}, \frac{\sqrt{49-24\lambda}}{10}]$
High	$\bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{24\lambda+25}}{10}, \frac{\sqrt{81-32\lambda}}{10}]$
Very high	$\bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{32\lambda+49}}{10}, \frac{\sqrt{100-19\lambda}}{10}]$
Highest	$\bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{19\lambda+81}}{10}, 1]$

In the three tables, all linguistic terms are described by fuzzy numbers. To have a good understanding, these terms are displayed by figures. For examples, Fig. 10.2 shows the linguistic term '*Medium,*' and Fig. 10.3 shows the linguistic term '*Very high*.'

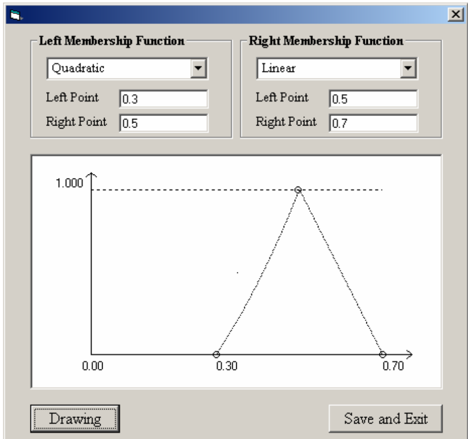


Fig. 10.2: Linguistic term '*Medium*'

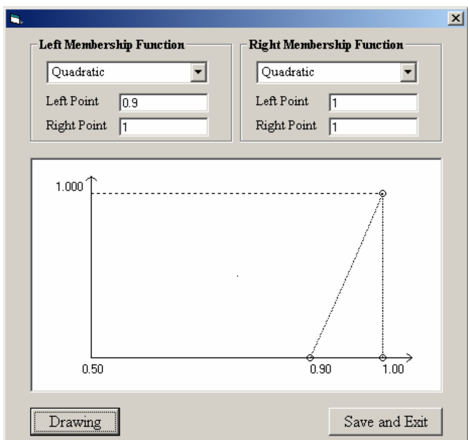


Fig. 10.3: Linguistic term '*Very high*'

The problem-solving process by using the proposed FMCGDM method is described as follows.

Stage 1: *Alternatives, assessment-criteria, and individual weights generation*

Step 1: To initiate the assessment, group members first list all issues/strategies related to business strategies to explore. Each member proposes one or more possible critical issues/strategies they concerned for the business situation. These issues/strategies are listed as alternative $S^{\#} = \{S_1^{p_1}, S_2^{p_1}, ... S_{m_{p1}}^{p_1} ...... S_1^{p_3}, S_2^{p_3} ...... S_{m_{p3}}^{p_3}\}$, where $s_j^{p_i}$ is the *j*th issue proposed by the member p_i. Through merging some similar issues an alternative list is finally determined for the decision group $S = \{S_1, S_2, S_4, S_4\}$:

S_1: developing new products *(New-prod);*

S_2: increasing international market development investigation *(Int-market);*

S_3: reduce product storage costs *(Stro-cost);* and

S_4: re-structural customer relationship management department *(Cust-relation).*

Step 2: These members have different concerns and opinions for assessing and raking these proposed alternatives. The group must assess each alternative by considering how urgent and critical the issue is and how effective will the issue enable the company to meet its objectives. Based on these alternatives, each of the three group members proposes a few assessment-criteria for assessing these alternatives. Through summarising concerns some similar criteria are merged. Finally, five assessment-criteria C_1, C_2, C_3, C_4, and C_5 are determined for the group:

C_1: internal and external stakeholders' perceptions about these issues/strategies (*Perception*)

C_2: the impact of a new program, such as a new product or new customer relationship management departments, on clients (*Impact*);

C_3: a program' cost and benefit (*Cost-benefit*);

C_4: a new program's competitive analysis (*Competitive*); and

C_5: defining previous implied strategies (*Previous*)

As the three members have the same weight in the group, all are assigned with '*normal.*'

Stage 2: *Individual preference generation*

Step 4: Each member gives an individual judgment for the five assessment-criteria by using pairwise comparison. Based on Tables 10.1-10.6, three pairwise comparison matrices E^1, E^2, and E^3, are thus established for the three members.

$$E^1 = E^2 = E^3 = \begin{pmatrix} EI & EI & * & * & EI \\ EI & EI & * & EI & * \\ * & * & EI & * & * \\ * & EI & * & EI & EI \\ EI & * & * & EI & EI \end{pmatrix} = \begin{pmatrix} a_4 & a_4 & * & * & a_4 \\ a_4 & a_4 & * & a_4 & * \\ * & * & a_4 & * & * \\ * & a_4 & * & a_4 & a_4 \\ a_4 & * & * & a_4 & a_4 \end{pmatrix}.$$

By using the linguistic inference rules, we get finalised pairwise comparison matrices to express the possibility of selecting a solution under certain criteria.

$$E^1 = E^2 = E^3 = \begin{pmatrix} EI & EI & * & EI & EI \\ EI & EI & * & EI & * \\ * & * & EI & * & * \\ EI & EI & * & EI & EI \\ EI & * & * & EI & EI \end{pmatrix} = \begin{pmatrix} a_4 & a_4 & * & a_4 & a_4 \\ a_4 & a_4 & * & a_4 & * \\ * & * & a_4 & * & * \\ a_4 & a_4 & * & a_4 & a_4 \\ a_4 & * & * & a_4 & a_4 \end{pmatrix}.$$

Through computing the geometric mean of each row of these matrices, the normalised resulting numbers are obtained. As

$$\begin{pmatrix} w_1^1 \\ w_2^1 \\ w_3^1 \\ w_4^1 \\ w_5^1 \end{pmatrix} = \begin{pmatrix} w_1^2 \\ w_2^2 \\ w_3^2 \\ w_4^2 \\ w_5^2 \end{pmatrix} = \begin{pmatrix} w_1^3 \\ w_2^3 \\ w_3^3 \\ w_4^3 \\ w_5^3 \end{pmatrix} = \begin{pmatrix} \sqrt[4]{a_4^4} \\ \sqrt[3]{a_4^3} \\ a_4 \\ \sqrt[4]{a_4^4} \\ \sqrt[3]{a_4^3} \end{pmatrix} = \begin{pmatrix} a_4 \\ a_4 \\ a_4 \\ a_4 \\ a_4 \end{pmatrix} = \begin{pmatrix} \bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{16\lambda+9}}{10}, \frac{\sqrt{49-24\lambda}}{10}] \\ \bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{16\lambda+9}}{10}, \frac{\sqrt{49-24\lambda}}{10}] \\ \bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{16\lambda+9}}{10}, \frac{\sqrt{49-24\lambda}}{10}] \\ \bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{16\lambda+9}}{10}, \frac{\sqrt{49-24\lambda}}{10}] \\ \bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{16\lambda+9}}{10}, \frac{\sqrt{49-24\lambda}}{10}] \end{pmatrix},$$

and $\sum_{i=1}^{5} w_{i\,0}^{1R} = \sum_{i=1}^{5} w_{i\,0}^{2R} = \sum_{i=1}^{5} w_{i\,0}^{3R} = 3.5$, we have

$$\begin{pmatrix} \tilde{w}_1^1 \\ \tilde{w}_2^1 \\ \tilde{w}_3^1 \\ \tilde{w}_4^1 \\ \tilde{w}_5^1 \end{pmatrix} = \begin{pmatrix} \tilde{w}_1^2 \\ \tilde{w}_2^2 \\ \tilde{w}_3^2 \\ \tilde{w}_4^2 \\ \tilde{w}_5^2 \end{pmatrix} = \begin{pmatrix} \tilde{w}_1^3 \\ \tilde{w}_2^3 \\ \tilde{w}_3^3 \\ \tilde{w}_4^3 \\ \tilde{w}_5^3 \end{pmatrix} = \frac{1}{3.5} \begin{pmatrix} a_4 \\ a_4 \\ a_4 \\ a_4 \\ a_4 \end{pmatrix}.$$

Step 5: Three belief level matrices are obtained, where b_{ij}^{k} expresses the possibility of selecting S_i under the assessment-criteria C_i by the group member P_k.

$$\begin{pmatrix} b_{11}^1 & b_{12}^1 & b_{13}^1 & b_{14}^1 & b_{15}^1 \\ b_{21}^1 & b_{22}^1 & b_{23}^1 & b_{24}^1 & b_{25}^1 \\ b_{31}^1 & b_{32}^1 & b_{33}^1 & b_{34}^1 & b_{35}^1 \\ b_{41}^1 & b_{42}^1 & b_{43}^1 & b_{44}^1 & b_{45}^1 \end{pmatrix} = \begin{pmatrix} M & VL & ** & ** & ** \\ VH & M & ** & ** & ** \\ ** & ** & M & VL & ** \\ ** & VL & ** & ** & M \end{pmatrix} = \begin{pmatrix} b_4 & b_1 & ** & ** & ** \\ b_7 & b_4 & ** & ** & ** \\ ** & ** & b_4 & b_1 & ** \\ ** & b_1 & ** & ** & b_4 \end{pmatrix},$$

$$\begin{pmatrix} b_{11}^2 & b_{12}^2 & b_{13}^2 & b_{14}^2 & b_{15}^2 \\ b_{21}^2 & b_{22}^2 & b_{23}^2 & b_{24}^2 & b_{25}^2 \\ b_{31}^2 & b_{32}^2 & b_{33}^2 & b_{34}^2 & b_{35}^2 \\ b_{41}^2 & b_{42}^2 & b_{43}^2 & b_{44}^2 & b_{45}^2 \end{pmatrix} = \begin{pmatrix} M & VL & ** & ** & ** \\ VH & ** & M & ** & ** \\ ** & ** & VL & M & ** \\ ** & M & ** & ** & VL \end{pmatrix} = \begin{pmatrix} b_4 & b_1 & ** & ** & ** \\ b_7 & ** & b_4 & ** & ** \\ ** & ** & b_1 & b_4 & ** \\ ** & b_4 & ** & ** & b_1 \end{pmatrix},$$

$$\begin{pmatrix} b_{11}^3 & b_{12}^3 & b_{13}^3 & b_{14}^3 & b_{15}^3 \\ b_{21}^3 & b_{22}^3 & b_{23}^3 & b_{24}^3 & b_{25}^3 \\ b_{31}^3 & b_{32}^3 & b_{33}^3 & b_{34}^3 & b_{35}^3 \\ b_{41}^3 & b_{42}^3 & b_{43}^3 & b_{44}^3 & b_{45}^3 \end{pmatrix} = \begin{pmatrix} VL & M & ** & ** & ** \\ M & ** & VH & ** & ** \\ ** & ** & M & VL & ** \\ ** & M & ** & ** & VL \end{pmatrix} = \begin{pmatrix} b_1 & b_4 & ** & ** & ** \\ b_4 & ** & b_7 & ** & ** \\ ** & ** & b_4 & b_1 & ** \\ ** & b_4 & ** & ** & b_1 \end{pmatrix}.$$

Step 6: The three belief vectors are then generated through aggregating the above belief level matrices:

$$\bar{b}_1^1 = \tilde{w}_1^1 b_{11}^1 + \tilde{w}_2^1 b_{12}^1 + \tilde{w}_3^1 b_{13}^1 + \tilde{w}_4^1 b_{14}^1 + \tilde{w}_5^1 b_{15}^1 = \frac{1}{3.5}(a_4{}^2 + a_4 a_1),$$

$$\bar{b}_2^1 = \tilde{w}_1^1 b_{21}^1 + \tilde{w}_2^1 b_{22}^1 + \tilde{w}_3^1 b_{23}^1 + \tilde{w}_4^1 b_{24}^1 + \tilde{w}_5^1 b_{25}^1 = \frac{1}{3.5}(a_4{}^2 + a_4 a_7),$$

$$\bar{b}_3^1 = \tilde{w}_1^1 b_{31}^1 + \tilde{w}_2^1 b_{32}^1 + \tilde{w}_3^1 b_{33}^1 + \tilde{w}_4^1 b_{34}^1 + \tilde{w}_5^1 b_{35}^1 = \frac{1}{3.5}(a_4{}^2 + a_4 a_1),$$

$$\bar{b}_4^1 = \tilde{w}_1^1 b_{41}^1 + \tilde{w}_2^1 b_{42}^1 + \tilde{w}_3^1 b_{43}^1 + \tilde{w}_4^1 b_{44}^1 + \tilde{w}_5^1 b_{45}^1 = \frac{1}{3.5}(a_4{}^2 + a_4 a_1),$$

$$\bar{b}_1^2 = \tilde{w}_1^2 b_{11}^2 + \tilde{w}_2^2 b_{12}^2 + \tilde{w}_3^2 b_{13}^2 + \tilde{w}_4^2 b_{14}^2 + \tilde{w}_5^2 b_{15}^2 = \frac{1}{3.5}(a_4{}^2 + a_4 a_1),$$

$$\bar{b}_2^2 = \tilde{w}_1^2 b_{21}^2 + \tilde{w}_2^2 b_{22}^2 + \tilde{w}_3^2 b_{23}^2 + \tilde{w}_4^2 b_{24}^2 + \tilde{w}_5^2 b_{25}^2 = \frac{1}{3.5}(a_4{}^2 + a_4 a_7),$$

$$\bar{b}_3^2 = \tilde{w}_1^2 b_{31}^2 + \tilde{w}_2^2 b_{32}^2 + \tilde{w}_3^2 b_{33}^2 + \tilde{w}_4^2 b_{34}^2 + \tilde{w}_5^2 b_{35}^2 = \frac{1}{3.5}(a_4{}^2 + a_4 a_1),$$

$$\bar{b}_4^2 = \tilde{w}_1^2 b_{41}^2 + \tilde{w}_2^2 b_{42}^2 + \tilde{w}_3^2 b_{43}^2 + \tilde{w}_4^2 b_{44}^2 + \tilde{w}_5^2 b_{45}^2 = \frac{1}{3.5}(a_4{}^2 + a_4 a_1),$$

$$\bar{b}_1^3 = \tilde{w}_1^3 b_{11}^3 + \tilde{w}_3^3 b_{13}^3 + \tilde{w}_3^3 b_{13}^3 + \tilde{w}_4^3 b_{14}^3 + \tilde{w}_5^3 b_{15}^3 = \frac{1}{3.5}(a_4{}^2 + a_4 a_1),$$

$$\overline{b}_2^3 = \tilde{w}_1^3 b_{21}^3 + \tilde{w}_2^3 b_{22}^3 + \tilde{w}_3^3 b_{23}^3 + \tilde{w}_4^3 b_{24}^3 + \tilde{w}_5^3 b_{25}^3 = \frac{1}{3.5}(a_4^{\ 2} + a_4 a_7),$$

$$\overline{b}_3^3 = \tilde{w}_1^3 b_{31}^3 + \tilde{w}_2^3 b_{32}^3 + \tilde{w}_3^3 b_{33}^3 + \tilde{w}_4^3 b_{34}^3 + \tilde{w}_5^3 b_{35}^3 = \frac{1}{3.5}(a_4^{\ 2} + a_4 a_1),$$

$$\overline{b}_4^3 = \tilde{w}_1^3 b_{41}^3 + \tilde{w}_2^3 b_{42}^3 + \tilde{w}_3^3 b_{43}^3 + \tilde{w}_4^3 b_{44}^3 + \tilde{w}_5^3 b_{45}^3 = \frac{1}{3.5}(a_4^{\ 2} + a_4 a_1).$$

Stage 3: *Group aggregation*

Step 7: As $v_1 = v_2 = v_3 = c_1$, and $\sum_{i=1}^{3} v_{i0}^R = 2.1$ we have

$$v_1^* = v_2^* = v_3^* = \frac{1}{2.1}c_1 = \frac{1}{2.1}a_4.$$

Step 8: We construct a weighted normalised fuzzy decision vector:

$$\tilde{r}_1 = v_1^*\overline{b}_1^1 + v_2^*\overline{b}_1^2 + v_3^*\overline{b}_1^3 = \frac{3}{3.5\times 2.1}a_4^2(a_4 + a_1)$$

$$= \bigcup_{\lambda\in[0,1]} \lambda\left[\frac{3(16\lambda+9)\sqrt{16\lambda+9}}{7350}, \frac{3(49-24\lambda)(\sqrt{49-24\lambda}+\sqrt{1-\lambda})}{7350}\right].$$

$$\tilde{r}_2 = v_1^*\overline{b}_2^1 + v_2^*\overline{b}_2^2 + v_3^*\overline{b}_2^3 = \frac{3}{3.5\times 2.1}a_4^2(a_4 + a_7)$$

$$= \bigcup_{\lambda\in[0,1]} \lambda\left[\frac{3(16\lambda+9)(\sqrt{16\lambda+9}+\sqrt{19\lambda+81})}{7350}, \frac{3(49-24\lambda)(\sqrt{49-24\lambda}+10)}{7350}\right].$$

$$\tilde{r}_3 = v_1^*\overline{b}_3^1 + v_2^*\overline{b}_3^2 + v_3^*\overline{b}_3^3 = \frac{3}{3.5\times 2.1}a_4^2(a_4 + a_1)$$

$$= \bigcup_{\lambda\in[0,1]} \lambda\left[\frac{3(16\lambda+9)\sqrt{16\lambda+9}}{7350}, \frac{3(49-24\lambda)(\sqrt{49-24\lambda}+\sqrt{1-\lambda})}{7350}\right].$$

$$\tilde{r}_4 = v_1^*\overline{b}_4^1 + v_2^*\overline{b}_4^2 + v_3^*\overline{b}_4^3 = \frac{3}{3.5\times 2.1}a_4^2(a_4 + a_1)$$

$$= \bigcup_{\lambda\in[0,1]} \lambda\left[\frac{3(16\lambda+9)\sqrt{16\lambda+9}}{7350}, \frac{3(49-24\lambda)(\sqrt{49-24\lambda}+\sqrt{1-\lambda})}{7350}\right].$$

Step 9: We calculate distances between positive and negative solutions for these four alternatives:

$$d_1^* = d(\tilde{r}_1, r^*) = \left(\int_0^1 \frac{1}{2}\left[\left(\frac{3(16\lambda+9)\sqrt{16\lambda+9}}{7350} - 1\right)^2\right.\right.$$

$$+\left(\frac{3(49-24\lambda)(\sqrt{49-24\lambda}+\sqrt{1-\lambda})}{7350}-1\right)^2\Bigg]d\lambda\Bigg)^{\frac{1}{2}} = 0.9344$$

$$d_2^* = d(\tilde{r}_2, r^*) = \left(\int_0^1 \frac{1}{2}\left[\left(\frac{3(16\lambda+9)(\sqrt{16\lambda+9}+\sqrt{19\lambda+81})}{7350}-1\right)^2\right.\right.$$

$$+\left(\frac{3(49-24\lambda)(\sqrt{49-24\lambda}+10)}{7350}-1\right)^2\Bigg]d\lambda\Bigg)^{\frac{1}{2}} = 0.8344$$

$$d_3^* = d(\tilde{r}_3, r^*) = \left(\int_0^1 \frac{1}{2}\left[\left(\frac{3(16\lambda+9)\sqrt{16\lambda+9}}{7350}-1\right)^2\right.\right.$$

$$+\left(\frac{3(49-24\lambda)(\sqrt{49-24\lambda}+\sqrt{1-\lambda})}{7350}-1\right)^2\Bigg]d\lambda\Bigg)^{\frac{1}{2}} = 0.9344$$

$$d_4^* = d(\tilde{r}_4, r^*) = \left(\int_0^1 \frac{1}{2}\left[\left(\frac{3(16\lambda+9)\sqrt{16\lambda+9}}{7350}-1\right)^2\right.\right.$$

$$+\left(\frac{3(49-24\lambda)(\sqrt{49-24\lambda}+\sqrt{1-\lambda})}{7350}-1\right)^2\Bigg]d\lambda\Bigg)^{\frac{1}{2}} =0.9344$$

$$d_1^- = d(\tilde{r}_1, r-) = \left(\int_0^1 \frac{1}{2}\left[\left(\frac{3(16\lambda+9)\sqrt{16\lambda+9}}{7350}\right)^2\right.\right.$$

$$+\left(\frac{3(49-24\lambda)(\sqrt{49-24\lambda}+\sqrt{1-\lambda})}{7350}\right)^2\Bigg]d\lambda\Bigg)^{\frac{1}{2}} = 0.0799$$

$$d_2^- = d(\tilde{r}_2, r-) = \left(\int_0^1 \frac{1}{2}\left[\left(\frac{3(16\lambda+9)(\sqrt{16\lambda+9}+\sqrt{19\lambda+81})}{7350}\right)^2\right.\right.$$

$$+\left(\frac{3(49-24\lambda)(\sqrt{49-24\lambda}+10)}{7350}\right)^2\Bigg]d\lambda\Bigg)^{\frac{1}{2}} = 0.1907$$

$$d_3^- = d(\tilde{r}_3, r-) = \left(\int_0^1 \frac{1}{2}\left[\left(\frac{3(16\lambda+9)\sqrt{16\lambda+9}}{7350}\right)^2\right.\right.$$

$$+\left(\frac{3(49-24\lambda)(\sqrt{49-24\lambda}+\sqrt{1-\lambda})}{7350}\right)^2\Bigg]d\lambda\Bigg)^{\frac{1}{2}} = 0.0799$$

$$d_4^- = d(\tilde{r}_4, r-) = \left(\int_0^1 \frac{1}{2}\left[\left(\frac{3(16\lambda+9)\sqrt{16\lambda+9}}{7350}\right)^2\right.\right.$$

$$+\left(\frac{3(49-24\lambda)(\sqrt{49-24\lambda}+\sqrt{1-\lambda})}{7350}\right)^2\Bigg]d\lambda\Bigg)^{\frac{1}{2}} = 0.0799$$

Step 10: Finally, for assessing or ranking these alternatives we calculate the closeness coefficient of each alternative:

$$CC_1 = \frac{1}{2}(d_1^- + (1-d_1^*)) = \frac{1}{2}(0.0799 + (1-0.9344)) = 0.0728$$

$$CC_2 = \frac{1}{2}(d_2^- + (1-d_2^*)) = \frac{1}{2}(0.1907 + (1-0.8344)) = 0.1782$$

$$CC_3 = \frac{1}{2}(d_3^- + (1-d_3^*)) = \frac{1}{2}(0.0799 + (1-0.9344)) = 0.0788$$

$$CC_4 = \frac{1}{2}(d_4^- + (1-d_4^*)) = \frac{1}{2}(0.0799 + (1-0.9344)) = 0.0728.$$

The result shows the ranking of the four alternatives are $S_{2,} S_{1,} S_{3,} S_4$ as $CC_2 > CC_3 > CC_1 = CC_4$. The alternative S_2 can be selected as the most critical issue for the business's situation. That is, '*increasing international market development investigation*' is identified as the key strategy for the current business situation. The result aggregates maximally all group members' roles, judgments and preferences for a solution in whole.

10.5 Summary

Uncertain factors often affect a group decision making. In this chapter, we identify three main uncertain factors, namely, makers' roles, preferences for alternatives, and judgments for assessment-criteria. We present an intelligent FMCGDM method to deal with the three uncertain factors and generate a group satisfactory decision. The proposed method has been implemented in a Web-based fuzzy group DSS, which will be presented in Chapter 11. More applications of the system will be presented in Chapter 15.

This page intentionally left blank

Chapter 11

A Web-Based Fuzzy Group DSS

Following the previous chapter, we will present a Web-based fuzzy group DSS (FGDSS). This system allows decision makers distributed in different locations to participate in a group decision-making activity through the Web. It manages the group decision process through criteria generation, alternative evaluation, opinion interaction, and decision aggregation using linguistic terms. We first outline the main features of the Web-based FGDSS, and then present this system's configuration and working process. Finally, we give two examples to demonstrate the system. To help readers use the system, we have also developed an off-line version of the system in the attached CD.

11.1 System Features

Decision group members may be distributed geographically in different locations. Nowadays, the Web is often acting as a mechanism for the support of decision making in geographically distributed organisations. Group decision support systems (GDSS) can therefore be implemented as a kind of Web-based services, and have been moving to a global environment. With the advance of Web technology, Web-based GDSS have been applying in widespread decision activities with the unified graphical user interface.

Web-based GDSS and FGDSS have basically four features.

(1) Supporting asynchronous communications among group members

An important feature provided by GDSS or FGDSS is to support the interpersonal communication and coordination among group members.

This feature aims at achieving a common understanding of the issues revealed and arriving at a group satisfactory decision. The communication and coordination activities of group members are facilitated by technologies that can be characterised along the three continua of time, space, and level of group support. By using the Web, group members can communicate asynchronously by emails, bulletin board systems, Internet newsgroups, and specific Web-based GDSS.

(2) Extending application ranges of traditional GDSS

Web-based GDSS or FGDSS can use the Web environment as a development and delivery platform. More recently, both e-business and e-government are increasing the demands for more online data analysis and decision support. This Web platform lends Web-based GDSS or FGDSS to have widespread use and adoption in organisations. At the same time, organisations can use Web-based GDSS or FGDSS to provide group decision support capabilities to managers over a proprietary Intranet, to customers and suppliers over an Extranet, or to any stakeholder over the global Internet.

(3) Reducing technological barriers

Web-based GDSS or FGDSS have reduced technological barriers and made less costly to develop and delivery themselves and provide decision-relevant information. Traditionally, GDSS or FGDSS required specific software on user computers, specific locations to set up, and users needed proper training to learn how to use a GDSS. Thanks to the Web platform, the use of GDSS can overcome these shortcomings. Furthermore, GDSS or FGDSS have a convenient and graphical user interface with visualisation possibilities and are automatically available to many decision makers.

(4) Improving decision making performance

Web-based GDSS and FGDSS can increase the range and depth of information access, and therefore solve group decision problems more effectively. Decision making, especially at upper management levels, relies heavily on data sources outside the organisations. They integrated with Web mining and related Web intelligence techniques allow decision makers to access internal and external data sources, such as competitors' product/service offerings, during the decision making process. In particular, under an uncertain environment, the organisations will find that Web-based FGDSS can more effectively assist their decision groups in making organisational strategic decisions where group members are distributed in different locations and with linguistic terms.

It is evident that Web-based GDSS and FGDSS can extend the applications of traditional GDSS and FGDSS to support more effectively organisational decision making. The development of a Web-based FGDSS will extend the current results by proving the ability of dealing with linguistic terms in a distributed group decision activity.

We developed two versions of FGDSS, one is Web-based online version, and another is off-line. We mainly describe the design of the Web-based version in this chapter, but to help readers using this system we put the off-line version in the attached CD, which can be used in a PC.

11.2 System Configuration

We adopt the client/server pattern in the Web-based FGDSS. At the client side, all group members access the system with the browser via the Web. The interface that is generated on the server side will be presented on the client side, and group members can also interact with the server for getting and supplying information by the browser.

At the server side, the Web server manages all Web pages of the system, traces user information, and provides simultaneously services to multiple group members through sessions, applications, and coking facilities. All Web pages developed in the Web-based FGDSS, for interacting dynamically with group members in solving their decision problems with linguistic terms, are created on the Web server. By using a server side application program, the Web server can manage and implement client tasks.

There are four components on the Web server: (1) Presentation, (2) Aggregation, (3) Model management, and (4) Data management. In addition, there are three bases: (a) Database, (b) Method-base, and (c)

Model-base. These bases are linked to the corresponding management components respectively.

The system is developed and implemented mainly in JSP combined with HTML and JavaScript. The typical characteristic of the JSP is that it can create dynamic web pages based on the different requests from the clients. For the system, when receiving a request from the client side, the web server will relay the request to the presentation component. The presentation component also delivers it to the management component or aggregation component. When the presentation component receives the result from the corresponding component, it will create an HTML file dynamically, and the web server sends this HTML file back to the client. Fig. 11.1 shows the structure of the Web-based FGDSS.

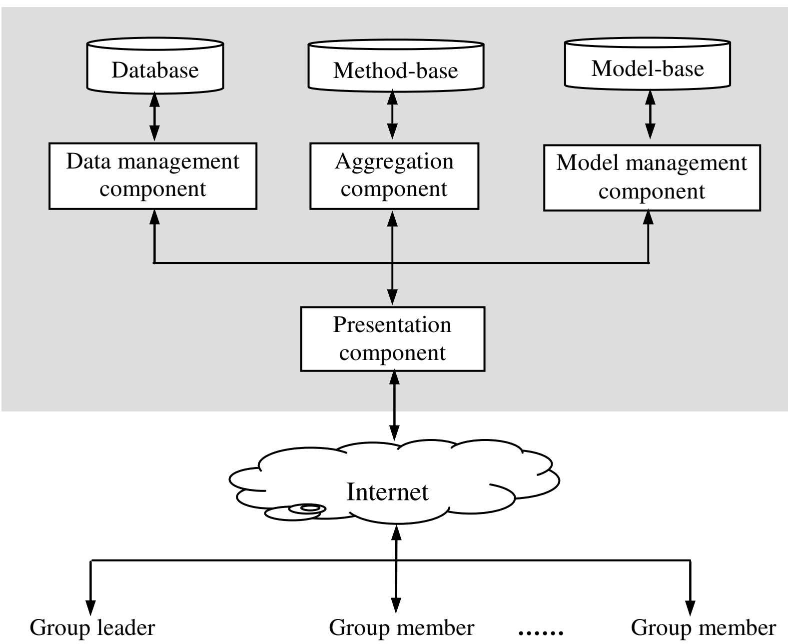


Fig. 11.1: The structure of the Web-based FGDSS

11.3 System Working Process

The working process of a decision group with the Web-based FGDSS is with five main steps. Fig. 11.2 shows the working process of Web-based fuzzy group decision making.

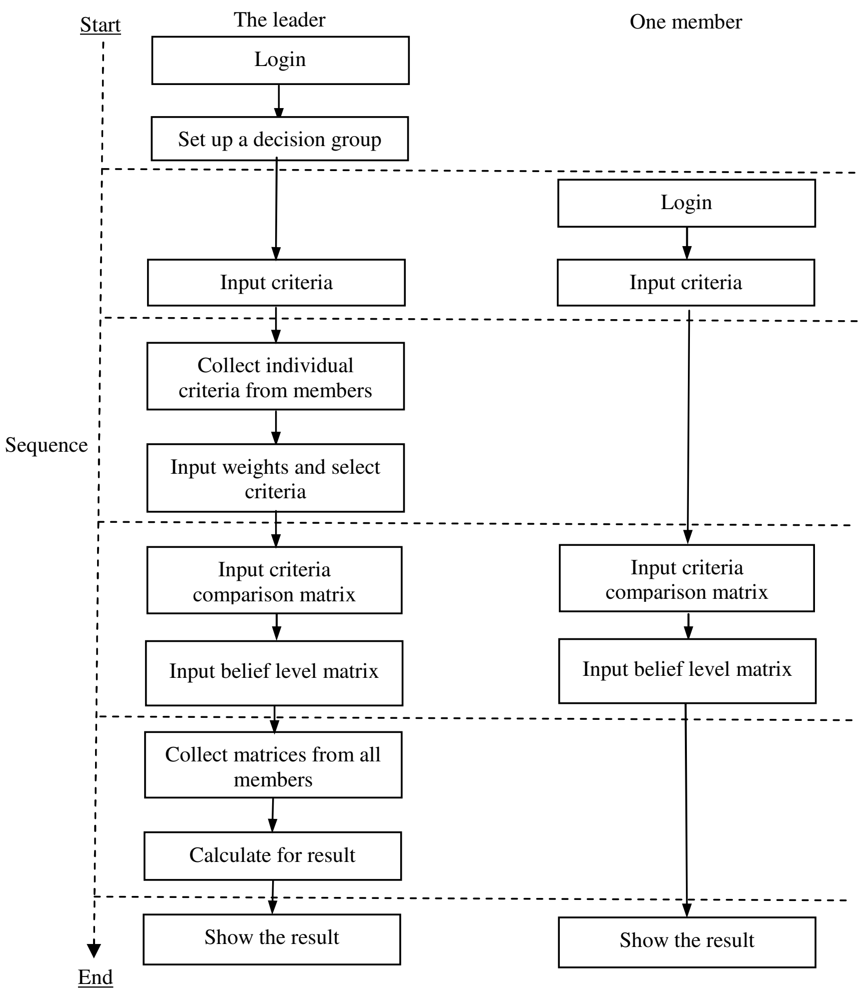


Fig. 11.2: Working process of the Web-based FGDM

The process can be further described by the following steps:

Step1: Setting up a decision group

The group leader first uses the browser to login to the system as shown in Fig. 11.3, and then defines a decision group as shown in Fig. 11.4 and Fig. 11.5 including:

- The title of the group;
- The problem description;
- The number of group members;
- The number of the alternatives; and
- The details of alternatives.

The server checks the group title assigned by the group leader. If the group title is valid, the server registers the decision group in the database and sends an approval to the client side.

Step 2: Input criteria by all group members

After the group leader sets up a decision group, other members can login to the group similar as Fig.11.6. Then the group information including the alternatives will be fetched from the database and sent to the client side by the server. Based on these alternatives, each group member including the group leader proposes some criteria as shown in Fig. 11.7 and Fig. 11.8 for selecting an alternative as the group satisfactory solution. All proposed criteria are then collected by the server application.

Step 3: Choose the top-t criteria and assign weights to group members

Referring to the criteria received from all members, the group leader chooses the top-t criteria as the assessment-criteria for the decision problem in the group. As group members play different roles, the leader will assign weights, described by linguistic terms, to all group members as shown in Fig. 11.9. All data about the top-t assessment-criteria and member's weights will be sent to the server, and then to the database server for its storage.

For other group members at the moment, as shown in Fig. 11.10, they will just be waiting until the leader sends the top-t criteria and weights for all group members back to the web server for a further procedure.

Step 4: Fill the criteria comparison matrix

Based on the assessment-criteria and alternatives received, each group member will fill a pairwise comparison matrix of the relative importance of these criteria as shown in Fig. 11.11.

Step 5: Fill the belief level matrix

Each group member will fill a belief level matrix to express the possibility of selecting a solution under some criteria as shown in Fig. 11.11.

Step 6: Generate the final result of the group decision-making problem

Once group members' two matrices are received, the server application first corrects the inconsistence of each pairwise comparison matrix of the assessment-criteria based on linguistic inference rules, then calculates the belief level matrices, the belief vector, the normalised weights of group members, the weighted normalised fuzzy decision vector, and the closeness coefficients of all alternatives consecutively. Finally, the web server constructs a final group decision page where the most satisfactory group solution, which is corresponding to the maximum closeness coefficient, is displayed to all the group members.

11.4 Case-Based Examples

Example 1 (*using Web-based FGDSS*): *Course software evaluation*

A department of a university tries to determine which online course software to be used in its teaching task. Four course softwares are available from four education consulting firms. Each has its advantages and disadvantages. The software S_1, S_2, S_3, and S_4 are as four alternatives for the department. The decision group consists of three members: *Peter*, *David*, and *Kim*, and *Peter* is the group leader. The three members have different opinions for selecting which course software. The group must evaluate each of software with its consulting firm by considering how to meet the department's teaching objectives.

Following the working process of a decision group using the Web-based FGDSS, the problem described above can be solved as follows:

Step 1: First, the group leader *Peter* logins to the system as shown in Fig. 11.3 and defines a decision making group. In Fig. 11.4, the number of group members is set to 3, and the number of alternatives is set to 4. The alternatives are entered as in Fig. 11.5.

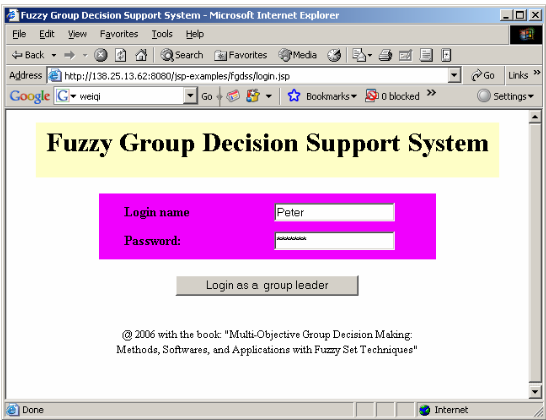


Fig. 11.3: Web page for a group leader to login

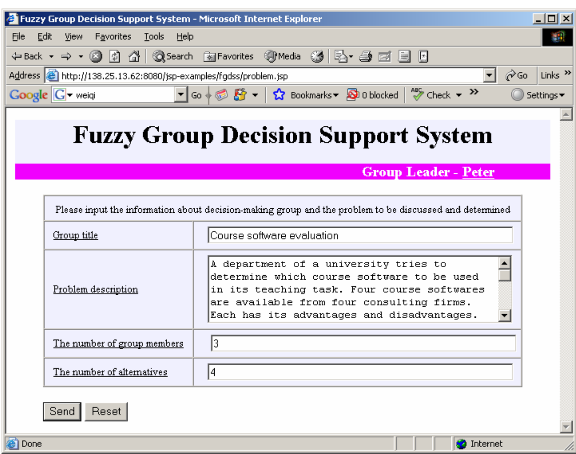


Fig. 11.4: Web page for a group leader to define a group

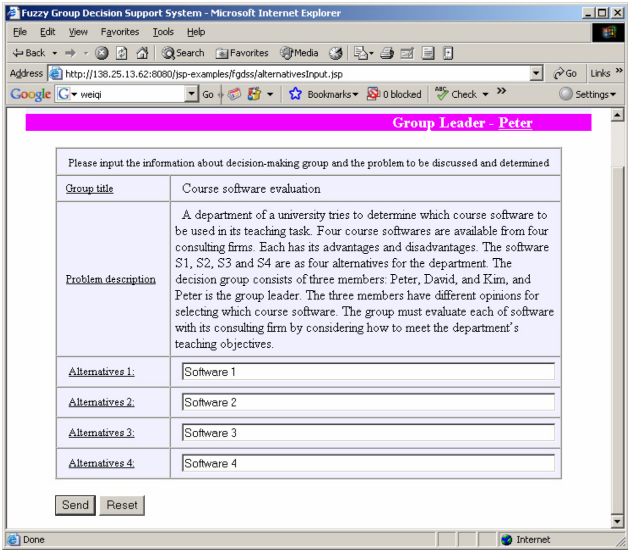


Fig. 11.5: Web page for a group leader to input alternatives

Step 2: After *Peter* has set up the decision group, other members can login to the group as shown in Fig. 11.6. Totally, three members join the decision making group.

Based on the four proposals (alternatives), three group members propose several criteria. Suppose *Peter* proposes '*Price*' and '*Development time*' as criteria for selecting a satisfactory firm from the four candidates shown in Fig. 11.7. *David* proposes '*Experience*' and '*Quality*' as criteria as shown in Fig. 11.8, and Kim proposes '*Service*' and '*Cost*,' as criteria.

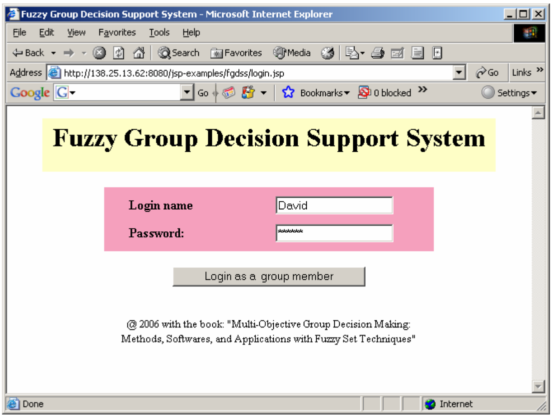


Fig. 11.6: Web page for a group member to login

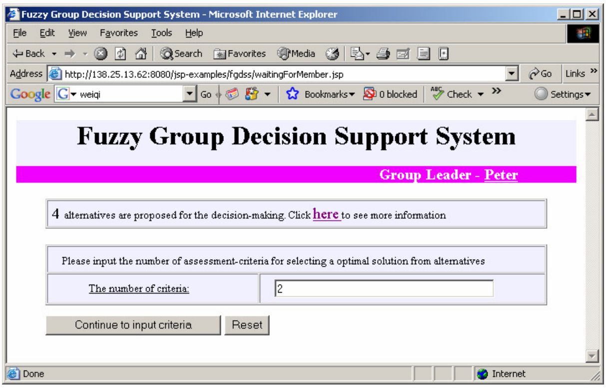


Fig. 11.7: Web page to input the number of criteria

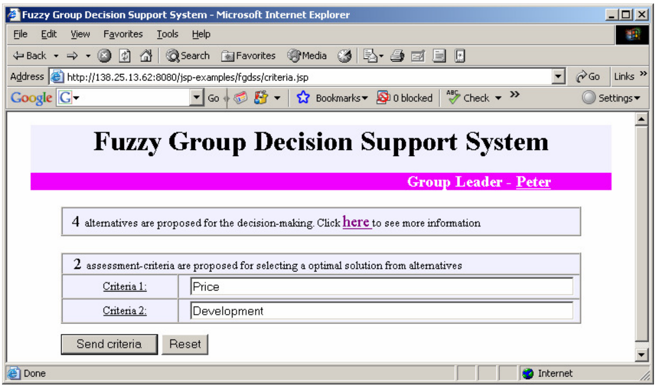


Fig. 11.8: Web page to input criteria

Step 3: *Peter* collects the criteria from all three group members. There are totally six criteria received. Suppose *Peter* regards ‘*Price*’ and ‘*Cost*’ as similar criterion, and selects ‘*Service*,’ ‘*Development time*,’ ‘*Experience*,’ ‘*Quality*,’ and ‘*Cost*’ as the final top five assessment-criteria as shown in Fig. 11.9.

Also, Peter assigns weight ‘*Most important*’ to himself, ‘*Important*’ to David, and ‘*Normal*’ to Kim as shown in Fig. 11.9.

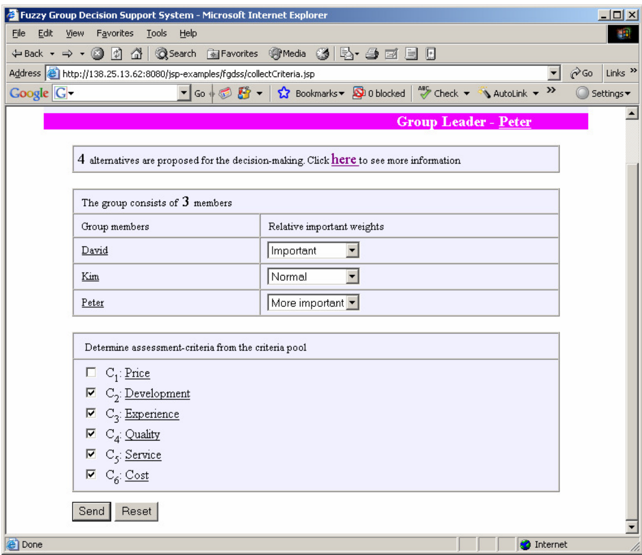


Fig. 11.9: Web page for choosing the criteria and assigning weights

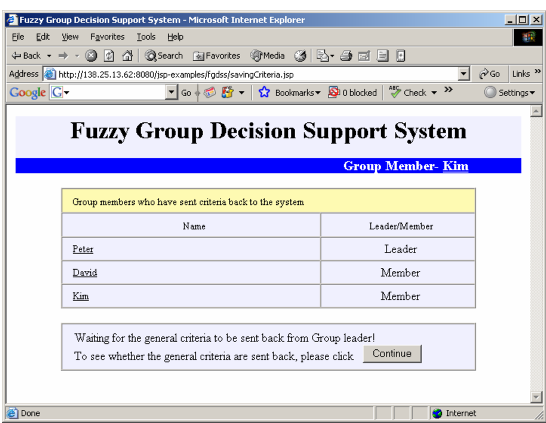


Fig. 11.10: Web page for waiting assessment-criteria sent back from the server

Step 4: Based on criteria received, each group member fills a pairwise comparison matrix of the relative importance of these criteria. Suppose *Peter* fills the matrix as in Fig. 11.11. In the criteria comparison matrix, the criterion *Cost* is thought as '*equally important*' as the criterion *Development time*; also the criterion *Quality* is '*much more important*' than the criterion *Service*, *etc*.

Step 5: Based on criteria and alternatives received, each group member fills a belief level matrix to express the possibility of selecting a solution under some criteria. In Fig. 11.11, comparing with other alternatives under the criterion *Cost*, the preference belief level of the alternative *Software 1* is regarded as '*high*,' the alternative *Software 2* is set as '*high*,' the alternatives *Software 3* as '*high*' and the alternative *Software 4* as '*medium*,' *etc*.

After having filled two matrices and sent them to the server, the group members will just be waiting for the result.

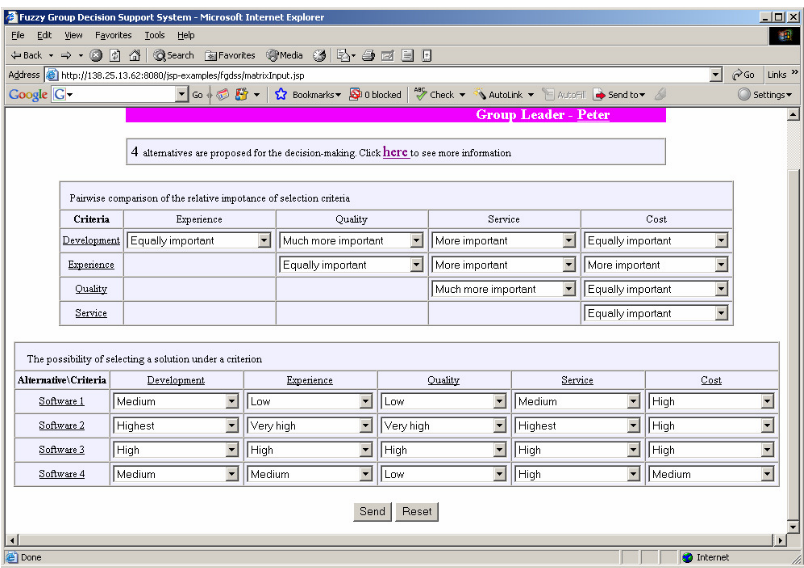


Fig. 11.11: Web page to input criteria comparison matrix and belief level matrix

Step 6: After all group members' matrices are received, the server application does a series of calculations to the belief vector, the weighted normalised fuzzy decision vector and the closeness coefficients of alternatives. The second closeness coefficient is the highest. Then the result is sent back to all group members as shown in Fig. 11.12.

Finally, suppose the group reaches the consensus to the solution for this determining software problem, and *software 2* is selected as the course software to be used in the department.

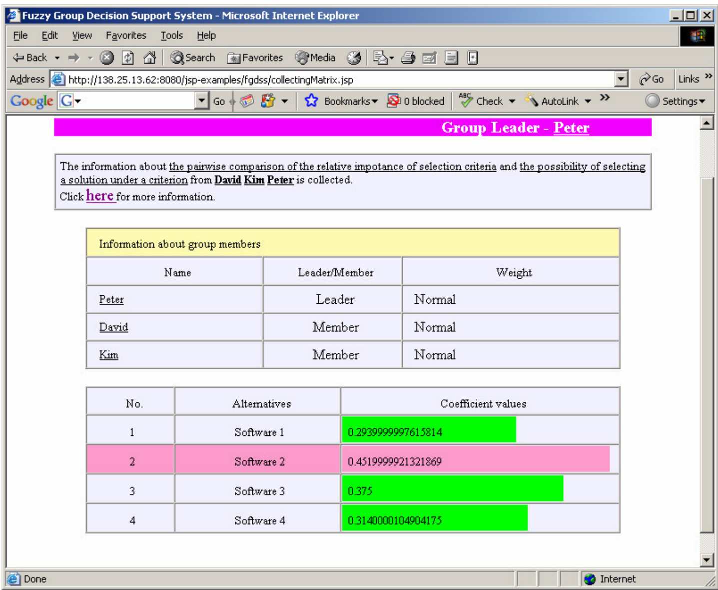


Fig. 11.12: Web page for showing the final result

Example 2 (*using the off-line FGDSS in the attached CD*): *Research project selection*

A research management committee of a university is required to assess a number of individual research projects and can only fund one. These projects proposed come from different departments, different kinds of researchers (earlier or established), with different research topics, and different budgets. In deciding which of the proposed project(s) are to be funded for the year, a number of criteria have to be taken into account, involving the aspects of significance of the project, research methodology, potential to attract external funds, personnel development, and so on. The committee members will discuss to finalise some criteria used in the selection. Support we have three members: *Peter*, *Chris*, and *Tom*, in the committee, and five proposals received: (1) Water resource management (Water-magt); (2) E-government personalisation (E-govt); (3) Data mining for bank customer classification (Data-mining); (4) E-learning system development (E-learning); and (5) Risk management (Risk-magt). The process to make a decision for this selection using the FGDSS is described as follows.

Step 1: Set up a decision making group

The committee chair *Peter* sets up a decision making group, including the title of the group and the issue description (Fig. 11.13), the names of group members (Fig. 11.14) and alternatives (Fig. 11.15).

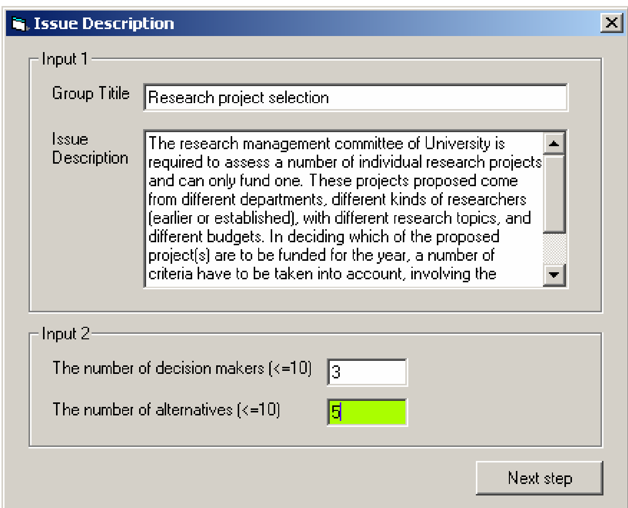


Fig. 11.13: Setting up a group

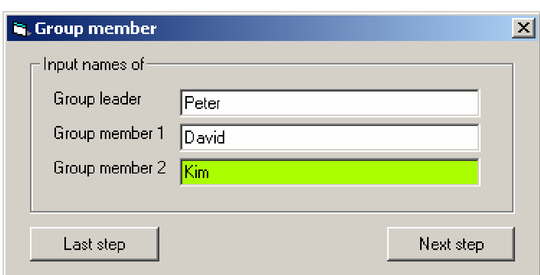


Fig. 11.14: Input the names of group members

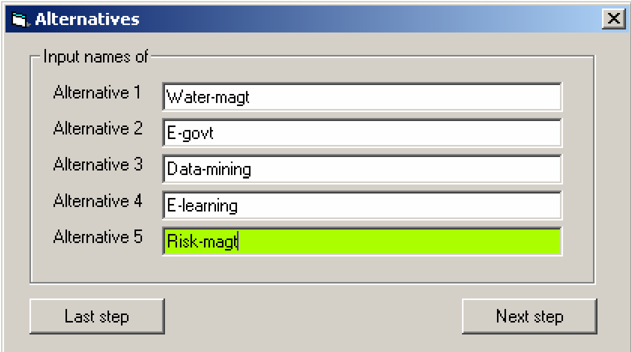


Fig. 11.15: Input alternatives

Step 2: Input criteria by all group members respectively

Here, *Peter* proposes four criteria (Fig. 11.16), *Chris* gives three criteria (Fig. 11.17), and *Tom* gives other three criteria as well (Fig. 11.18).

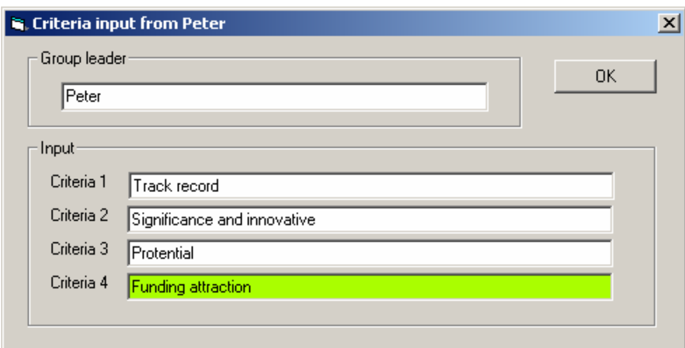


Fig. 11.16: *Peter*'s four criteria

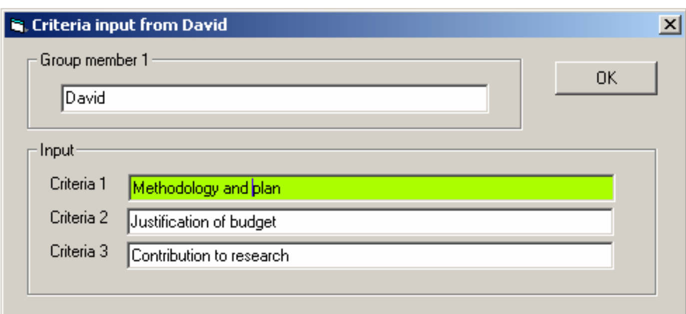


Fig. 11.17: *Chris*'s three criteria

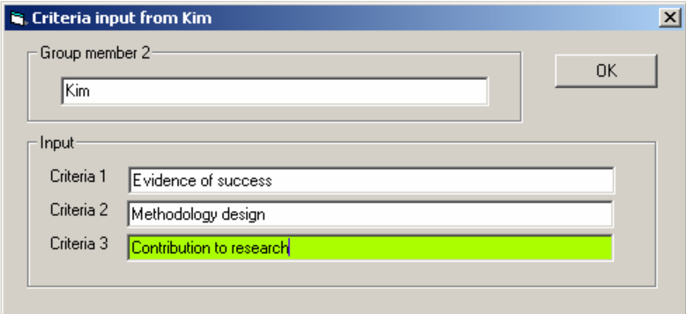


Fig. 11.18: *Tom*'s three criteria

Step 3: Choose the top-t criteria and assign weights

All individual criteria proposed by group members in Step 2 are listed. Obviously, some criteria can be merged or combined, and some may be not relevant to the decision problem. Finally, the following four criteria are chosen as assessment-criteria for the decision problem in the group (Fig. 11.19):

(1) Track record
(2) Significance and innovative
(3) Methodology and plan
(4) Justification of budget

Also, assign weights for all members. Here, *Peter* is assigned as '*most important*,' *Chris* is assigned as '*important*,' and *Tom* is assigned as '*normal*' (Fig. 11.19).

Step 4: Fill the criteria comparison matrix

Based on the assessment-criteria, every member fills a pairwise comparison matrix of the relative importance of these criteria (Fig. 11.20). For example, *Peter* thinks the criterion '*Track record*' is '*much more important*' than the criterion '*Methodology and plan.*'

Step 5: Fill the belief level matrix

Based on the assessment-criteria and alternatives, every member fills a belief level matrix to express the possibility of selecting a solution under some criteria (Fig. 11.18). For example, under the criterion '*Track* *record,*’ the preference belief level of the alternative ‘*Water-magt*’ is regarded as ‘*very high*.’

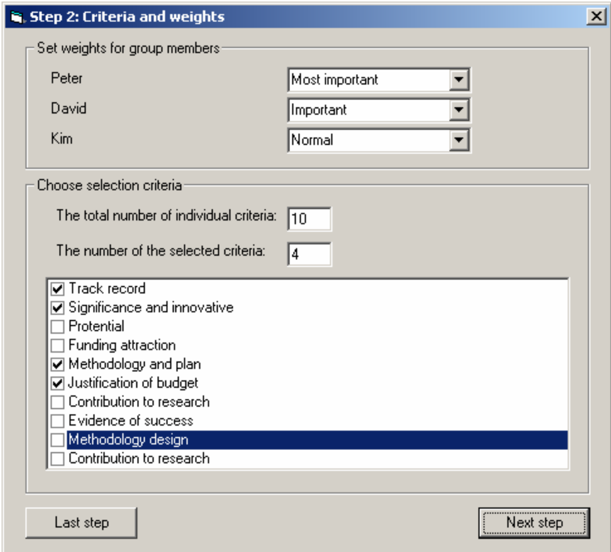


Fig. 11.19: Choosing the criteria and assigning weights

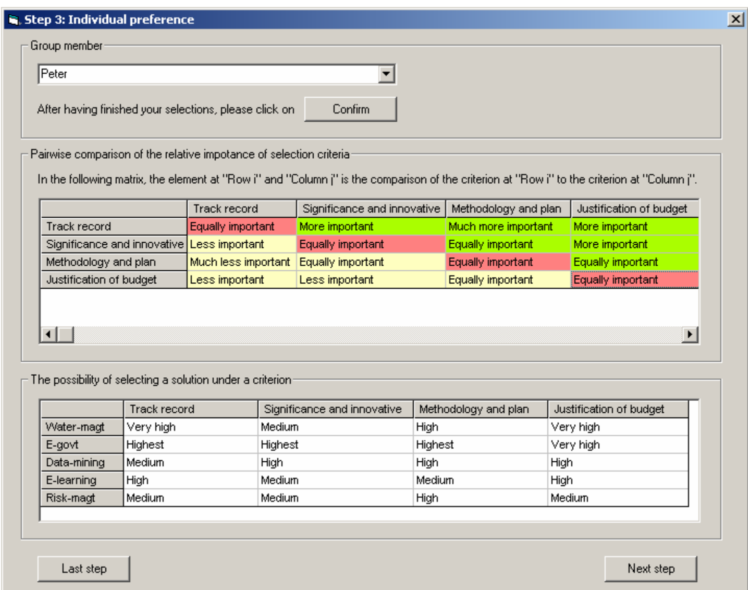


Fig. 11.20: Filling the criteria comparison matrix and the belief level matrix

Step 6: Generate the final result of the problem

Finally, the research project '*E-govt*' is chosen by the committee as it received the highest closeness coefficient value 0.5172 (Fig. 11.21).

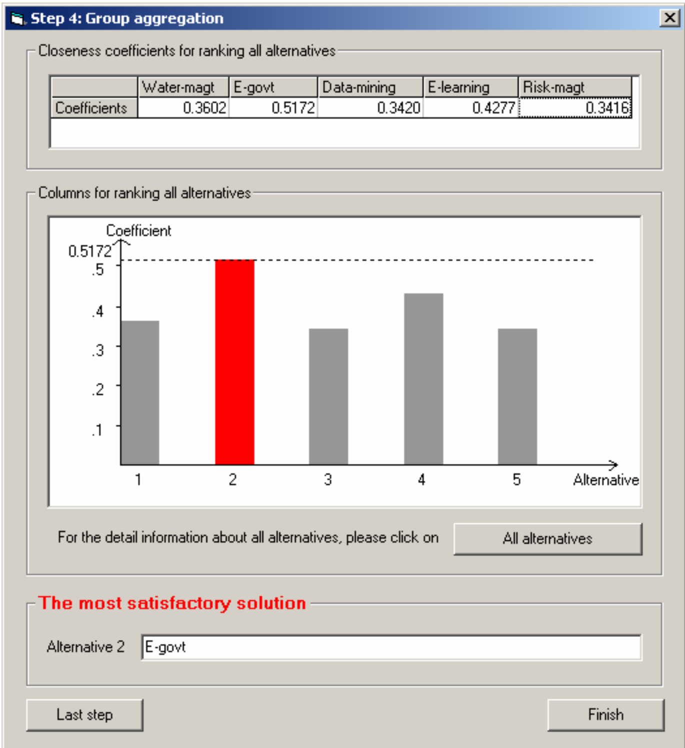


Fig. 11.21: The window for showing the result

However, if the university can fund more projects, '*E-learning*' will be selected as it received the second highest value (0.4277).

11.5 Summary

A Web-based FGDSS and examples illustrated how to use the system in distributed decision making. To help readers use FGDSS, an off-line version is available in the attached CD to run the given example.

Part IV

Fuzzy Multi-Objective Group Decision Making

Chapter 12

Multi-Objective Group DSS

An MODM problem has multiple non-commensurable objectives needed to be achieved. Balancing tradeoffs between multiple objectives will be more important in group than for individuals due to conflicting objectives and opposing viewpoints. We focus on multi-objective group decision-making (MOGDM) techniques, and present an MOGDM framework with five multi-objective group aggregation methods in this chapter. We particularly introduce an intelligent multi-objective group DSS (IMOGDSS) developed, including its architecture, design, and implementation.

12.1 Frameworks

Generally, an MODM problem (also see Chapter 2) can be formulated as follows:

$$\text{(MODM)} \quad \begin{cases} \max & f(x) \\ \text{s.t.} & x \in X = \{x \in R^n \mid g(x) \le b, x \ge 0\} \end{cases} \tag{12.1.1}$$

where $f(x)$ represents n conflicting objective functions, $g(x) \le b$ represents m constraints, and x is an n-vector of decision variables, $x \in R^n$.

Group decision making for solving an MODM problem is named as *multi-objective group decision making* (MOGDM). It provides a group of decision makers with feedback to individual preferences regarding possible solutions to the MODM problem. With several alternatives to the MODM problem, the group members' preferences are aggregated and a final compromise consensus solution is reached. The solution process of the MODM problem can be as the first part of the MOGDM process, and be mixed with the whole group decision process. We therefore have two kinds of frameworks: asynchronous and synchronous.

An *asynchronous MOGDM framework*, as presented in Fig. 12.1, has three stages to complete the process of MOGDM for solving an MODM problem in a group.

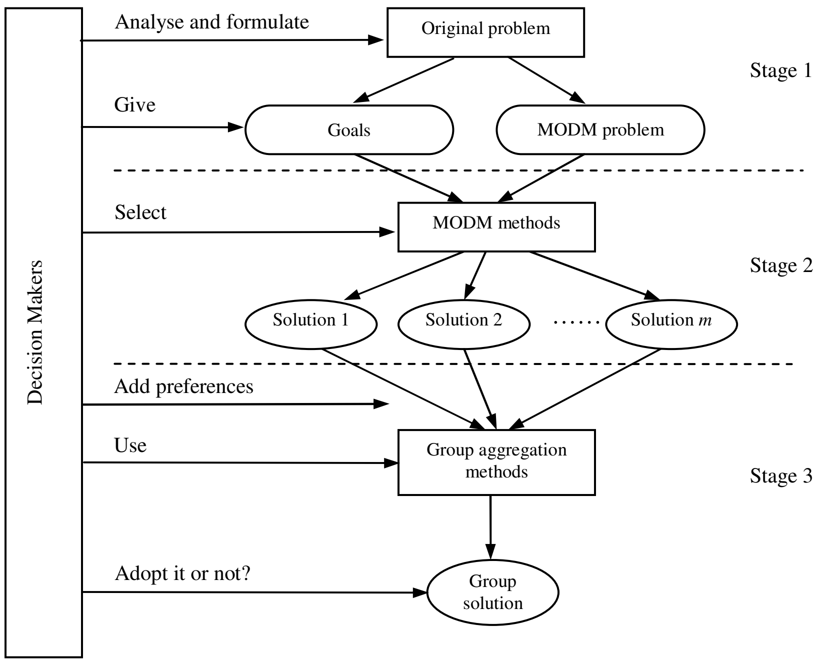


Fig. 12.1: A framework of the MOGDM

In Stage 1, *the initialisation*, a decision group is set up and an MODM problem including its variables, objectives, and constraints are determined. Each member can define their goals or weights to these objectives, which are used to generate individual solutions to the MODM problem.

In Stage 2, *the individual solution*, each decision maker obtains an optimal solution by using a suitable MODM method under their goals and preferences among several methods that are available. They then report their solutions and the aspiration levels for each objective into the group.

In Stage 3, *the group solution*, these individual solutions are as alternatives to form the group's solution for the problem. The decision group members exchange their ideas, express their preferences or judgements on the alternatives, and identify desirable solutions. Each member is given a weight, if it needs, and a utility group aggregation method is then used to determine the 'best' alternative, a compromise solution in general, to the MODM problem through aggregation of individual solutions and their weights. To generate the group solution, each decision maker's individual solution may be given an equal or non-equal priority.

A *synchronous MOGDM framework* consists of three major stages as well. The first stage elicits the decision problem, the weight for each decision maker, and their goals and their minimum acceptable attainment levels for each objective.

The second stage requires group members to indicate their demands for the decision problem. Demands are incorporated into the MODM model as goals in a goal programming formulation. An initial solution for the group is then generated.

In the third stage, group members can indicate their wants, which are not just acceptable levels but desired levels of attainment on these objectives. The wants are then formulated into different prioritised goals to form a new group goal for the problem. A new solution is then offered to the decision group. Group members can also relax one or more objectives so as to allow improvements in other objectives. The interaction with decision makers continues until a final solution is accepted by the group. Obviously, under this framework, group decision methods and MODM methods are mixed to achieve a solution for group.

An MOGDM framework is implemented in a DSS, called an MOGDSS. An MOGDSS, as a specific GDSS, offers multiple decision makers with supporting to reach an agreement on decision involving multi-objectives under a framework of DSS. In general, an MOGDSS provides an interactive procedure for dealing with group decision making problems in which the decision is formulated as an MODM form. For example, Iz (1992) presented two GDSS prototypes based on MOLP with an integrated technique. One aggregated decision makers' individual rankings of the efficient solutions from the Tchebycheff method (Steuer and Choo, 1983) into a group ranking by solving a pure network model suggested by Cook and Kress (1985). Another one embedded the Tchebycheff method into AHP. In Section 12.3, we will introduce an intelligent MOGDSS, which integrates an MODM method-base, a multi-objective based group aggregation method-base, and a knowledge-base into a DSS.

12.2 Multi-Objective Based Aggregation Methods

Based on the asynchronous framework and with some interactive features of the synchronous framework, we propose five multi-objective based aggregation methods (also known as *multi-objective group aggregation* methods): (1) Average solution method (ASM), (2) Weighting objective method (WOM), (3) Weighting member method (WMM), (4) Ideal solution method (ISM), and (5) Solution analysis method (SAM). They are used in Stage 3 of the MOGDM framework presented in Fig. 12.1, but the SAM is interactive and can be used in a synchronous case. In these methods, the term '*solution*' means the objective function values under an optimal solution of an MODM problem.

12.2.1 *Average solution method*

The ASM is also called the *shortest average distance* method. The concept of the shortest distance is applied with a single distance criterion, an average solution, in the method. The objective of the ASM is to obtain the *average* compromise solution from the existing set of solutions provided by group members. The average solution represents the direction of the compromise solution.

Let $S=(S_1,S_2,\cdots,S_n)$, n be the number of decision makers $(n>2)$, $S_i=(s_{i1},s_{i2},\cdots,s_{im})$ be the optimal objective function values under an optimal solution of an MODM problem for simplicity, called '*solution*' (the same for other four methods), from the ith group member. The

MODM problem consists of m objectives (m>1). Mathematically, the ASM is formulated as follows.

$$\begin{aligned} &\text{find} \quad p && (12.2.1)\\ &\text{s.t} \quad d^* = d_p = \min\{d_i; i=1,2,\cdots,n\}\\ &\qquad\qquad = \min\{\sum_{j=1}^{m} | s'_{ij} - av_j |; i=1,2,\cdots,n\} \end{aligned}$$

where

$$s'_{ij} = \begin{cases} s_{ij}/\tilde{s}_j & \text{if } \tilde{s}_j \neq 0; \\ 0 & \text{if } \tilde{s}_j = 0, \end{cases} \quad i=1,2,\cdots,n,\ j=1,2,\cdots,m$$

$$\tilde{s}_j = \max\{s_{ij}; i=1,2,\cdots,n\},\ j=1,2,\cdots,m,$$

$$av_j = \sum_{i=1}^{n} s'_{ij}/n,\ j=1,2,\cdots,m.$$

The solution process involves the following six steps:

Step 1: Input all solutions and establish a solution matrix S

$$S = \begin{bmatrix} s_{11} & s_{12} & \cdots & s_{1m} \\ s_{21} & s_{22} & \cdots & s_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ s_{n1} & s_{n2} & \cdots & s_{nm} \end{bmatrix}$$

Step 2: Calculate the maximum value for each decision objective and establish a relative solution matrix S'

Let

$$\tilde{s}_j = \max\{s_{1j}, s_{2j}, \ldots, s_{nj}\},\ \ j=1,\ldots,m,$$

$$s_{\max} = (\tilde{s}_1, \tilde{s}_2, \ldots, \tilde{s}_m),$$

$$s'_{ij} = \begin{cases} s_{ij}/\tilde{s}_j & \text{if } \tilde{s}_j \neq 0; \\ 0 & \text{if } \tilde{s}_j = 0; \end{cases} \quad i=1,2,\cdots,n,\ j=1,2,\cdots,m.$$

We obtain

$$S' = \begin{bmatrix} s'_{11} & s'_{12} & \cdots & s'_{1m} \\ s'_{21} & s'_{22} & \cdots & s'_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ s'_{n1} & s'_{n2} & \cdots & s'_{nm} \end{bmatrix}$$

Obviously,

$$s'_{ij} \in [0,1]$$

Step 3: Calculate the average solution AV

$$AV = (av_1, av_2, \ldots, av_m)$$

$$av_j = \sum_{i=1}^{n} s'_{ij} / n$$

Step 4: Estimate the distance for each objective of solutions to the average solution. A distance matrix D for each objective of the solutions from the average solution is thus established.

$$D = \begin{bmatrix} d_{11} & d_{12} & \cdots & d_{1m} \\ d_{21} & d_{22} & \cdots & d_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ d_{n1} & d_{n2} & \cdots & d_{nm} \end{bmatrix}$$

where $d_{ij} = \left| s'_{ij} - av_j \right|$, $i = 1, \ldots, n$, $j = 1, \ldots, m$.

Step 5: Sum the distances from different objectives of each solution, we have

$$d_i = \sum_{j=1}^{m} d_{ij}, \ i = 1, \ldots, n$$

Step 6: Find the solution that has the shortest distance

In order to find the solution, the following simple auxiliary problem should be solved.

$$\text{find} \quad p$$
$$\text{s.t.} \quad d^* = d_p = \min\{d_i, i = 1,2,\ldots,n\},\ 1 \le p \le n,$$

where d^* is the shortest total distance between the solutions and the average solution, the pth solution (the optimal objective function values and related optimal solution) is the best compromise solution of this MODM problem for the group. The value chosen for p reflects the way of achieving a compromise by minimising the relative-distance sum of the deviations of objectives from the reference point (average solution).

12.2.2 *Weighting objective method*

The WOM is also called the *weighted shortest average distance method*, which aims to combine group members' preferences and their ranking for each objective into a *relative average solution*.

Let $S = (S_1, S_2, \cdots, S_n)$, $S_i = (s_{i1}, s_{i2}, \cdots, s_{im})$, $i = 1, \cdots, n$, be a solution of an MODM problem from the ith group member, $w_{ij} \in w_j$ be the weight of the jth objective provided by the ith decision maker. Mathematically, the WOM is formulated as follows.

$$\text{find} \quad p \tag{12.2.2}$$
$$\text{s.t} \quad d^* = d_p = \min\{d_i; i = 1,2,\cdots,n\}$$
$$= \min\{\sum_{j=1}^{m} \overline{w}_j \mid s'_{ij} - av_j \mid; i = 1,2,\cdots,n\}$$

where

$$s'_{ij} = \begin{cases} s_{ij} / \tilde{s}_j & \text{if } \tilde{s}_j \neq 0; \\ 0 & \text{if } \tilde{s}_j = 0, \end{cases} \quad i = 1,2,\cdots,n,\ j = 1,2,\cdots,m$$
$$\tilde{s}_j = \max\{s_{ij}; i = 1,2,\cdots,n\},\ j = 1,2,\cdots,m,$$
$$av_j = \sum_{i=1}^{n} s'_{ij} / n,\ j = 1,2,\cdots,m,$$
$$\overline{w}_j = \sum_{i=1}^{n} w_{ij} \big/ n, \quad j = 1,2,\cdots,m.$$

The solution process involves eight steps:

Step 1: Establish a solution matrix S

$$S = \begin{bmatrix} s_{11} & s_{12} & \cdots & s_{1m} \\ s_{21} & s_{22} & \cdots & s_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ s_{n1} & s_{n2} & \cdots & s_{nm} \end{bmatrix}$$

Step 2: Calculate the maximum value for each decision objective and establish a relative solution matrix S'

Let

$$\tilde{s}_j = \max\{s_{1j}, s_{2j}, \ldots, s_{nj}\}, \ j = 1, \ldots, m,$$

$$s_{\max} = (\tilde{s}_1, \tilde{s}_2, \ldots, \tilde{s}_m),$$

$$s'_{ij} = \begin{cases} s_{ij} / \tilde{s}_j & \text{if } \tilde{s}_j \neq 0; \\ 0 & \text{if } \tilde{s}_j = 0, \end{cases} \quad i = 1, 2, \cdots, n, \ j = 1, 2, \cdots, m \cdot$$

We obtain

$$S' = \begin{bmatrix} s'_{11} & s'_{12} & \cdots & s'_{1m} \\ s'_{21} & s'_{22} & \cdots & s'_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ s'_{n1} & s'_{n2} & \cdots & s'_{nm} \end{bmatrix},$$

where $s'_{ij} \in [0,1]$.

Step 3: Calculate an average solution AV

$$AV = (av_1, av_2, \ldots, av_m)$$

$$av_j = \sum_{i=1}^{n} s'_{ij} / n, \ \ j = 1, \ldots, m.$$

Step 4: Evaluate the intensity of importance for each decision objective

Each group member assigns an intensity of importance for each objective as the weight of this objective. Each weight's determination involves the comparison with other elements and their relative importance to the group members with respect to each objective. A weight matrix $W_{n \times m}$ is generated as a result of this process, where the weight is defined according to the interpretation of the scale as follows.

1--*Less important*: experience and judgment slightly favours one of these objectives;

3--*Important*: experience and judgment strongly favour one of these objectives;

5--*More important*: an objective is strongly favoured and its dominance is demonstrated by past experience;

7--*Absolutely important*: very strong evidence favouring one objective over others; and

2,4,6--intermediate values used when further compromise is needed.

We therefore have

$$W = \begin{bmatrix} w_{11} & w_{12} & \cdots & w_{1m} \\ w_{21} & w_{22} & \cdots & w_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ w_{n1} & w_{n2} & \cdots & w_{nm} \end{bmatrix},$$

where $w_{ij} \in (1,2,3,4,5,6,7)$, $i = 1,\ldots,n$, $j = 1,\ldots,m$.

Step 5: Obtain the *average weight* for each objective

The weights at each objective from these group members are processed to determine the average weights of the objective. The major assumption behind the method is that the solutions have reflected decision makers' preferences for their goals, but group members often have conflicting goals for each objective. The weight matrix W is expected to address the conflicts. To reveal decision makers' preferences for each objective the *average weights* are calculated.

$$\overline{w} = (\overline{w}_1, \overline{w}_2, \ldots, \overline{w}_m),$$

$$\overline{w}_j = \sum_{i=1}^{n} w_{ij} / n, \quad j = 1,\ldots,m.$$

Step 6: Estimate the distance of each solution to the *average solution*.

A distance matrix D is thus established.

$$D = \begin{bmatrix} d_{11} & d_{12} & \cdots & d_{1m} \\ d_{21} & d_{22} & \cdots & d_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ d_{n1} & d_{n2} & \cdots & d_{nm} \end{bmatrix}$$

where $d_{ij} = \left| s'_{ij} - av_j \right|$, $i = 1, \ldots, n$, $j = 1, \ldots, m$. Obviously, $0 \le d_{ij} < 1$.

Step 7: Calculate the *weighted distances* from different objectives of each solution.

The weighted distances d_i of each solution S_i, $i = 1, \ldots, n$, from distance matrix D are obtained.

$$d_i = \sum_{j=1}^{m} \overline{w}_j d_{ij}, \quad i = 1, \ldots, n$$

Step 8: Calculate the solution that has the *shortest weighted distance*.

The following simple auxiliary problem should be solved:

find p

s.t. $d^* = d_p = \min\{d_i, i = 1, 2, \ldots, n\}$, $1 \le p \le n$.

The solution S_p is found as the *shortest weighted distance* and it is thus the '*best*' compromise solution of this MODM problem in the group.

12.2.3 *Weighting member method*

This method aims to combine group members' preferences and the ranking of each group member into an average solution of the MODM problem. The degrees of importance of group members are often different. Particularly, when a group meeting has a leader, this leader's preference should be reflected more in the final solution. Thus, this leader may have a higher weight for the solution than other members. In this case, the aggregation of alternative solutions is not only the aggregation of the objective values of the solutions, but also of group members' weights.

Let $S=(S_1,S_2,\cdots,S_n)$, $S_i=(s_{i1},s_{i2},\cdots,s_{im})$, $i=1,\cdots,n$, be a solution of an MODM problem from the ith group member, $w_{ij}\in w_j$ be the weight of the ith member provided by the jth objective. Mathematically, the WMM is formulated as follows.

$$\begin{aligned} &\text{find} \quad p \\ &\text{s.t.} \quad d^* = d_p = \min\{d_i; i=1,2,\cdots,n\} \\ &\qquad\qquad\quad = \min\{\overline{w}_i \sum_{i=1}^{m} | s'_{ij} - av_j |; i=1,2,\cdots,n\} \end{aligned} \tag{12.2.3}$$

where

$$s'_{ij}=\begin{cases} s_{ij}/\tilde{s}_j & \text{if } \tilde{s}_j \neq 0; \\ 0 & \text{if } \tilde{s}_j = 0, \end{cases} \quad i=1,2,\cdots,n,\ j=1,2,\cdots,m$$

$$\tilde{s}_j = \max\{s_{ij}; i=1,2,\cdots,n\},\ j=1,2,\cdots,m,$$

$$av_j = \sum_{i=1}^{n} s'_{ij}/n,\ j=1,2,\cdots,m,$$

$$\overline{w}_i = \sum_{j=1}^{m} w_{ij}\big/n, \quad i=1,2,\cdots,n.$$

The aggregation process involves eight steps, which are similar to those of the WOM presented in Section 12.2.2 except Steps 4 and 5.

In Step 4, the intensity of importance is assigned for each group member. The determination of each weight involves the comparison with other group members' relative importance.

$$W=\begin{bmatrix} w_{11} & w_{12} & \cdots & w_{1m} \\ w_{21} & w_{22} & \cdots & w_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ w_{n1} & w_{n2} & \cdots & w_{nm} \end{bmatrix},$$

where $w_{ij}\in(1,2,3,4,5,6,7)$, $i=1,\ldots,n$, $j=1,\ldots,m$.

In Step 5, the weights obtained for group members are processed to determine the average weights of these group members.

$$\overline{w}=(\overline{w}_1,\overline{w}_2,\ldots,\overline{w}_m),$$

$$\overline{w}_i = \sum_{j=1}^{m} w_{ij} / n \text{ , } i = 1,2,\cdots,n \text{ .}$$

In Step 6, the distance matrix D is obtained.

$$D = \begin{bmatrix} d_{11} & d_{12} & \cdots & d_{1m} \\ d_{21} & d_{22} & \cdots & d_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ d_{n1} & d_{n2} & \cdots & d_{nm} \end{bmatrix}$$

where $d_{ij} = \left| s'_{ij} - av_j \right|$, $i = 1,\dots,n$, $j = 1,\dots,m$.

Step 7 calculates the weighted distances d_i of each solution S_i, $i=1,2,..,n,$ from the distance matrix D.

$$d_i = \overline{w}_i \sum_{j=1}^{m} d_{ij} \text{ , } i = 1,\dots,n$$

In Step 8, the problem is solved by the following formula:

find p

s.t. $d^* = d_p = \min\{d_i, i = 1,2,\dots,n\}$, $1 \le p \le n$.

The solution S_p has the shortest weighted distance and it is thus the best compromise solution. A new solution can be generated by changing these weights.

12.2.4 *Ideal solution method*

In this method, the distance from the ideal solution is used to evaluate all solutions provided by group members. The method aims to obtain the '*best*' compromise solution, which is the one that is the closest to the ideal solution, that is, it has the shortest distance from the ideal solution.

Let $S = (S_1, S_2, \cdots, S_n)$, $S_i = (s_{i1}, s_{i2}, \cdots, s_{im})$, $i = 1, \cdots, n,$ be a solution of an MODM problem from the ith group member, m be the number of objectives (m>1). $S_0 = (s_{01,} s_{02} ,..., s_{0m})$ be the ideal solution. Mathematically, the ISM is formulated as follows.

find p (12.2.4)

s.t. $d* = d_p = \min\{d_i; i = 1,2,\cdots,n\}$

$$= \min\{\sum_{j=1}^{m} | s'_{ij} - s_{0j} |; i = 1,2,\cdots,n\}$$

where

$$s'_{ij} = \begin{cases} s_{ij} / \tilde{s}_j & \text{if } \tilde{s}_j \neq 0; \\ 0 & \text{if } \tilde{s}_j = 0, \end{cases} \quad i = 1,2,\cdots,n,\ j = 1,2,\cdots,m$$

$$\tilde{s}_j = \max\{s_{ij}; i = 1,2,\cdots,n\},\ j = 1,2,\cdots,m.$$

The group aggregation process involves six steps, which are similar to those of the ASM presented in Section 12.2.1 except the average solution is changed to the ideal solution. The calculation of an ideal solution is the same as those of the ISGP method (discussed in Table 2.1).

When an ideal solution $S_0 = (s_{01,} s_{02}, ..., s_{0m})$ is generated, the algorithm starts to measure the distance of the ideal solution to each other solution. A distance matrix D for each objective of solutions to the ideal solution is thus established.

$$D = \begin{bmatrix} d_{11} & d_{12} & \cdots & d_{1m} \\ d_{21} & d_{22} & \cdots & d_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ d_{n1} & d_{n2} & \cdots & d_{nm} \end{bmatrix}$$

where $d_{ij} = \left| s'_{ij} - s_{0j} \right|$, $i = 1, \ldots, n$, $j = 1, \ldots, m$.

The distances from different objective values of each solution are obtained:

$$d_i = \sum_{j=1}^{m} d_{ij},\ i = 1, \ldots, n$$

The final solution that has the shortest distance is then found from

find p

s.t. $d^* = d_p = \min\{d_i, i = 1,2, \ldots, n\}$, $1 \le p \le n$,

where d^* is the shortest total-distance between the solutions and the ideal solution, the pth solution is the most closest solution as the final compromise solution of this MODM problem in the group.

12.2.5 *Solution analysis method*

This method is designed to use a relaxation process for these objective values based on a preliminary solution, which can be produced by the ISM presented in Section 12.2.4. The method provides more interaction and negotiation for group members.

Let $S=(S_1,S_2,\cdots,S_n)$, $S_i=(s_{i1},s_{i2},\cdots,s_{im})$, $i=1,2,\cdots,n$, be a solution of an MODM problem from the ith group member, m be the number of objectives ($m>1$). Mathematically, the SAM is formulated as follows.

Find p' through relaxing s_p based on

{

find p (12.2.5)

s.t. $d^*=d_p=\min\{d_i;i=1,2,\cdots,n\}$

$$=\min\{\sum_{j=1}^{m}|s'_{ij}-s_{0j}|;i=1,2,\cdots,n\}$$

where

$$s'_{ij}=\begin{cases} s_{ij}/\tilde{s}_j & \text{if } \tilde{s}_j\neq 0; \\ 0 & \text{if } \tilde{s}_j=0, \end{cases} \quad i=1,2,\cdots,n,\ j=1,2,\cdots,m$$

$$\tilde{s}_j=\max\{s_{ij};i=1,2,\cdots,n\},\ j=1,2,\cdots,m.$$

}

The solution process involves two stages:

Stage 1: Produce a preliminary solution by using Steps 1 to 6 of ISM;

Stage 2: If some of the objectives are satisfactory and others are not, we will use the STEM (see Table 2.1) method to relax one of the satisfactory objectives enough so as to allow improvements in unsatisfactory objectives. A new solution is generated. If it is not accepted, we will have next iterative cycle. The interaction with decision makers continues until a compromise solution for the MODM problem is accepted in the group. In some cases decision makers' relaxation values are not feasible. By using this method, when a relaxation fails, the method will enable users to continue to re-enter a new set of relaxation values.

12.3 An Intelligent MOGDSS

We now present an intelligent multi-objective group DSS (IMOGDSS), which could be applied to solve MOLP problems in a decision group. To utilise the potential of the MODM method-base effectively, this IMOGDSS is designed to include seven popular MODM methods and has the capability of guiding decision makers to select the most suitable MODM method from the seven methods for solving their particular problems. A knowledge-based intelligent guide is provided to achieve the aim. As the IMOGDSS is used in a decision group, after each group member gives a solution for an MODM problem, a group subsystem is launched to exchange ideas for decision objectives and their goals, and to identify acceptable and desirable solutions. Usually a negotiation about their solutions is processed so that this decision group achieves a compromise but consensus solution of the MODM problem. A GDM method-base that consists of five group aggregation methods described in Section 12.2 is utilised to find a compromise solution. In these methods, SAM provides a possibility to have more interaction in a group, and can be suitable for both frameworks of asynchronous and synchronous.

As shown in Fig. 12.2, the IMOGDSS has five bases: (a) database, (b) MODM method-base, (c) GDM method-base, (d) model-base, and (e) knowledge-base. These resources can be accessed by seven major subsystems: (1) interface subsystem, (2) problem input subsystem, (3) intelligent guide subsystem, (4) method subsystem, (5) result (management) subsystem, (6) model (management) subsystem, and (7) group (aggregation) subsystem.

Decision makers

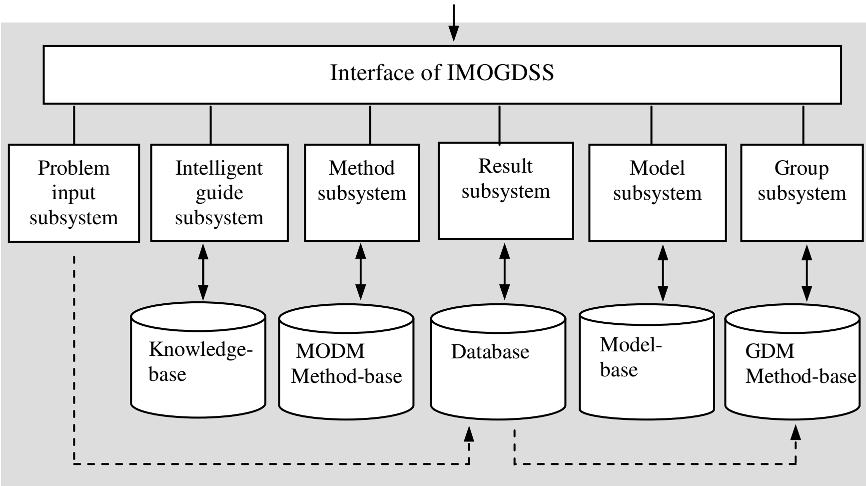


Fig. 12.2: The structure of the IMOGDSS

Some methods are more suitable and efficient than others in the solution of a particular decision problem of particular decision makers. Hence, the IMOGDSS contains seven methods in its method-base for decision makers to select the most suitable one for solving their problems. These methods are:

- Efficient Solution via Goal Programming (ESGP) (Ignizio, 1981),
- Interactive Multiple Objective Linear Program (IMOLP) (Quaddus and Holzman, 1986),
- Interactive Sequential Goal Programming (ISGP) (Hwang and Masud, 1979),
- Linear Goal Programming (LGP) (Ignizio, 1976),
- Step Method (STEM) (Benayoun *et al.*, 1971),
- STEUER (Steuer, 1977), and
- Zionts and Wallenius (ZW) (Ziont and Wallenius, 1975).

(More details have been shown in Table 2.1 in Chapter 2).

We have also implemented the five group aggregation methods presented in Section 12.2 in the GDM method-base. The focus of these methods is to determine a compromise solution to an MODM problem, which best conforms to the preferences of the group members. These methods are implemented as independent executables to facilitate the flexibility required of the system. These methods share similar data acquisition routines and these routines are developed as independent modules so that data acquired could be accessed by all the methods.

The selection of the most suitable method from the MODM method-base is always difficult to accomplish because of the dearth of expertise and experience needed to understand the specific features of the available MODM methods, as well as the ability to match an MODM model with current decision needs. Usually, only experts in the field are able to take advantage of an MODM method-base. This is because sophisticated analytical skills on the part of decision makers are required to identify the problems and match each problem with an appropriate MODM method. Therefore, an intelligent technique is needed to support the selection of methods. A knowledge-base system is utilised to provide the guidance on the selection of suitable MODM methods according to different problem situations and decision makers' situations. With the design, the IMOGDSS allows non-technical decision makers to interact fully with the system and get recommendations for a suitable decision method.

12.4 Design of the Intelligent Guide Subsystem

The knowledge-based intelligent guide subsystem plays an important role in the IMOGDSS. Through helping decision makers choose a suitable method, it can effectively improve their confidence and truthness to use this DSS to solve their problems. Its design is described in this section.

12.4.1 *Knowledge acquisition process*

The knowledge acquisition is the process of capturing the experts' knowledge about a domain into a system. The process includes two main phases: the identification and collection of knowledge, then the representation of the facts representing the expertise to be kept in a system's knowledge-base. The following steps are used to identify and collect the experts' knowledge about MODM methods:

- *Method identification*: identifying a number of traditional and popular MODM methods to build an MODM method-base.
- *Validity recognition*: a number of validities are recognised. They are conceptual validity, logical validity, experimental validity, and operational validity.
- *Methods comparison*: comparing all methods included in this system through different points of view and classes.
- *Characteristics and concepts identification*: the characteristics and concepts of the MODM methods are identified.
- *Selection of the type of knowledge representation*: there are four main types of knowledge representation schemes in a knowledge-base: production rules, semantic nets, frames, and logic. We used the type of *production rules*.

12.4.2 *Characteristics analysis models*

To build the knowledge-base in the intelligent guide subsystem, the knowledge for the selection of MODM methods is first structured by capturing both the MODM methods and their characteristics.

The characteristics of MODM methods are classified into four classes, that is, *DMs* (*decision makers*)*-related*, *Methods-related*, *Problems-related*, and *Solutions-related* characteristics. By studying the characteristics of the seven methods implemented in the IMOGDSS prototype, four analysis models for the four classes of characteristics are produced respectively.

The *DMs-related* characteristics analysis model includes the characteristics that are related to decision makers' preference for selecting a method to solve a decision-making problem. Some of these characteristics are decision makers' desire to interact with the system, decision makers' ability to provide data for a specific MODM method. The *Methods-related* characteristics analysis model consists of the characteristics that are related to the solution process of MODM methods, such as whether to use a linear programming technique or a goal programming, whether to define an ideal solution. The *problems-related* characteristics analysis model includes the characteristics that are dependent on the actual decision problem. For example, some MODM methods such as IMOLP and LGP require the provision of weights for each objective, while ISGP and LGP need to provide the goals for each objective. The *solutions-related* characteristics analysis model consists of the characteristics that are related to the types of solution processed. Some MODM methods such as ESGP, ISGP, LGP produce only a subset of the efficient solutions, while others such as STEUER produces all efficient solutions.

12.4.3 *Novice and intermediate modes*

To ensure the consistency of knowledge in a knowledge-base, the principle of assimilation is applied for combining the characteristics in each characteristic model and to produce the *characteristic-method* models. To provide the appropriate guidance for decision makers possessing different levels of knowledge about MODM methods, we capture the characteristics into two groups in order to build the question models as a front-end for the knowledge-base. The two groups of characteristics are provided, namely, the *novice* and *intermediate* modes.

The *novice* mode includes non-technical characteristics that are applied to decision makers who are totally unfamiliar with MODM methods. The *novice* mode will correspond to a set of general non-technical questions regarding a decision problem, its expected solution(s), and its decision makers' preferences. From the answers obtained from decision makers, the most suitable method will be found and recommended. A total of 10 characteristics are identified for the *novice* mode as listed in Table 12.1, and will be used in the fact-base of the expert system we developed in this system.

The *intermediate* mode is designed for decision makers who are familiar with some methods of MODM, or not so familiar with the methods but have basic knowledge on MODM models and solution process. It consists of 14 characteristics of MODM methods. It will be used to find methods corresponding to a set of inputs for decision makers using the intermediate mode. Decision makers can discover which method corresponds to a set of inputs by responding to some technical questions based on their decision problems, desired solutions, and data preparation. We only discuss the *novice* mode here.

Table 12.1: Characteristics (Char.) and *facts* related for the novice mode

Char. No.	Char. Name	Char. Definition	Char. Facts
1	*Interaction*	more interaction with the system	Char. 1
2	*Subset*	system provides a set of solutions	Char. 2
3	*Unique*	system provides a unique solution	Char. 3
4	*S-Selection*	system selects one satisfactory solution	Char. 4
5	*D-Selection*	user selects one satisfactory solution	Char. 5
6	*Analyse*	solution analysis (*e.g.* improving/ sacrificing the value of objectives)	Char. 6
7	*Ideal*	system defines an ideal solution	Char. 7
8	*Weight*	set up weights for objectives	Char. 8
9	*Goal*	set up goals for objectives	Char. 9
10	*Priority*	set up priorities for objectives	Char. 10

12.4.4 *Logical connectivity and characteristics*

We conduct a connection analysis between the seven MODM methods and the 10 characteristics for the *novice* mode (and 14 characteristics for the *intermediate* mode). Fig. 12.3 shows the logical connectivity among the seven MODM methods and the 10 characteristics for the *novice* mode. We can see that each method is connected with several characteristics. For example, the ISGP method is connected with the characteristics of the '*interaction*,' '*subset*,' '*D-selection*,' '*ideal*,' and '*goal*.'

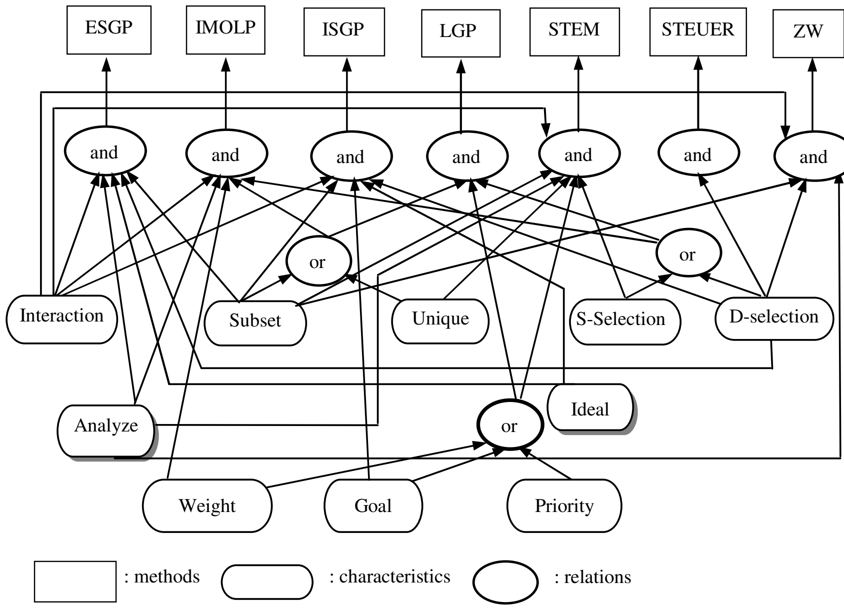


Fig. 12.3: Logical connectivity between MODM methods and their characteristics

12.4.5 *Questions and responses*

Based on the two modes, two groups of questions are designed to the two levels of decision makers, respectively. They are shown through a series of dialog boxes in the IMOGDSS. Each dialog box shows one question, with two response items: T (yes or the first option) and B (not or the second option), and a list of weights to choose for indicating the intensity of importance of the preferred characteristics. Four levels of the weights are defined in the system:

(1) Very important,
(2) Important,
(3) General, and
(4) Less important.

These responses are used to match the characteristics of one method. The weights are used to measure which method is the most appropriate if no method fully matches with decision makers' preferred characteristics. The relationships among these questions, responses, and characteristics for the *novice* mode are shown in Table 12.2.

Table 12.2: Questions (Que.), responses (Res.), and characteristics (Char.) for the novice mode

No	*Questions*	*Resp.*	*Char. Name*	*Char. No*
1	*Would you like to have more interaction with the system?*	T	Interactive	1
		B	Not	Not
2	*Would you like the system to provide a set of solutions or a unique solution?*	T	Subset	2
		B	Unique	3
3	*Would you like the system or yourself to select a satisfactory solution?*	T	S-Selection	4
		B	D-Selection	5
4	*Would you like to analyse solutions (e.g., improving/sacrificing the value of objectives)?*	T	Analyse	6
		B	Not	Not
5	*Would you like the system to define an ideal solution?*	T	Ideal	7
		B	Not	Not
6	*Have you prepared a weight for every objective?*	T	Weight	8
		B	Not	Not
7	*Have you prepared a goal for every objective?*	T	Goal	9
		B	Not	Not
8	*Have you prepared a priority for every objective?*	T	Priority	10
		B	Not	Not

12.4.6 *Inference process*

We first give the definitions of completed match and n-step match, and then introduce ignoring characteristic match strategy (ICMS).

Let $M = \{M_1, M_2, ..., M_7\}$ be a method set, $\overline{C}$ be a characteristics set of MODM methods, $C_i = (C_{i1}, C_{i2}, ..., C_{ik})$, $C_{ij} \in \overline{C}$ $(j = 1,2,..., k)$ be characteristics of M_i, $R = (R_1, R_2, .., R_k)$ be characteristics of decision makers preferences (it is covered by the responses of decision makers for the questions listed in Table 12.2). For any $p \in \{1,2,..., k\}$ there exists an *i* and j such that $C_{ij} = R_p$, $W = (W_1, W_2, .., W_k)$ is a weighted vector for R, k =10 for the novice mode.

Definition 12.1 *RC* Completed match: if there exists an $i \in \{1,2,..,7\}$ such that for any $j \in \{1,2,..,k\}$,

$$R_j = C_{ij},$$

we then say R and M_i is a *RC* completed match and denote it as $R \equiv C_i$ or $R^0 = C_i^0$. A *completed match* means the characteristics of a method completely match with decision makers' preferred characteristics.

Definition 12.2 *RC* n-step match: set $R^n = (R_{j1}, R_{j2}, ..., R_{jk-n})$, and $\{R_{j_1}, R_{j_2}, ..., R_{j_{k-n}}\} \subset \{R_1, R_2, .., R_k\}$, $n = 1,2,..k-1$, if there exists an $i \in \{1,2,..,7\}$ such that $\forall j \in \{j_1, j_2, .., j_{k-n}\}$

$$R_j^n = C_{ij},$$

we then say R and M_i is a *RC* n-step match and denote it as $R_j^n = C_{ij}$, and n is called a *match degree*, where $C_i^n = \{C_{ij_1}, C_{ij_2}, ..., C_{ij_{k-n}}\} \subset \{C_{i1}, C_{i2}, .., C_{ik}\}$. An n-step match means that only k-n characteristics of a method match with decision makers' preferred characteristics.

Theorem 12.1 If for any $i \in \{1,2,..,7\}$, and R and M_i is not a completed match, then there exists $n < k$, such that R and M_i is *RC* n-step match.

Proof. Obvious.

We describe the inference process as follows:

Each preferred characteristic is given a weight by decision makers. A weighted vector of characteristics is therefore built. Through this weighted vector, the lowest weight W_l ($1 \le l \le k$) is obtained from the weight vector W, R_l and C_{il} (i=*1*,...,7) that according to W_l are then found and ignored, if for any i, R and C_i is not a completed match. If there is an existing M_i such that R and C_i is a 1-step match, this method M_i is then recommended to decision makers. Otherwise, the second lowest weight is determined, another characteristic is ignored and a *RC*

2-step match is measured. Based on *Theorem 12.1*, an n-step match method will be found after ignoring process *($n<k$) n* times. This strategy is called the *ignore characteristic match strategy*. According to this strategy, two different methods may be recommended to two different decision makers for the same decision problem because they are assigned different weights for characteristics even though their responses for the questions are the same.

Decision makers' responses and weights for these questions are converted to a response vector R that consists of the characteristics decision makers need, and a weighted vector W that consists of the weight of each characteristic. If decision makers' responses are a *RC completed match* with the characteristics of an MODM method, this method is recommended without the use of *ICMS*. However, it is not often that decision makers' responses exactly match the characteristics of one method. The *ICMS* is thus used based on Theorem 12.1 to find M_{i0} such that a *RC* n-step *match* is found. The objective of this method is to combine decision makers' preferences and the weights for each characteristic to find the most suitable method that best satisfies decision makers' requirement.

12.5 Implementation

This section describes the implementation of three major subsystems: MODM method subsystem, intelligent guide subsystem, and group subsystem in the IMOGDSS.

12.5.1 *The MODM method subsystem*

The MODM method subsystem is used to execute the seven selected MODM methods. As described in Section 12.4, these methods share similar data acquisition routines so that data acquired could be accessed by all the methods.

Interactions are carried out during a solution process in this subsystem. Interactive approach explores promising solutions rather than simply finding the *optimal* solution. Through interaction with the problem owners, *i.e.*, decision makers, the solution process generates solutions that reflect their preferences at most. Interactions can generate multiple alternative solutions for evaluation and selection, and it thus becomes a learning process for decision makers to understand problems better. There are different types of interactions among the seven methods and each method takes one or more of these types. As introduced in Chapter 2, Table 2.2, the first type of interaction, *pre-interaction*, is performed before the solution process even starts. In this type, explicit preference function of decision makers is needed. In the second type, *pro-interaction*, the preference information of decision makers is needed during the solution process. In this case, decision makers are required to provide online preference information, but no explicit preference function is needed. This type of approach is widely known as an interactive approach. The third type of approach, *post-interaction*, requires preference information after a set of candidate solutions has been generated. In this case, decision makers are simply required to choose the most satisfactory solution from the final set.

The solution process of each method is quite different. For example, LGP uses pre-interaction with users via collecting the weights, goals and priorities of the objectives. On the other hand, IMOLP and ISGP use all the three interactions. The IMOGDSS takes care of all interactions via windows and produces the final accepted solution (decision variables and objectives) by decision makers. Referring to the production planning example given in Chapter 2, Section 2.3.4, Fig. 12.4 shows such a final solution selected by decision makers using the ISGP method.

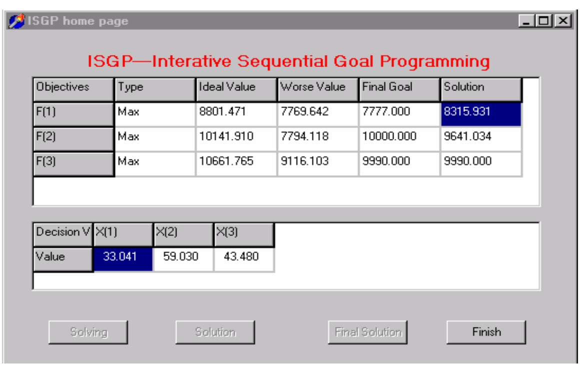


Fig. 12.4: A final solution selected by decision makers using the ISGP method

The solution shows that when producing 33.041 units of product x_1, 59.030 units of x_2, and 43.480 units of x_3, the company will obtain maximised *profits*, *quality*, and *work satisfaction*.

12.5.2 *The intelligent guide subsystem*

The intelligent guide subsystem consists of five sub-sub systems (we just call subsystems for simplicity): *question*, *response*, *method-show*, *ignoring (missing) characteristic strategy* (*ICS*), and *main-control* subsystems, and a knowledge-base. The knowledge-base includes a set of *facts* to define the knowledge about the MODM methods and a set of *rules* for finding a suitable method for a particular decision maker (Fig. 12.5).

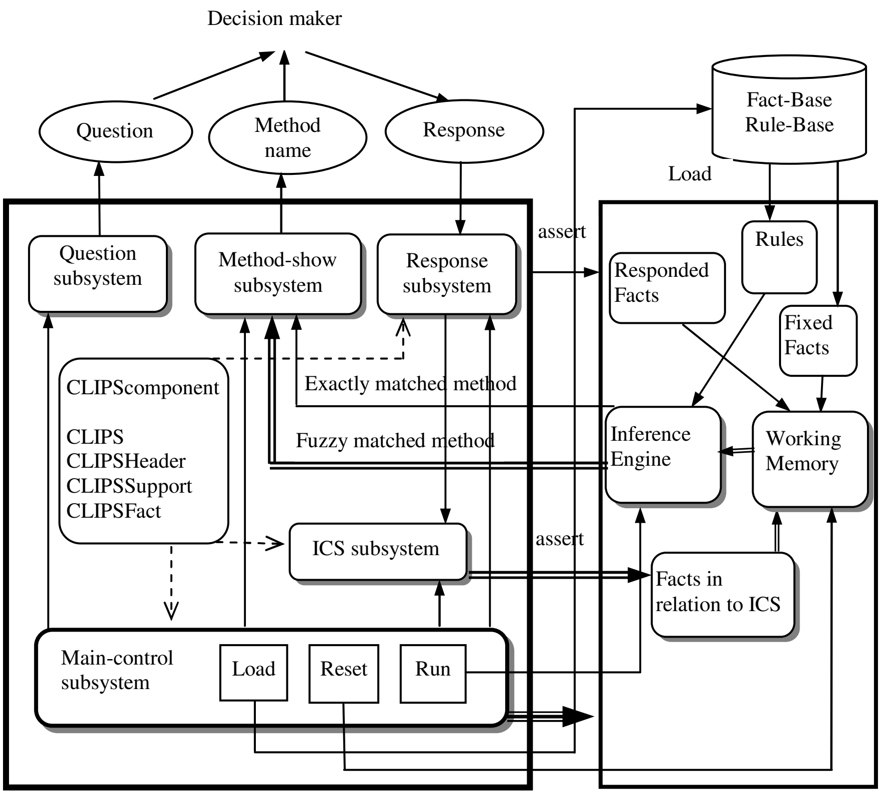


Fig. 12.5: Intelligent guide subsystem and its working principle

The IMOGDSS uses the inference engine provided by the expert system shell *CLIPS*. The question subsystem first questions decision makers by using an elicitation technique. The responses are received and analysed by the response subsystem. The responses to each question are asserted in the working memory by the inference engine, and responses to the weight of each question are sent to the ICS subsystem. If a suitable method is found the name of the method will be displayed to decision makers by the method-show subsystem. Otherwise a fuzzy (*n*-step) match strategy is performed.

Facts are one of the basic high-level forms for representing information in a knowledge-base system. Each fact represents a piece of information that has been in the current list of facts. The knowledge-base for the selection of MODM includes several groups of facts that have different functions. The basic knowledge about each MODM method and its various characteristics are described by a group of facts as listed in Table 12.1. Another group of facts is to relate the response of each question to the facts to be asserted by the inference engine into the working memory. We also need to get a number of facts to relate each characteristic to its corresponding question. The next group of facts relates to follow-up questions to follow given responses. It is necessary to get a set of facts to relate facts that are grouped under the same class. The last set of facts is used to initialise the inference process.

The knowledge is represented using '*def-templates*' and '*def-facts*.' Every '*def-facts*' defines directly a fact. A def-template defines a group of related fields in a pattern similar to the way in which a record is a group of related data. The definition of a piece of def-facts is shown in the following code:

method: an MODM method and its characteristics

```
(deffacts  Method1
   (Method
      (Number 1)
      (Name ESGP)
      (Char1 interaction)
      (Char2 subset)
      (Char5 d-selection)
      (Char6 analyse)
      (Char7 ideal)
   )
```

```
)
```

Rules are used to represent heuristics to specify a set of actions to be performed for a given situation. This study defines a set of rules, which collectively works together for the method selection. The method selection knowledge-base system attempts to match all the characteristics of a method to those already asserted into the working memory. If the match failed, a characteristic which has the least weight will be ignored. A method will be selected if all its characteristics (or after ignoring) are found in the working memory. We have also incorporated many heuristics that assist the system in the conflict resolution phase of the inference. For example, the rule to inform the user that a suitable method has been found shall have priority over other rules. The definition of an example of rules is shown in the following code**:**

call-question: a rule relating to get the questions' number and its responses' number.

```
(defrule get-question
   (declare (salience 10))
    ?v1 <- (Question (Number ?num1))
      (test (neq ?num1 -1))
=>
   (retract ?v1)
   (bind ?response (quest ?num1))
   (assert (Response (Question ?num1)(Answer ?response)))
)
```

All patterns must be satisfied by the facts in the fact-list for the rules to fire. A program won't start running unless there are rules whose left-hand side (LHS) is satisfied by the facts. The inference engine sorts the activations according to their salience. This sorting process eliminates the confliction of deciding which rule should be fired next.

A DELPHI-CLIPS interface program supports the execution of the CLIPS operations in the DELPHI working environment. Within this interface program, the intelligent subsystem can assert a set of facts by a public method or a function, such as *AssertString* through the *TClips* code. The subsystem can also use *FactCount* and *Fact* properties for getting all the facts in the fact-base, such as the *Assert* and *Retract* methods to assert and retract a fact. The *Tclips* component also has a set of events to be used to monitor CLIPS and its execution.

When CLIPS is called, the intelligent guide subsystem first checks if the CLIPS supporting files are in the correct location. The subsystem then calls *Initialise* CLIPS for initialisation. The subsystem again calls procedure *Clear* to clear the fact-base. The next step is to load the CLIPS file that includes all fixed facts and rules. After this file is loaded, the subsystem executes *Reset procedure* and all fixed facts are entered into the agenda. The last function, *Run* is then called. All responses of decision makers will be converted into the facts and the intelligent subsystem asserts them in the fact-base. At the same time, the rules are fired and the subsystem starts an inference process. The CLIPS system attempts to match the patterns of rules with the facts in the fact-list. If all the patterns of a rule match the facts, the rule is activated and put on the agenda. The agenda is a collection of activations that are those rules that match pattern entities. Fig. 12.5 shows the intelligent subsystem and its working principle.

12.5.3 *The group subsystem*

The group subsystem in the IMOGDSS is used to aggregate multiple decision makers' solutions for an MODM problem. It includes two input schemes: *online* scheme and *offline* scheme. The online schema is used to read solution data from a text file, and the offline one obtains data through the keyboard.

Different methods offer different solution processes, and have different input requirements. For example, WMM method needs members to give weights (see Fig. 12.6 for the production planning example) and ISM needs showing an ideal solution.

All decision makers' solutions (here, still mean the optimal objective function values), the average solution and the ideal solution, can be shown in a chart in order to view and understand the distances between the *average* (or *ideal*) solution and the decision makers' solutions. Fig. 12.7 displays three group members' solutions and an *average* solution for the production planning problem. We can see from the figure that two members' solutions (Solutions 1 and 3) are very close in Objective 2 (*Quality*) and have the almost same value in Objectives 1 (*Profits*) and 3

(*Worker satisfaction*). The *average* solution (indicated in white circles) has its three objectives' values close to Solution 1 and Solution 3.

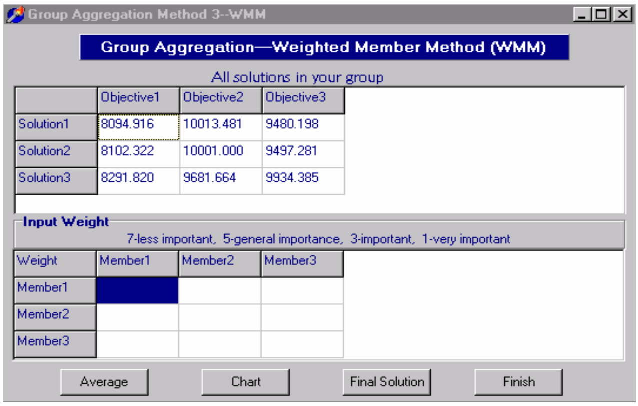


Fig. 12.6: A screen of WMM in the group aggregation process

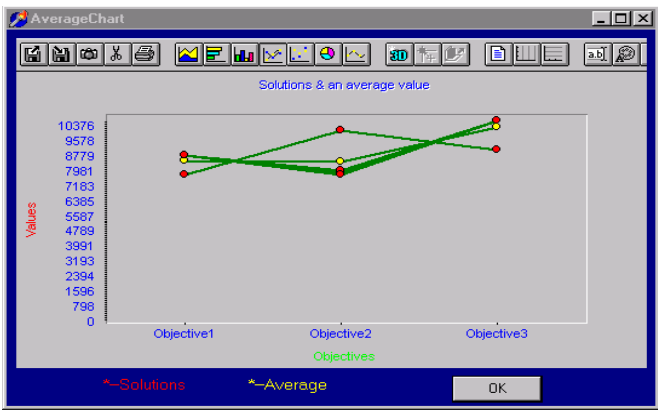


Fig. 12.7: A graphical display for a group of solutions with an average solution

Example: Production planning

Now we suppose a decision group has four members for making the product planning described in Section 2.3.4. They used the IMOGDSS and obtained individual optimal solution (optimal objective function values) for this problem. Fig. 12.8 shows the four members' solutions, for example, Solution 1 (member 1's solution) is (8800.00, 7800.777, 10660.194).

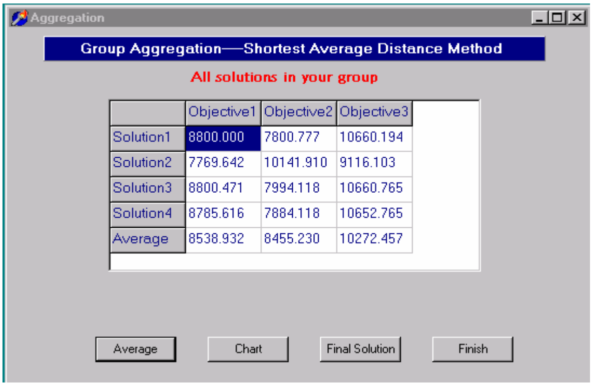


Fig. 12.8: A digital display for a group of solutions with an average solution

The group determines to use the shortest average distance method, that is, ASM to get a solution for this group. The average solution is obtained (8538.932, 8455.250, 10272.457) as shown in Fig. 12.8.

Calculation result shows that Solution 3 has the 'shortest distance' to the average solution. It is therefore selected as the final solution for the group for the product planning problem.

12.6 Summary

Many multi-objective decisions are often taken in a group environment, which is called the *multi-objective group decision making* (MOGDM). We focused on the development of MOGDM methods and system in this chapter. Under a three-stage framework of MOGDM, we presented five multi-objective group aggregation methods. This framework and the five methods have been implemented in an IMOGDSS with the design and support of an intelligent guide subsystem.

This page intentionally left blank

Chapter 13

Fuzzy Multi-Objective Group DSS

Combining fuzzy multi-objective decision making with group decision making methods, we will present a method and a system to solve fuzzy multi-objective linear programming (FMOLP) problems in a group. We also use a case-based example to illustrate how an FMOLP problem is solved in a group supported by a DSS.

13.1 A Decision Method

As discussed in Chapter 6, an FMOLP problem can be formulated as follows:

$$\text{(FMOLP)} \begin{cases} \max & \tilde{f}(x) = \tilde{C}x \\ \text{s.t.} & x \in X = \left\{ x \in R^n \mid \tilde{A}x \precsim \tilde{b},\, x \geq 0 \right\} \end{cases} \qquad (13.1.1)$$

where $\tilde{C}$ is a $k \times n$ matrix, $\tilde{A}$ is an $m \times n$ matrix, $\tilde{b}$ is an m-vector, and x is an n-vector of decision variables, $x \in R^n$. Here, all parameters of objective functions and constraints in (13.1.1) are fuzzy numbers.

When this problem is solved in a group, we call it as *fuzzy multi-objective group decision making* (FMOGDM). Similar as the MOGDM framework shown in Fig. 12.1, the working process for the FMOGDM is split into three stages: the initialisation, the individual solution, and the group consensus solution. We use an FMOLP problem to describe the decision method.

Stage 1: Initialisation

Step 1: Set up a decision group. This work includes determining group members, clarifying a group decision problem, and formulating the problem into an FMOLP model (decision variables, objective functions, and constraints).

Step 2: Input the FMOLP problem, including objective functions, constraints, and membership functions of fuzzy parameters in these objective functions and constraints.

Stage 2: Generating individual solution

Step 3: According to the understanding and preference to the problem, group members generate their own solutions to the FMOLP problem by using any FMOLP method.

Step 4: All group members report their own solutions to the group.

Stage 3: Generating group consensus solution

Step 5: The group leader collects group members' solutions to the FMOLP problem. These solutions are as alternatives for the following group decision making.

Step 6: Each group member including the group leader proposes one or more criteria for assessing these alternatives. All criteria are put into a criterion pool and top-t criteria are chosen as assessment-criteria used for finding the group satisfactory solution.

Step 7: Each group member expresses an opinion to the assessment-criteria by pairwise comparison of the relative importance of these criteria. Each member has a criteria comparison matrix. The comparison scale between each criterion is described as linguistic terms by means of fuzzy numbers (also see Table 10.2).

Step 8: Each group member expresses their opinion to the alternatives with respect to each criterion. This can be carried out by introducing a belief level that represents the possibility of selecting a solution under a criterion. Then, a belief level matrix is generated from each group member. Also, the belief level is associated with a set of linguistic terms that contain various degrees of preferences to the alternatives under the assessment-criterion. These linguistic terms are represented by fuzzy numbers (also see Table 10.3).

Step 9: By aggregating the information in the criteria matrix and the belief level matrix, the preference ranking of each group member to the alternatives is obtained.

Step 10: The group leader assigns each group member with a weight that is described by a linguistic term. Fuzzy numbers (as shown in Table 10.1) are used to deal with these linguistic terms.

Step 11: The ranking to the alternatives in the group is generated by combining all group members' the preference ranking to the alternatives in Step 9 with the different weights of group members in Step 10. Consequently, a group compromise solution to the FMOLP problem is obtained as the alternative with the top rank.

Step 12: If the consensus to the solution is reached at this stage, the whole group decision-making process stops. Otherwise, the process will go back to modify some opinions in the group level for getting the consensus solution for the FMOLP problem.

Figure 13.1 shows the working process of the proposed FMOGDM method. Obviously, this method combines the FMODM method and the FGDM method. The individual solution for an FMODM problem is as an alternative in the group decision making.

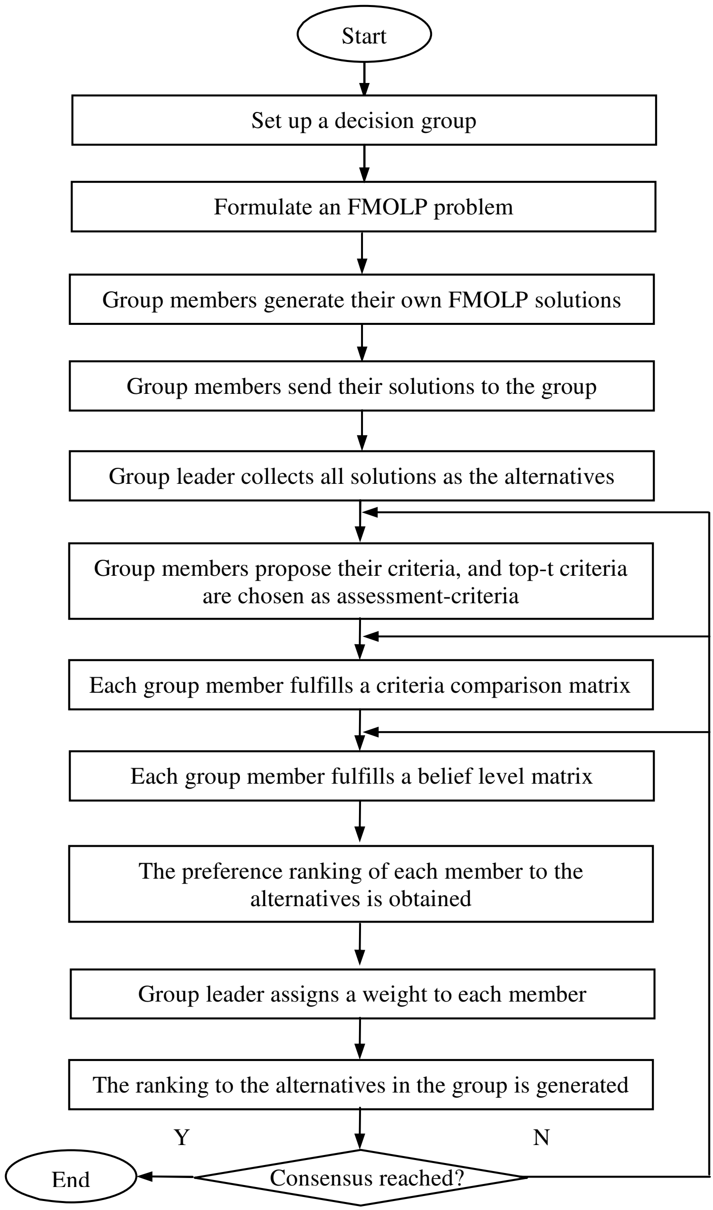


Fig. 13.1: Working process of the FMOGDM method

13.2 System Configuration

An FMOGDSS is developed by implementing the methods presented in Section 13.1 for solving FMOLP problems under a group environment. The FMOGDSS consists of five major software components: (1) Input-and-display management component, (2) Model management component, (3) Method management component, (4) Aggregation component, and (5) Data management component. It also has four bases: (a) database, (b) FMOLP method-base, (c) FGDM method-base, and (d) model-base. These bases are linked to the corresponding management components, respectively. Fig. 13.2 shows the structure of the FMOGDSS.

With the first component, input data includes information about the decision group, the FMOLP models, the alternative definition, the assessment criteria, the criteria matrix, and the belief level matrix, *etc*.; output data includes the individual solutions to the FMOLP problem and the group satisfying solution to the FMOLP problem, *etc*. These input/output data with the initial, intermediate, and final data during algorithms/methods running will be stored in the database by the data management component.

A model management component is combined with a model-base in the system. The model-base is used for storing user application models. The model management component is to define and structure an FMOLP problem and to generate users' decision- making models based on their data input for the further processing. This component has functionalities to build a new model, open an existing model stored in the model-base, or store the current model to the model-base for the further use or modification. Generally, the model management component is connected with the database and the data management component. It also links to the input-and-display management component.

An FMOLP method-base has three methods: FMOLP, FMOLGP, and IFMOLGP (presented in Chapter 7). Depending on the selection of a suitable method from group members for solving their decision problems, the method management component picks up the method from the FMOLP method-base and retrieves the related data from the database. The results from the component will be also stored in the database, and be displayed by the input-and-display management component.

An aggregation component is combined with an FGDM method-base in the system. Currently, an FGDM method is implemented and stored in the FGDM method-base. Based on the FMOLP solutions from decision makers as the alternatives, by combining all decision makers' preferences and opinions to the alternatives, the aggregation component will generate the ranking to the alternatives. The results will be also stored in the database and be displayed for the further group consensus decision.

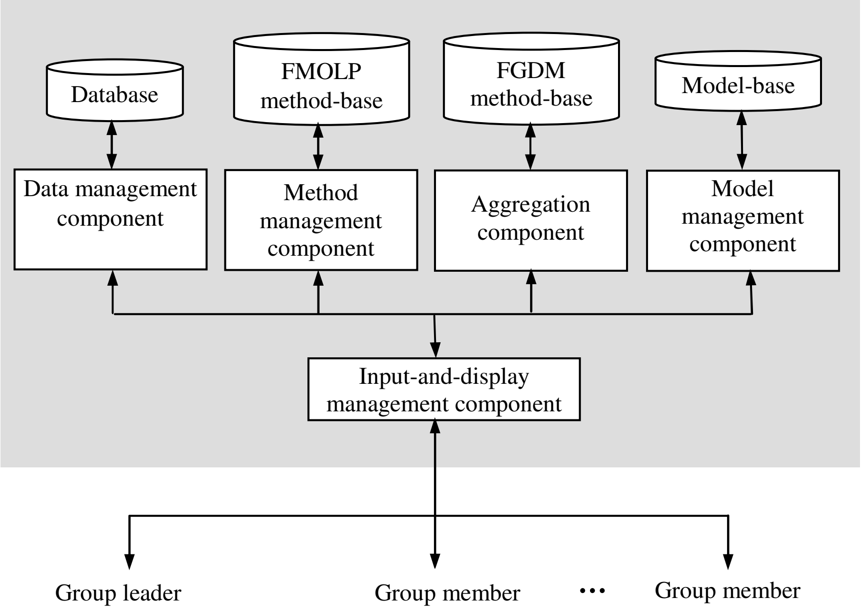


Fig. 13.2: The structure of the FMOGDSS

13.3 System Interface

The interface of the FMOGDSS consists of a system desktop with a pull-down menu bar at the top. There are five menus that form the functions of the user interface: *File*, *Individual Decision Making*, *Group Decision Making*, *Display,* and *About.*

Following the three stages of the FMOGDM method described, the system interface is basically split into three parts as follows:

(1) Initialisation

By clicking the item of *New Application* in the *File* menu, a procedure starts for setting up a new decision group for solving an FMOLP problem. A window pops up for defining a decision-making group. The group title, the issue description and the number of group members are entered. Then another window is shown in Fig. 13.3 for entering the name of all group members.

By clicking the item of *New FMOLP Model* in the *Individual Decision Making* menu, we can input a new FMOLP problem. Based on the FMOLP model (13.1.1), a window shown in Fig. 13.4 is to input the number of decision variables, the number of fuzzy objective functions, and the number of fuzzy constraints. The parameters of fuzzy objective functions and fuzzy constraints also need to be input for set up the FMOLP problem. Fig. 13.5 shows the general information about the FMOLP problem.

(2) Individual solution

By clicking the item of *FMOLP* in the *Individual Decision Making* menu, a window as Fig. 13.6 is used for each group member including the group leader to generate solutions to the FMOLP problem. Each group member can choose one of the three methods from the window. Click on the *Run* button with the choice of a method, one of the three windows, which is as Fig. 13.7, Fig. 13.8 or Fig. 13.9, will show up. These three windows are for the FMOLP method, the FMOLGP method, and the IFMOLP method, respectively. Each group member can generate a solution to the FMOLP problem by using one of the three windows. The details about how to use these functions are already described in Section 8.4.

The solutions generated by group members with Fig. 13.7, Fig. 13.8, or Fig. 13.9 will be sent back to the main FMOLP window as Fig. 13.6, and are displayed. Click on the *Add* button, the current solution will be put into a solution pool.

(3) Group consensus solution

Based on the alternatives generated from Stage 2, each group member can propose several criteria for selecting an alternative as the group satisfactory solution. By clicking the item of *Individual Criteria* in the *Group Decision Making* menu, a window is shown for each group member to input the number of criteria. Click the *Continue* button, another window shown in Fig. 13.11 is to input the criteria. After all group members have finished their criteria input, all proposed criteria are then collected and put into a criteria pool.

By clicking the item of *Criteria and Weights* in the *Group Decision Making* menu, a window is shown in Fig. 13.12 for the group leader to assign weights for group members and to choose assessment-criteria for evaluating the alternatives. In the window, each member is set to a weight by the leader with a linguistic term.

After getting assessment-criteria, each member can express opinions to these assessment-criteria by pairwise comparison of the relative importance of them. Also each member can express opinions to the alternatives with respect to each criterion. By clicking the item of *Individual Preference* in the *Group Decision Making* menu, a window is shown in Fig. 13.13 for this purpose.

Finally, by clicking the item of *solution* in the *Group Decision Making* menu, a window is shown in Fig. 13.14 for displaying the solution of the group. Obviously, it is generated by fully considering the weights to group members and their preferences.

13.4 A Case-Based Example

In this section, we consider the production-planning problem (Example 1 in Section 8.5), which can be described as the following FMOLP model:

$$\max \begin{pmatrix} \tilde{f}_1(x) \\ \tilde{f}_2(x) \\ \tilde{f}_3(x) \end{pmatrix} = \max \begin{pmatrix} \tilde{50}\, x_1 + \tilde{100}\, x_2 + \tilde{17.5}\, x_3 \\ \tilde{92}\, x_1 + \tilde{75}\, x_2 + \tilde{50}\, x_3 \\ \tilde{25}\, x_1 + \tilde{100}\, x_2 + \tilde{75}\, x_3 \end{pmatrix} \tag{13.4.1}$$

$$\text{s.t.} \begin{cases} \tilde{g}_1(x) = \tilde{12}\,x_1 + \tilde{17}\,x_2 \preceq \tilde{1400} \\ \tilde{g}_2(x) = \tilde{3}\,x_1 + \tilde{9}\,x_2 + \tilde{8}\,x_3 \preceq \tilde{1000} \\ \tilde{g}_3(x) = \tilde{10}\,x_1 + \tilde{13}\,x_2 + \tilde{15}\,x_3 \preceq \tilde{1750} \\ \tilde{g}_4(x) = \tilde{6}\,x_1 + \tilde{16}\,x_3 \preceq \tilde{1325} \\ \tilde{g}_5(x) = \tilde{12}\,x_2 + \tilde{7}\,x_3 \preceq \tilde{900} \\ \tilde{g}_6(x) = \tilde{9.5}\,x_1 + \tilde{9.5}\,x_2 + \tilde{4}\,x_3 \preceq \tilde{1075} \\ x_1, x_2, x_3 \geq 0 \end{cases}$$

In this model, the unified form for all membership functions of fuzzy parameters of the objective functions and constraints is as follows:

$$\mu_{\tilde{a}}(x) = \begin{cases} 0 & x < a \text{ or } c < x \\ (x^2 - a^2)/(b^2 - a^2) & a \leq x < b \\ 1 & x = b \\ (c^2 - x^2)/(c^2 - b^2) & b < x \leq c \end{cases} \tag{13.4.2}$$

All membership functions of fuzzy parameters of the FMOLP model (13.4.1) are listed in the triple pair form in Tables 13.1 and 13.2, respectively.

Table 13.1: Membership functions of fuzzy objective functions' parameters

$\tilde{c}_{ij}$	1	2	3
1	(45, 50, 55)	(90, 100, 110)	(15.75, 17.5, 17.25)
2	(82.8, 92, 101.2)	(67.5, 75, 82.5)	(45, 50, 55)
3	(22.5, 25, 27.5)	(90, 100, 110)	(67.5, 75, 82.5)

Table 13.2: Membership functions of fuzzy constraints' parameters

$\tilde{a}_{ij}$	1	2	3	$\tilde{b}_i$
1	(12.8, 12, 13.2)	(15.3, 17, 18.6)	(0, 0, 0)	(1260, 1400, 1540)
2	(2.7, 3, 3.3)	(8.1, 9, 9.9)	(7.2, 8, 8.8)	(900, 100, 110)
3	(9, 10, 11)	(11.7, 13, 14.3)	(13.5, 15, 16.5)	(1575, 1750, 1925)
4	(5.4, 6, 6.6)	(0, 0, 0)	(14.4, 16, 17.6)	(1192.5, 1325, 1457.5)
5	(0, 0, 0)	(10.8, 12, 13.2)	(6.3, 7, 7.7)	(810, 900, 990)
6	(8.55, 9.5, 10.45)	(8.55, 9.5, 10.45)	(3.6, 4, 4.6)	(967.5, 1075, 1182.5)

Based on the FMOGDM method given in Section 13.1, the procedure for solving the production-planning problem in a group is as follows:

Stage 1: Initialisation

Step 1: Set up a decision group, with *Peter* as the group leader and *David* and *Kim* as group members (Fig. 13.3).

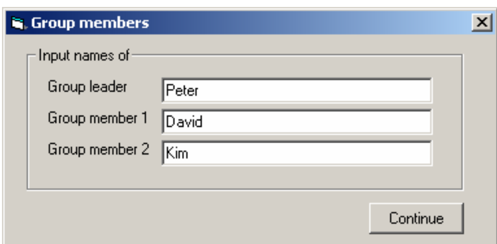


Fig. 13.3: Set up a decision group

Step 2: Set up the FMOLP problem (Fig. 13.4) and its model (13.4.1) as shown in Fig. 13.5.

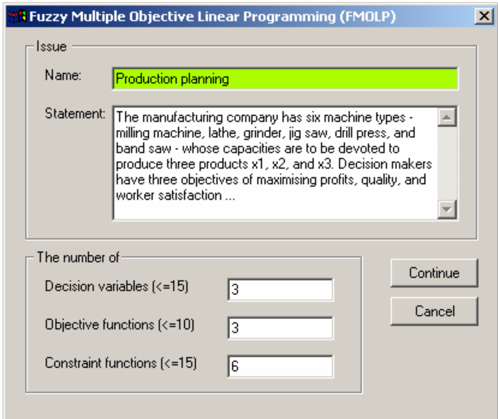


Fig. 13.4: Set up the ‘*Production-planning*’ problem

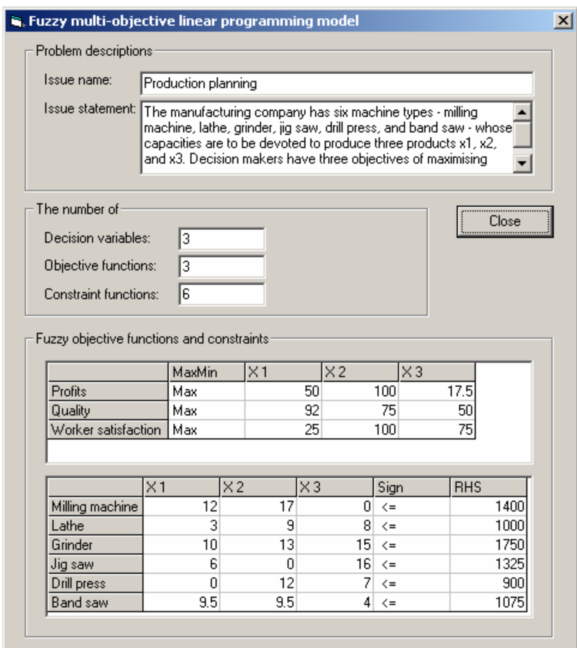


Fig. 13.5: General information about the ‘*Production-planning*’ problem

Stage 2: Generating FMOLP solutions at the individual level

Step 3: Suppose the three group members, *Peter*, *David*, and *Kim*, generate their solutions to the FMOLP problem using FMOLP method, FMOLGP method, and IFMOLP method, respectively as shown in Figs. 13.7, 13.8, and 13.9. The results are collected as in Fig. 13.6. That is, the solution generated by *Peter* is $x_1^* = 68.7569$, $x_2^* = 25.4063$, and $x_3^* = 45.1126$; the *David*'s solution is $x_1^* = 37.5281$, $x_2^* = 55.3371$, and $x_3^* = 33.7079$; and the *Kim*'s solution is $x_1^* = 44.8217$, $x_2^* = 50.5013$, and $x_3^* = 41.4878$. Obviously, they have different solutions.

Step 4: Each group member sends their own solutions to the group level for the further group decision making.

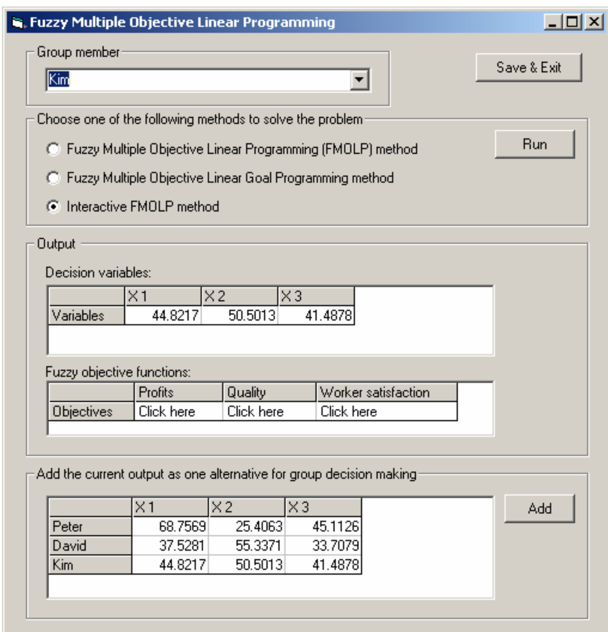


Fig. 13.6: The FMOLP window for all group members

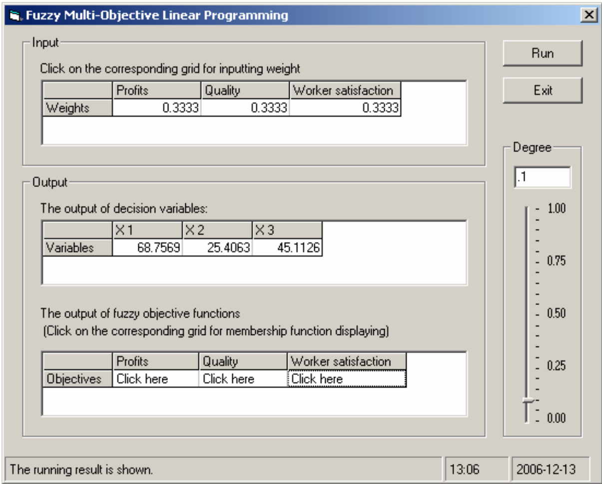


Fig. 13.7: The window for the FMOLP method

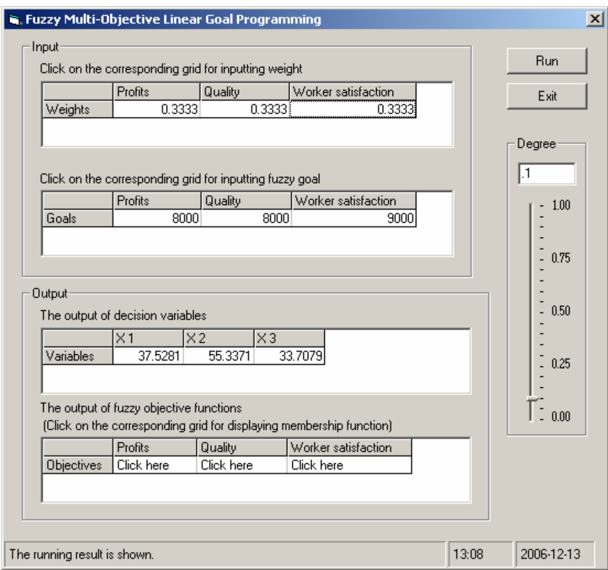


Fig. 13.8: The window for the FMOLGP method

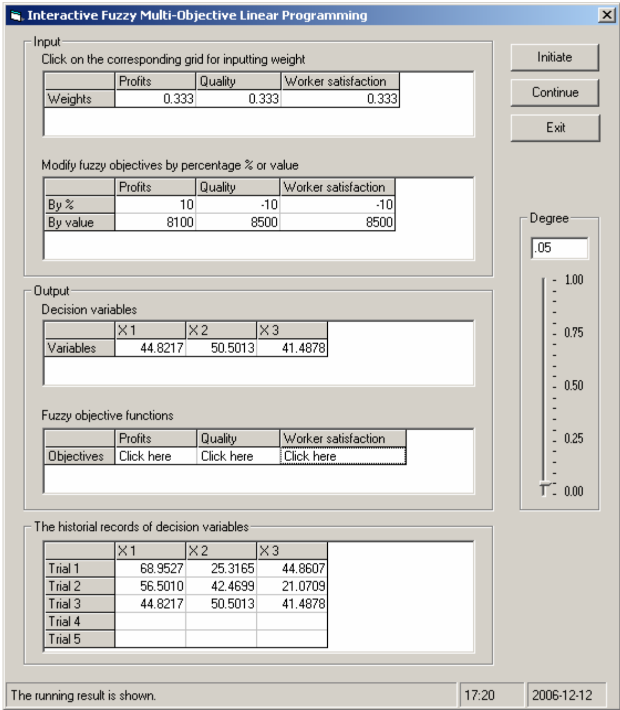


Fig. 13.9: The window for the IFMOLP method

Stage 3: Generating the group consensus solution at the group level

Step 5: The group leader, *Peter*, collects group members' solutions as the group alternatives. These alternatives are summarised in Fig. 13.10. The plan proposed by *Peter*, $x_1^* = 68.7569$, $x_2^* = 25.4063$, and $x_3^* = 45.1126$, is as *alternative 1*; the *David*'s plan, $x_1^* = 37.5281$, $x_2^* = 55.3371$, and $x_3^* = 33.7079$, is as *alternative 2*; and the *Kim*'s plan $x_1^* = 44.8217$, $x_2^* = 50.5013$, and $x_3^* = 41.4878$, is as *alternative 3*.

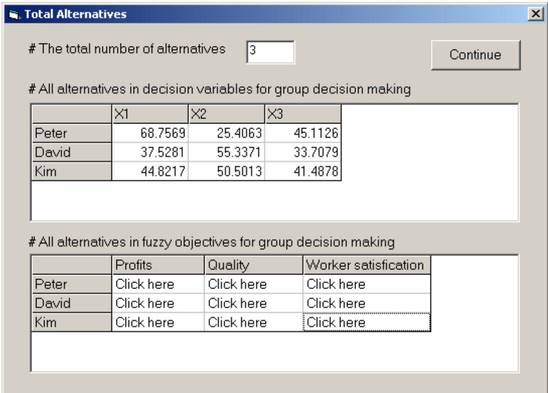


Fig. 13.10: All alternatives generated by group members

Step 6: Based on the FMOLP solutions (alternatives), each group member proposes one or more criteria for getting a group satisfactory solution. Suppose *Peter* proposes two criteria, which are '*Profit*' and '*Pollution*' as shown in Fig. 13.11. *David* also proposes two criteria, '*Quality*' and '*Worker satisfaction*,' and *Kim* proposes '*Environment*' as one more criterion. Then these five proposed criteria are collected into a criteria pool.

Step 7: The group leader, *Peter*, assigns '*important*,' '*normal*,' and '*most important*' to *David*, *Kim*, and himself, respectively, and choose four criteria as assessment-criteria from the criteria pool for evaluating the alternatives as shown in Fig. 13.12.

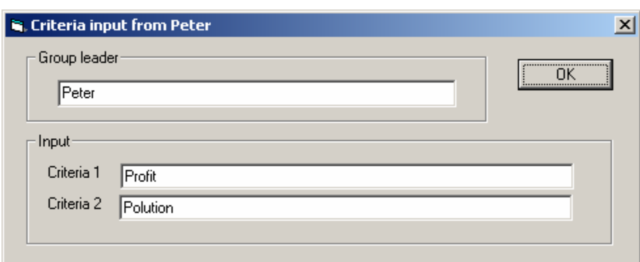


Fig. 13.11: Input assessment-criteria

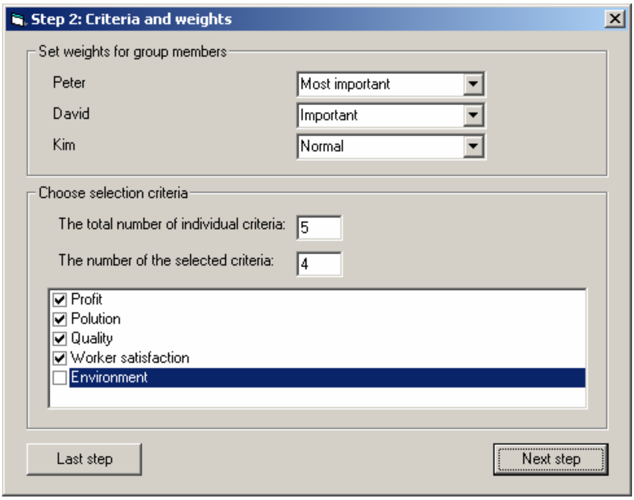


Fig. 13.12: Choosing weights and criteria

Step 8: Each group member inputs a criteria comparison matrix with pairwise comparisons of the relative importance of these criteria similar as in Fig. 13.13.

Step 9: Each group member inputs a belief level matrix with the preference to the alternatives against each criterion similar as in Fig. 13.13.

Step 10: The ranking to the alternatives in the group is obtained as in Fig. 13.14, and the second alternative, proposed by *David*, has the highest rank.

Finally, the group reaches the consensus to the solution for this production-planning problem as $x_1^* = 37.5281$, $x_2^* = 55.3371$, and $x_3^* = 33.7079$. That is, 37.5281 units of product x_1, 55.3371 units of product x_2, and 33.7079 units of product x_3 will be produced, respectively. The *profit* $\tilde{f}_1^*$ is about 8000 units, the *quality* $\tilde{f}_2^*$ is about 9288 units, and the *worker satisfaction* $\tilde{f}_3^*$ is about 9000 units. The membership functions of the fuzzy objective functions $\tilde{f}_1^*$, $\tilde{f}_2^*$ and $\tilde{f}_3^*$ are shown in Fig. 13.15, respectively.

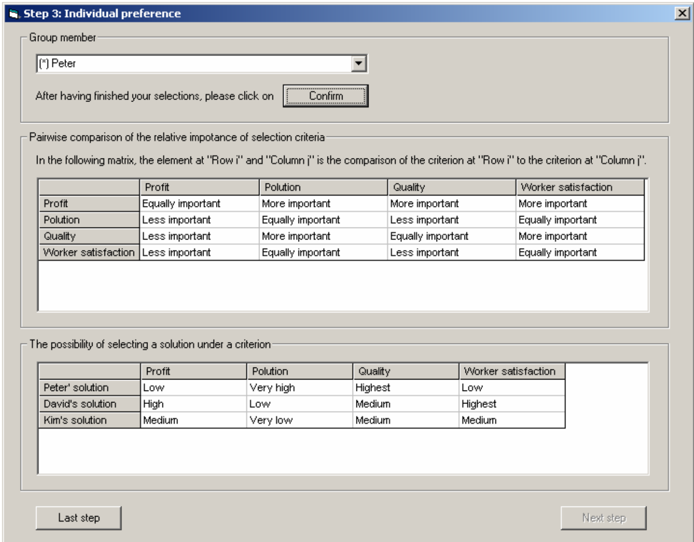


Fig. 13.13: Criteria comparison matrix and belief level matrix

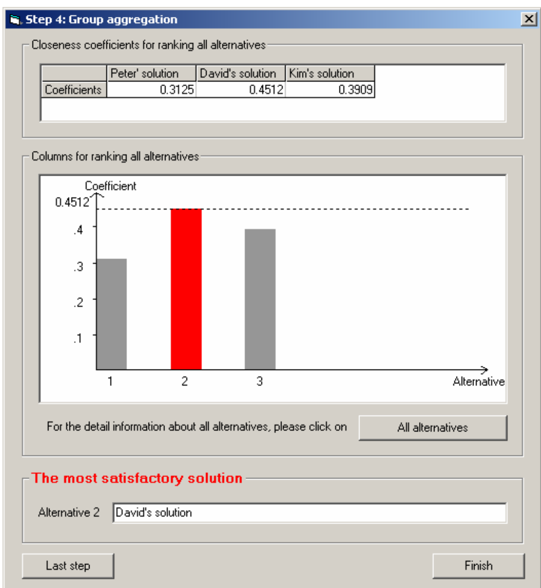


Fig. 13.14: Alternative ranking

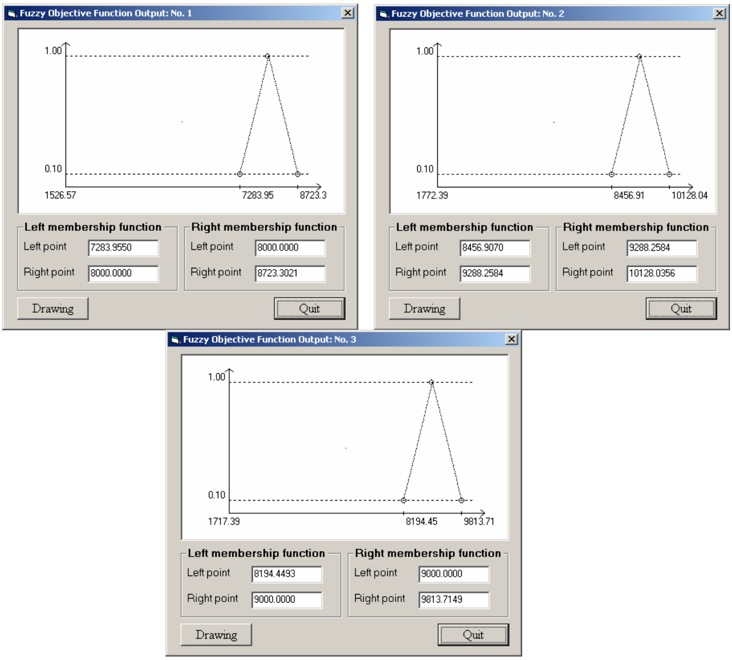


Fig. 13.15: Membership functions of the fuzzy objective functions for the solution

13.5 Summary

The procedure for a group to make a decision on fuzzy multi-objective problem needs three stages, in general. After the initialisation, each group member first generates an individual solution, based on their preference, supported by FMOLP methods. All group members' solutions are then aggregated to get a group solution through a multi-criteria decision-making method. This FMOGDSS is an extension of FMODSS from individuals to groups. It is also a special case of group decision support systems.

Part V

Applications

This page intentionally left blank

Chapter 14

Environmental Economic Load Dispatch

The increasing energy demand and decreasing energy resources have necessitated the optimum use of available resources. Economic dispatch is the optimisation scheme intended to find the generation outputs that minimise the total fuel cost subject to several unit and system constraints. We first present a novel environmental economic load dispatch model, which has the cost and emission objective functions with uncertain parameters. We then convert the model into a single objective optimisation problem, and develop a hybrid genetic algorithm with quasi-simplex techniques to solve the corresponding single objective optimisation problem. Finally, we validate the model and the effectiveness of the algorithm for a real economic dispatch problem.

14.1 The Problem

The conventional economic dispatch problem mainly concerns the minimisation of operating cost subject to diverse units and system constraints. Recently, the environmental pollution problem caused by electricity generation has been proposed and discussed by both industry managers and researchers. How to decrease the emission of maleficent gases has become an important issue in the electricity generation.

Some related feasible strategies have been proposed to reduce the atmospheric emissions. These include (1) installation of pollutant cleaning equipments, (2) switching to low emission fuels, (3) replacement of the aged fuel-burners, and (4) generator units, and emission dispatching. The first three strategies should be as long-term options. The emission dispatching option is an attractive short-term alternative. In fact, those three options should be determined by the generation companies, but not by regulation departments, especially in the environment of the power market. As the aim to pursue in a long run is to reduce the emission of harmful gases, we should reduce the emission of maleficent gases of the generation companies by regulating principles. Therefore, the environmental economic load dispatch problem considering the emission of harmful gases becomes a key issue in the power market.

Many uncertain factors are involved in modelling the cost and emission functions for environmental economic load dispatch problems. However, most existing environmental economic load dispatch models lack a suitable consideration for the uncertainty issue. We therefore apply fuzzy number to represent uncertain values of parameters to build an environment economic load dispatch model, called a *fuzzy dynamic environmental economic load dispatch* (FDEELD) model. To get an optimal solution from the model, a weighting ideal point method (WIPM) is proposed. The WIPM converts the FDEELD model into a single objective fuzzy non-linear programming problem. A hybrid genetic algorithm with quasi-simplex techniques is then developed to seek optimisation solutions for the single objective non-linear programming problem. A fuzzy number ranking method is applied to compare the fuzzy function values of different points for the single objective function to obtain the optimal solution for the FDEELD problem.

14.2 A Fuzzy Dynamic Model

The basic structure of the power market presented in the literature (Watts *et al.*, 2002) consists of *Power Exchange* (PX) and *Independent System Operator* (ISO). In this market structure, PX is in charge of the spot trade, the economic load dispatch is the main task of the PX and ISO takes responsibility for network security and auxiliary service. Therefore, the load dispatch model may neglect the network constraints and spinning reserve. But it must consider the ramp rate limit in order to assure an optimum solution. As the parameters of cost and emission functions are with uncertainties, they are denoted by fuzzy numbers. The proposed FDEELD model is described as follows:

$$
\begin{aligned}
&\min_{P_j(t)\in R} \mathrm{c} = \sum_{t=1}^{T}\sum_{j=1}^{N}\left(\tilde{a}_j + \tilde{b}_j P_j(t) + \tilde{c}_j P_j^2(t)\right) \\
&\min_{P_j(t)\in R} e = \sum_{t=1}^{T}\sum_{j=1}^{N}\left(\tilde{\alpha}_j + \tilde{\beta}_j P_j(t) + \tilde{\gamma}_j P_j^2(t)\right) \\
&\text{s.t.}\quad \sum_{j=1}^{N} P_j(t) = P_D(t) + P_L(t) \\
&\qquad\ 0 \le P_{jlow}(t) \le P_j(t) \le P_{jhigh}(t)
\end{aligned}
\tag{14.2.1}
$$

where T is the number of time segments; N is the number of committed units; $P_j(t)$ is the output active power of the unit j at the time segment t; c is a cost (fuel) function, $\tilde{a}_j, \tilde{b}_j, \tilde{c}_j$ are fuzzy parameters of the cost function of unit j; e is an emission function, $\tilde{\alpha}_j, \tilde{\beta}_j, \tilde{\gamma}_j$ are fuzzy parameters of the emission function of the unit j; $P_{j\min}$ and $P_{j\max}$ are minimum and maximum outputs of the unit j, respectively; $P_D(t)$ is a load demand at the time segment t; $P_L(t)$ is network loss at time segment t; D_j is a down ramp rate limit of the unit j, R_j is an up ramp rate limit of the unit j. We also define

$$P_{jlow}(t) = Max\{P_{j\min}, P_j(t-1) - D_j\} \tag{14.2.2}$$

$$P_{jhigh}(t) = Min\{P_{j\max}, P_j(t-1) + R_j\} \tag{14.2.3}$$

Obviously, the FDEELD model is a fuzzy bi-objective non-linear programming problem. In Section 14.5, we will solve a real-case problem, with $T = 24$ and $N = 7$, by a proposed algorithm.

14.3 A Transformation Method

Both the weighting and reference point methods are effective in finding the Pareto optimal solutions for multi-objective non-linear programming problems. Strictly speaking, the *weighting method* only represents the relative importance of the goal values of an objective rather than of different objectives. It is hard to know the magnitude of effect of the set of weights to each objective function value. The *reference point method* is a relatively practical interactive approach to multi-objective optimisation problems. It introduces the concept of a reference point suggested by decision makers that present in some desired values of the objective functions. It is very hard to determine weights and reference points in applications, besides the interactive approach increases heavily computing burden. A *weighting ideal point method* (WIPM), proposed here, does not require any interaction, and can predict the effect magnitude the weights to each objective function value.

To describe the proposed WIPM method, we write a general multi-objective non-linear programming problem as

$$\min_{x\in S} f(x) = (f_1(x), f_2(x), \cdots, f_k(x)) \tag{14.3.1}$$

where $f_1(x), \cdots, f_k(x)$ are k distinct objective functions and S is the constraint set defined by

$$S = \{x \in R^n \mid g_j(x) \le 0, j = 1, \cdots, m\} \tag{14.3.2}$$

Let

$$g(x) = w_1 (\frac{f_1 - f_1^{\min}}{f_1^{\min}})^2 + \cdots + w_k (\frac{f_k - f_k^{\min}}{f_k^{\min}})^2 \tag{14.3.3}$$

where $f_i^{\min} = \min_{x\in S} f_i(x)$, $f_i^{\min} \ne 0$, $i = 1, 2, \cdots, k$. $f^{\min} = (f_1^{\min}, \cdots, f_k^{\min})$ is so-called an *ideal* or *utopia* point, $w = (w_1, \cdots, w_k) > 0$, $\sum_{i=1}^{k} w_i = 1$ is a weight vector.

To get the Pareto optimal solution, it can be transformed to solve the single objective optimisation problem (14.3.3) below:

$$\min_{x\in S} g(x) \tag{14.3.4}$$

Since the values of different objective functions in (14.3.1) can be very different, it is hard to know the effect magnitude of the weights to each objective function value. In the model (14.3.4), all objectives are converted into the same level by using the formula

$$\frac{f_i - f_i^{\min}}{f_i^{\min}}.$$

We can therefore predict the effect quantity of the weights to each objective function value. For example, if $w_1 = 2w_2$, then

$$\frac{f_2^* - f_2^{\min}}{f_2^{\min}} \approx 2\frac{f_1^* - f_1^{\min}}{f_1^{\min}}$$

where $f_i^* = f_i(x^*)$, $i = 1,2$, x^* is the optimal solution. In other words, the weights given in WIPM can reflect the trade-off rate information among the values of objective functions. When the parameters of non-linear objective functions are fuzzy numbers, we also use the model (14.3.4) to convert (14.3.1) into a corresponding single objective fuzzy optimisation problem.

14.4 A Solution Technique

To solve the single objective problem (14.3.4), a hybrid genetic algorithm with quasi-simplex techniques is proposed. Simplex method is one of the widely accepted conventional direct search methods. A simplex in an n-dimensional space is defined by a convex polyhedron consisting of $n+1$ vertices, which are not in the same hyper-plane. Assume that there are $n+1$ points in an n-dimensional space, denoted by x_i and the objective function values over these points are denoted by f_i, $i = 1, ..., .n+1$. The worst and the best points in terms of function values are denoted by x^H and x^B, respectively, and can be determined by

$$f(x^H) = \max_i f_i, \quad i = 1, ..., .n+1, \tag{14.4.1}$$

and

$$f(x^B) = \min_i f_i, \quad i = 1, ..., .n+1. \tag{14.4.2}$$

The quasi-simplex technique uses two search directions in generating prospective offspring. One direction is the worst-opposite direction, which is used in the conventional simplex techniques, and the other is the best-forward direction, which is a ray from the centroid of a polyhedron whose vertexes are all the points but the best one towards the best point of the simplex. Along the worst-opposite direction, four individuals will be generated by using the reflection, expansion and compression operations, respectively, and can be determined by

$$x = x^C + \alpha(x^C - x^H) \tag{14.4.3}$$

where x^C is the centroid and can be calculated by

$$x^C = \left[\left(\sum_{i=1}^{n+1} x^i \right) - x^H \right] \Big/ n, \tag{14.4.4}$$

α is a parameter whose value determines the position of the corresponding potential better point along this direction. For example, $\alpha = 1$, $\alpha > 1$, $-1 < \alpha < 0$, and $0 < \alpha < 1$ are corresponding with the reflection point x^R, the expansion point x^E, compression points x^M and x^N, respectively.

Along the best-forward direction, four individuals x^e, x^r, x^m and x^n, will be calculated by the operations that are similar to the expansion, reflection, and compression operations used in the conventional simplex method by the following formula

$$x = x^B + \beta(x^B - x^D) \tag{14.4.5}$$

where x^D denotes the centroid of the polyhedron whose vertexes are all the best points and can be calculated by

$$x^D = \left(\left(\sum_{i=1}^{n+1} x^i \right) - x^B \right) \Big/ n \tag{14.4.6}$$

β is a parameter whose value determines the position of the corresponding calculated point on the best-forward line. The four prospective points along the best-forward direction are with the corresponding value ranges of $\beta > 1$, $\beta = 1$, $-1 < \beta < 0$, and $0 < \beta < 1$, respectively.

In contrary to conventional optimisation method, genetic algorithms (GA) have a strong global search capability and a weak local search ability. To increase search performance, it is common to combine GA with conventional optimisation methods. Based on this idea, a new optimisation method, hybrid GA with quasi-simplex techniques (GAQST) is proposed to solve general single non-linear optimisation problems with the following iteration steps:

Step 1: Initialise a population of size $\mu = K(n+1)$.

Step 2: Evaluate the fitness values for each individual x_i of the population based on the objective function $f(x_i)$.

Step 3: Subdivide the population into K subpopulations.

Step 4: For each subpopulation, create an offspring by genetic operations and quasi-simplex techniques in parallel. In order to increase subpopulation varieties, select respectively the best point as offspring from the points obtained by (14.4.3) and (14.4.5). The rest offspring of subpopulation are created by reproduction, crossover and mutation operations.

Step 5: Unite all offspring created in Step 4 to form a new generation population.

Step 6: Stop the iteration if the termination criterion is satisfied, and an optimal solution is obtained. Otherwise, go back to Step 2.

Through these steps, the proposed GAQST can solve a general single objective non-linear optimisation problem

$$\min_{x \in R^n} g(x). \tag{14.4.7}$$

14.5 A Case Study

By combining the proposed WIPM, GAQST and fuzzy number ranking methods, we present an approach to solve the environmental economic load dispatch problem, which is described by a bi-objective fuzzy non-linear programming (14.2.1). Here, we have seven committed units, and the number of time segment is 24. Firstly, we convert the FDEELD into the single objective optimisation problem by using WIPM. Secondly, use the Lagrange relaxation method to form a Lagrange function. Finally, use GAQST to optimise the Lagrange function. In the process of the iteration, the fuzzy number ranking method is used to compare fuzzy function values of different points for the single objective function.

By the Lagrange relaxation method, a *penalty function* (h) can be written as

$$h\left(P_j(t)\right) = w_1\left(\frac{c - c^{\min}}{c^{\min}}\right)^2 + w_2\left(\frac{e - e^{\min}}{e^{\min}}\right)^2 + M\max\left\{0, P_{jlow} - P_j(t)\right\} + M\max\left\{0, P_j(t) - P_{jhigh}\right\} \tag{14.5.1}$$

where, $t = 1, 2 \ldots, 24, \quad j = 1, 2 \ldots, 7.$

Tables 14.1 to 14.3 show the data of the units output, the fuzzy parameters of cost (fuel) function (c), and the fuzzy parameters of emission function (e), respectively, for the environmental economic load dispatch problem. We use triangular fuzzy numbers to describe these fuzzy parameters, which are obtained form a set of experiments.

Table 14.1: Limits of unit output and ramp rate

Unit No. (j)	$P_{j\min}(MW)$	$P_{j\max}(MW)$	D_j	R_j
1	20	125	40	30
2	20	150	40	30
3	35	225	50	40
4	35	210	50	40
5	130	325	60	50
6	120	310	60	50
7	125	315	60	50

Table 14.2: Fuzzy parameters of the cost (fuel) function

Unit No. (j)	$\tilde{a}_j$			$\tilde{b}_j$			$\tilde{c}_j$		
	a_0	a_1	a_2	b_0	b_1	b_2	c_0	c_1	c_2
1	800.95	825.72	846.36	37.46	38.53	39.46	0.168	0.162	0.166
2	625.96	645.32	661.45	41.32	42.51	43.53	0.120	0.122	0.126
3	1107.49	1135.89	1158.61	38.83	39.83	40.62	0.027	0.027	0.026
4	1168.89	1198.86	1222.84	36.90	37.85	38.60	0.034	0.035	0.035
5	1555.00	1586.73	1610.54	36.58	37.32	37.92	0.025	0.025	0.026
6	1269.74	1295.65	1315.09	38.29	39.08	39.70	0.017	0.017	0.017
7	1466.72	1496.65	1519.10	36.52	37.27	37.86	0.020	0.020	0.020

Table 14.3: Fuzzy parameters of the emission function

Unit No. (j)	$\tilde{\alpha}_j$			$\tilde{\beta}_j$			$\tilde{\gamma}_j$		
	α_0	α_1	α_2	β_0	β_1	β_2	γ_0	γ_1	γ_2
1	15.18	15.65	16.04	0.28	0.29	0.29	0.00382	0.00392	0.00400
2	15.18	15.65	16.04	0.28	0.29	0.29	0.00382	0.00392	0.00400
3	34.69	35.58	36.29	-0.54	-0.53	-0.52	0.00698	0.00712	0.00725
4	34.69	35.58	36.29	-0.54	-0.53	-0.52	0.00698	0.00712	0.00725
5	42.04	42.89	43.54	-0.52	-0.51	-0.50	0.00453	0.00461	0.00468
6	40.92	41.76	42.38	-0.53	-0.52	-0.51	0.00464	0.00472	0.00479
7	40.92	41.76	42.38	-0.53	-0.52	-0.51	0.00464	0.00472	0.00479

As $T = 24$, we list all correspondent load demand ($D(t)$) in each time segment t $(t = 1, 2 \ldots, 24)$ in Table 14.4.

Table 14.4: Load demand

t	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*
D(*t*)	690	670	670	680	730	800	870	840	890	920	950	910
t	*13*	*14*	*15*	*16*	*17*	*18*	*19*	*20*	*21*	*22*	*23*	*24*
D(*t*)	890	890	930	970	930	950	1070	1040	950	850	760	730

t--time segment; *D*(*t*)--correspondence load demand

The penalty function h is a high-dimension non-linear function, and therefore it is hard to know where the global minimum point is. To demonstrate the effectiveness of the proposed algorithm, the mean and standard deviation of *fuzzy cost*, *fuzzy emission*, and *fuzzy total value* corresponding to the optimal outputs are tested. In addition, to compare the effect magnitude the weights to fuzzy cost and fuzzy emission, we calculate three group weights. Table 14.5 lists the *means* and *standard* *deviations* of fuzzy cost, fuzzy emission, and fuzzy total cost by the proposed algorithm through running independently 10 times, where MC, ME, and MT present the means of the *cost*, *emission*, and *total*, respectively; STDEV-C, STDEV-E, and STDEV-T present corresponding standard deviations.

Table 14.5: The comparison of the results obtained for different weights

(w_1, w_2)	*MC*	*STDEV-C*	*ME*	*STDEV-E*	*MT*	*STDEV-T*
(0.3, 0.7)	1067359 1092154 1112300	291.4 303.8 312.4	11423.23 11993.81 12539.55	2.2 2.5 2.7	1078780 1104148 1124838	290.7 300.7 310.4
(0.5, 0.5)	1061711 1086218 1106213	472.8 377.1 615.9	11466.7 12041.25 12596.67	4.2 5.3 9.4	1073181 1098320 1118805	468.5 540.6 607.9
(0.7, 0.3)	1053936 1078110 1097695	57 58.3 60	11600.89 12184.12 12744.69	1.3 1.4 1.5	1065537 1090295 1110440	55 58.2 60.9

As the standard deviations of every result are all significantly small, the results are believable. The fuel cost decreases and the emission increase when the weight of the fuel cost is assigned higher.

Table 14.6 shows the optimal power output of units on one run for the weights (0.3, 0.7) for the two objectives of cost and emission. We can see that within 24 time segments, we obtain optimal power output for each of the seven units.

Table 14.6: An optimal power output of units for weights (0.3, 0.7)

$P_j(t)$ / Time segment (t)	Unit number (j)						
	1	2	3	4	5	6	7
1	51.46	53.05	91.02	88.99	136.41	134.07	134.99
2	49.21	49.30	89.04	88.18	131.88	130.08	132.34
3	49.52	49.59	88.67	87.94	132.89	129.43	131.96
4	50.27	50.22	89.99	89.38	133.92	133.12	133.10
5	56.06	61.42	95.36	93.57	141.20	140.42	141.96
6	68.33	71.53	100.77	99.66	153.84	152.07	153.80
7	76.66	82.65	110.36	107.55	165.83	164.45	162.50
8	69.73	78.13	106.59	105.29	160.19	160.29	159.79
9	80.40	86.77	111.39	110.45	168.51	165.29	167.20
10	86.43	90.07	114.66	112.83	174.17	170.07	171.78
11	90.33	92.03	119.42	116.21	181.36	174.44	176.22
12	86.41	87.87	113.15	112.45	171.72	168.67	169.74
13	80.76	85.96	111.19	110.33	169.91	165.58	166.26
14	75.09	88.24	111.12	110.66	169.85	165.97	169.07
15	87.70	89.80	116.03	112.73	176.57	172.51	174.67
16	94.07	98.21	119.83	116.49	183.22	177.82	180.37
17	86.22	90.68	118.52	114.26	175.70	172.11	172.51
18	87.79	93.94	120.20	116.19	181.15	176.30	174.42
19	103.47	111.97	132.23	130.48	200.75	195.36	195.75
20	106.17	104.57	128.55	126.04	194.27	190.44	189.96
21	89.82	92.90	119.69	115.03	181.85	174.92	175.78
22	68.41	81.11	109.17	105.86	163.49	161.03	160.93
23	57.34	66.97	98.25	97.35	147.49	145.48	147.11
24	55.12	62.23	94.79	94.43	141.83	140.30	141.30

Table 14.7 shows the two optimal objective function values described by triangular fuzzy numbers.

Table 14.7: Two objective values described by triangular fuzzy numbers

Objectives	*Cost (c)*	*Emission (e)*
Triangular fuzzy numbers	(1066800, 1091570, 1111700)	(11427.6, 11998.6, 12544.8)

14.6 Summary

A new environmental economic load dispatch model is proposed with a consideration of uncertainty in the parameters of cost and emission functions. It integrates the weighting ideal point method, hybrid genetic algorithm with quasi-simplex techniques and fuzzy number ranking method to solve the optimisation problem with two main advantages: (1) more precisely characterising the cost and emission; and (2) providing more information than real number-based methods. A case study displays the fuzzy multi-objective optimisation problem in details.

Chapter 15

Team Situation Awareness

Situation awareness (SA) is an important element to support making right decisions to crisis problems. The process of achieving SA is performed through situation assessments. As a given situation is normally required to be assessed by a given group, the group SA features including collaboration and information sharing become non-negligible issues. In the meantime, various uncertainties are involved in situation information obtained and awareness generation. Also, when the collaboration is across distances, the Web-based technology can facilitate the form of team SA. The Web-based fuzzy group DSS (FGDSS) from Chapter 11 can support the creation of team SA. We present applications of the Web-based FGDSS to support team SA by introducing the background of SA, identifying three uncertain issues in SA, and demonstrating the working process of the system to support a team SA.

15.1 Situation Awareness

Critical situation management mainly focuses on the immediate aspect of a disaster and its post-disaster recovery. It also pays more attentions on finding ways to avoid crisis problems in the first place if possible and preparing suitable responses to minimise the lose for those that undoubtedly will occur, such as fires, floods, epidemic, and even terrorism. This mission requires technical support in effectively analysing information about a situation, providing awareness information to emergency management officers for understanding the situation, and making suitable decisions. SA has been considered as an important element to help emergency management officers take responses and make decisions for criteria situations correctly and accurately.

Severe acute respiratory syndrome (SARS) is an example of emergency problems. Suppose a group of similar SARS reports is discovered in a region. The health organisation for that region would start a process responsible for understanding the nature of SARS there and containing the outbreak. The process primarily involves interviewing doctors and patients, communication with the World Health Organisation (WHO), and communication with news agencies and doctors involved in containing the outbreak. The whole process includes various types of information processing, such as information gathering, representation, judgment, filtering and integration, and related SARS awareness deriving. Such a situation is also required to be assessed in a group. Reaching a consensus SA in the group is a pre-emptive requirement for a group decision making. Typically, collaboration and information sharing are the main features in a group to achieve awareness for a situation.

15.2 Uncertainty, Inconsistency, and Distributed Environment

Situation awareness is defined by Endsley (1995) "as the perception of elements in the environment, the comprehension of their meaning in terms of task goals, and the projection of their status in the near future." The process of achieving SA is called situation analysis or situation assessment. Situation assessment is based on acquired situation information that can be implicit or explicit. Awareness information (or call SA information) is derived as results of situation assessment. The term SA is commonly used in the human-computer interaction community where the concerns are to design computer interfaces so that a human operator can achieve SA in a timely fashion. It is also used in the data fusion community where it is more commonly referred to as situation assessment. SA has been largely studied as an important element in diverse military and pilot systems using observation, experiments, and empirical methods. There are three main issues that influence situation analysis and awareness deriving to be solved.

(1) *Situation information uncertainty*: there are two basic elements needed to support the generation of SA. The first is the representation of a situation. The second is the approaches or tools for situation assessment. Naturally, in a real world people often only imprecisely or ambiguously know a situation and use uncertain (fuzzy) information to present it. Particularly, some explicit situation information cannot be expressed into precise information. Therefore, SA has to be generated based on imprecise and inaccuracy information through suitable fuzzy information processing.

(2) Team SA inconsistency: Controlling large dynamic systems, such as an emergency co-ordination among several large organisations, is beyond the competence of one single individual. Instead a team works cooperatively to coordinate and control the environment. The degree to which every team member possesses SA for task performance is called team SA. The level of overall SA across the team becomes an important issue, possibly leading to performance errors in team SA. An example of this can be found in the context of a building security control room. Several security personnels need to know certain pieces of information to safely and effectively complete a work process. If one person acting as a supervisor is aware of the critical information, but another person in direct control of the process is not, the SA of the team may be deficient. Consequently, performance and system safety may suffer from this case. This is a typical SA inconsistency situation. Team members often use linguistic terms to communicate each other and to describe their identification and judgment for a situation in attempting to reach an optimal solution. In a sequence processing, relevant SA information is passed on to the next person that may produce fewer uncertain hypotheses. Parallel processing would make team members develop different situation models that at the end might lead team members to talk about different conceptions of the ambiguous situation.

(3) Distributed environment: Perceiving, recognising, and understanding activities of other members are basic requirements for a collaborative

team work and more generally for members' communication and interaction. When team members collaborate in a face-to-face environment, they can easily share information obtained for a situation in a group session. Although each individual may have personal prior knowledge, experience, and opinions, the shared physical environment provides a common reference to support the communication among team members and develop an information sharing working environment. When individuals collaborate across distances, each individual's SA, including the awareness of the local and remote situations, would be facilitated and supported by technologies (Sonnenwald *et al.*, 2004). The Web-based technology is an approach to deal with the issue. The Web-based FGDSS presented in Chapter 11 provides a way to support such group's awareness for a situation through online information sharing, interaction, and assessment.

The three issues identified above generate a crucial requirement for team SA with uncertain information processing technique in a distributed environment support by related software systems.

15.3 A Case-Based Example

The following example is a demonstration on using the Web-based FGDSS to support reaching a team SA.

Addressed to national health authorities, WHO has set out a series of guidance for the global surveillance and reporting of SARS as an ongoing strategy for rapidly detecting cases and preventing further national or international spread. WHO guidelines aim at the early detection and investigation of individuals with clinically apparent SARS-associated infection. The late revised guidelines draw on experiences during four incidents in which cases of SARS occurred following breaches in laboratory bio-safety, or human exposure to an animal reservoir or other environmental source. Apart from demonstrating the importance of continued vigilance, these incidents revealed the need for more precise guidance on laboratory testing and on the requirements for official reporting to WHO. According to the guidelines, it is necessary to identify and be aware of the risk levels of SARS epidemic in a region based on various evidences and criteria such as if SARS is circulating in a big human populations, the detection of human chains of transmission, or the evidence of international spread.

Suppose a survaillance team is collaboratively observing the SARS outbreak and epidemic for a certain region. The team consists of five members: *Officer 1*, *Officer 2*, *Officer 3*, *Officer 4*, and *Officer 5*. Here *Officer 1* is the Chief Observer (who takes intellectual responsibility for the survaillance and report) and other four members are the Partner Observers. The five members come from different organisations and play different roles in the group. After receiving and studying reports from different resources, each member judges the current situation of the region's SARS epidemic. These individuals' awareness for the situation will be combined into a consensus SA in the team for further activity recommendations. During the information sharing process, each member has their own understanding to the current situation, and has different opinions about which risk level of SARS epidemic is in that region. At this stage, a consensus SA should be reached in order to issue a suitable SARS alert.

As the data shown in these reports has uncertainty, inconsistence, and incompleteness, it is hard to directly use the obtained data to determine the risk level of SARS epidemic for the region. The group defines four risk levels: Level 1 (low risk epidemic), Level 2 (middle risk epidemic), Level 3 (high risk epidemic), and Level 4 (very high risk epidemic). These group members need to have consensus awareness on the level of current SARS epidemic in the region through the meeting so that to determine the degree of SARS alert. The developed Web-based FGDSS can support, in some degree, reaching a consensus team SA on the risk level of SARS epidemic, which is described as follows.

Step 1: *Officer 1* logins to the system first and sets up a discussion group. A collaboration and information sharing environment is formed.

Step 2: All members express their opinions about the current SA of that region's SARS epidemic. Both *Officer 1* and *Officer 2* believe that the current situation of SARS epidemic in the region is with *Risk level* 1, *Officer 3* and *Officer 4*'s are *Risk level 2*, and the *Officer 5*'s is *Risk level 3*. To support further discussion and get a consensus SA, the three kinds of opinions as alternatives for the team SA problem are shown in Fig. 15.1.

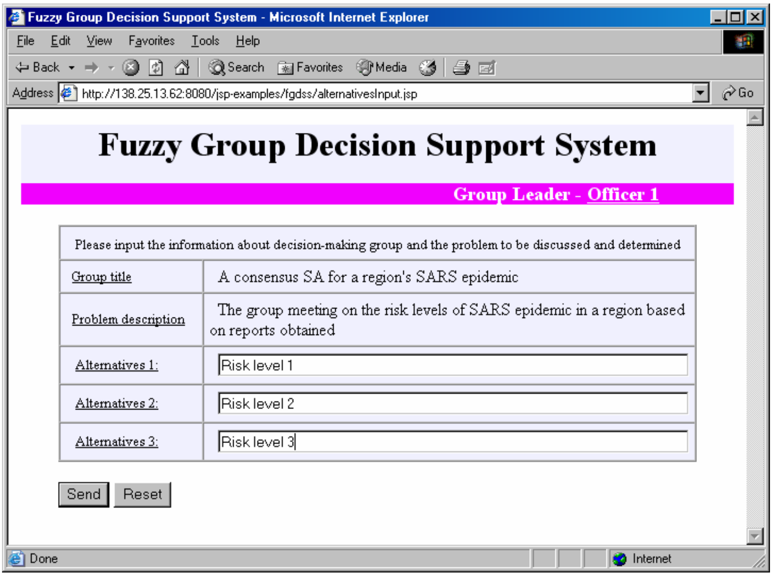


Fig. 15.1: Three SA alternatives proposed by all members

Step 3: Each group member proposes a few criteria for ranking and assessing these SA alternatives. *Officer 1* proposes two criteria that are '*The atypical presentations*' and '*The epidemic time*' as shown in Fig. 15.2; other officers propose criteria including '*The clinical symptoms and signs*,' '*The number of infected patients*,' '*The number of death*,' and '*The epidemic time*.' All these criteria are put into a criterion pool and finally four of them are chosen as assessment-criteria in the group, which are as shown in Fig. 15.3. The information sharing here is very important. It fully uses all members' knowledge and experiences to show their opinions and communicate with each other.

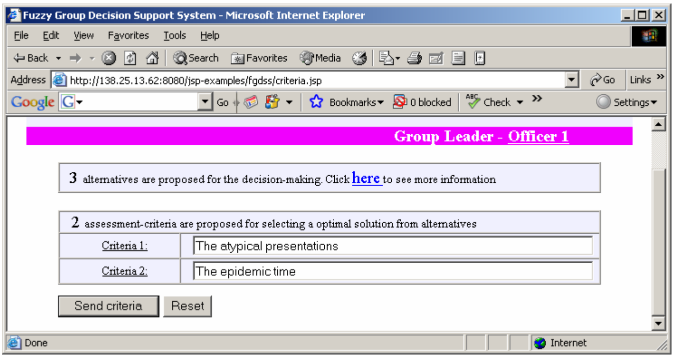


Fig. 15.2: The criteria proposed by *Officer 1*

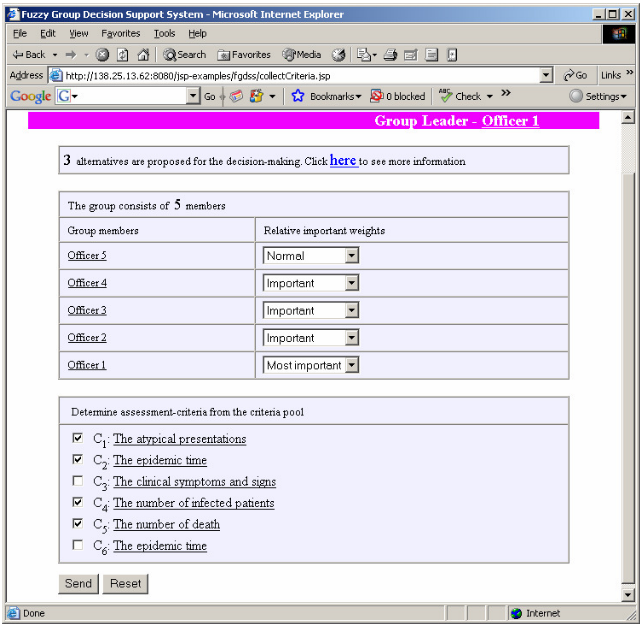


Fig. 15.3: Four assessment-criteria and group members' weights

Step 4: As group members have different experience and play different roles in the group, each member is assigned with a weight that is described by a linguistic term. Here, *Officer 1* is assigned as '*Most important*,' *Officer 2*, *Officer* 3, and *Officer 4* are assigned as '*Important*,' respectively, and *Officer 5* as '*Normal*,' which are shown in Fig. 15.3.

Step 5: Based on the criteria proposed, each group member fills a pairwise comparison matrix of the relative importance of these criteria and a belief level matrix to express their opinion about the current SA under the four assessment-criteria. Suppose *Officer 1* fills the two matrices as in Fig. 15.4. In the criteria comparison matrix, the criterion '*The atypical presentations*' is thought as '*more important*' than the criterion '*The epidemic time*;' also the criterion '*The number of infected patients*' is '*less important*' than the criterion '*The number of death*,' *etc*. Also in the preference belief level matrix, comparing with other alternatives under the criteria '*The atypical presentations*,' the preference belief level of *Risk level 1* for the current SA epidemic is regarded as '*very high*,' *Risk level 2* as '*high,*' *Risk level 3* as '*medium,*' *etc*. Obviously, group members' preferences are fully expressed here.

Step 6: Based on the normalised weights of all group members proposed in Step 4, and the criteria comparison matrices and the belief level matrices generated by all members in Step 5, all opinions of the members are aggregated. The final ranking result as shown in Fig. 15.5.

Based on the result generated, the consensus SA in the team is that the current situation of SARS epidemic in that region is about on *Risk level 2*.

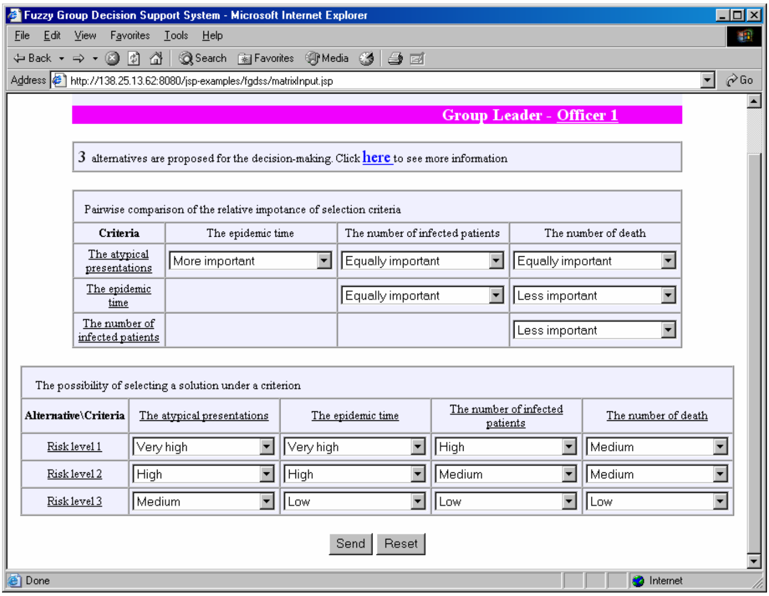


Fig. 15.4: Criteria comparison matrix and belief level matrix filled by *Officer 1*

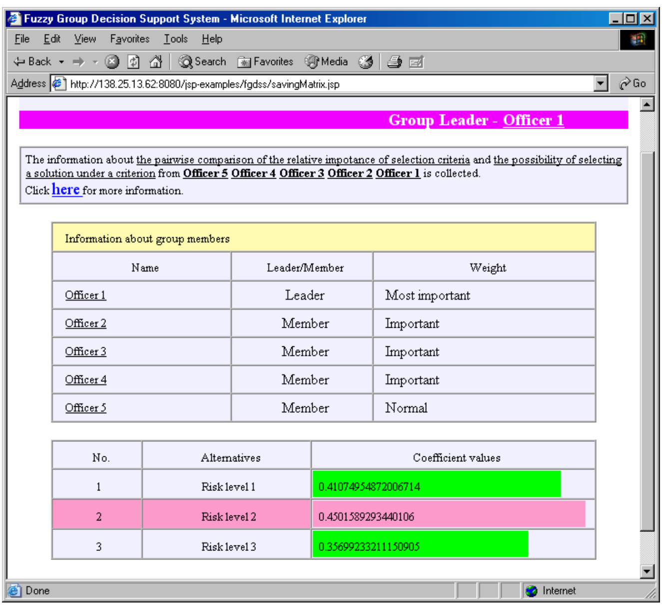


Fig. 15.5: The final result for the consensus SA in the research group

15.4 Summary

Generally, SA is the continuous extraction, integration and use of environmental knowledge by a single person. When the concept of SA is extended to teams, the team SA will have the meaning of understanding of the activities of the others, which may affect the whole team's goals and/or procedures. Various inconsistence and uncertainties are involved in situation information and its processing. When individual team members collaborate across distances, they have not only different knowledge and abilities, but also different physical environments. Each individual SA would be facilitated and supported by information technologies. The use of the Web-based FGDSS is an optional approach for this purpose.

Chapter 16

Reverse Logistics Management

Reverse logistics has gained increasing importance as a profitable and sustainable business strategy. As a reverse logistics chain has strong internal and external linkages, the management of a reverse logistics chain becomes an area of organisational competitive advantage, in particular, with the growth of e-business applications. To be effectively managed a reverse logistics chain always involves a decision optimisation issue in which uncertain information, individual situation, multiple criteria, and dynamic environment all need to be considered.

In this chapter, we address the need of supporting reverse logistics managers in selecting an optimal alternative for goods return. After briefly introducing reverse logistics, we first analyse the main operational functions in a reverse logistics chain and the characteristics of decision making in selecting the best way to handle reverse logistics. We then establish a multi-stage multi-criteria decision support model for the reverse logistics management. Finally, we use the hybrid FMCDM method presented in Chapter 9 to illustrate how to support goods return decision making in the reverse logistics.

16.1 Reverse Logistics Chain

As companies are increasing their levels of outsourcing, buying goods or services, they are spending increasing amounts on supply related activities. Logistics, the key of supply chain management, has become a hot competitive advantage.

There are two logistics channels in a supply chain system of a company. Forward logistics channel concerns the movement of goods from source to the point of consumption. A backward movement can be happened to return goods to suppliers called *reverse logistics*. Forward logistics usually brings profit to all operational departments involved, while reverse logistics usually cannot. However, the high rate of goods return from online purchases, the increasing environmental regulations and standards, and the growing consumer awareness of recycling have brought a need to rethink the significance of reserve logistics. Some reports have shown that companies trying to hide from the significance of reverse logistics miss tremendous profit making opportunities. The reason is that companies can use reverse logistics as an opportunity for maintaining customer support, building good customer relationship, and reaching the ultimate business aspect of profitability. Moreover, many companies have discovered that effective management for a reverse logistics chain such as the reductions in inventory carrying costs, transportation costs, and waste disposal costs can be also substantial with the supply chain program.

Products may become obsolete, damaged, or non-functioning and need to be returned to their source points for replacing, repairing, or disposition. This procedure forms a reverse logistics chain. The reverse logistics is therefore defined as the process of planning, implementing, and controlling flows of raw materials, in process inventory, and finished goods, from a manufacturing, distribution or use point to a point of recovery or point of proper disposal.

A reverse logistics chain involves a series of stages, each concerns a kind of activities associated with the management of goods (can be products, materials or components) return, with different facilities. These stages/facilities are interrelated in a way that a decision made at previous stage affects the decision making in the following stages. In general, the stages of a reverse logistics chain typically include collection, combined testing/sorting/inspection/separation process, reprocessing/repairing or direct recovery, and redistribution/resale/reusing or disposal which can be also happened with other operational functions such as testing. As shown in Fig. 16.1, *Supply*, *Manufacture*, *Distribution*, and *Consumer* form a flow of forward logistics. A reverse logistics flow has a backward movement from *Consume* to *Supply*. The stage *Collection* refers to all activities rendering goods to be returned available and physically moving them to some point where a further treatment is taken care of stage. *Testing* (or inspection) determines whether collected goods are in fact reusable or how much work needs to be paid in order to make it usable. *Sorting* (or separation) decides what to do with each or a set of collected goods, including reprocessing and disposal. Thus, *testing* and *sorting* will result in splitting the flow of collected goods according to distinct treatment options. *Reprocessing* means the actual transformation of returned goods into usable products again. The transformation may take different forms including *recycling*, *reconditioning*, and *remanufacturing*. Disposal could be an option at this stage as well. *Redistribution* refers to directing reusable products to a potential reuse market and to physically moving them to future end customers. Therefore, the reverse logistics can simply be just reselling a product, or can be accompanies by a series of processes, as shown in Fig. 16.1, from *collection* to *reuse* or *disposal*.

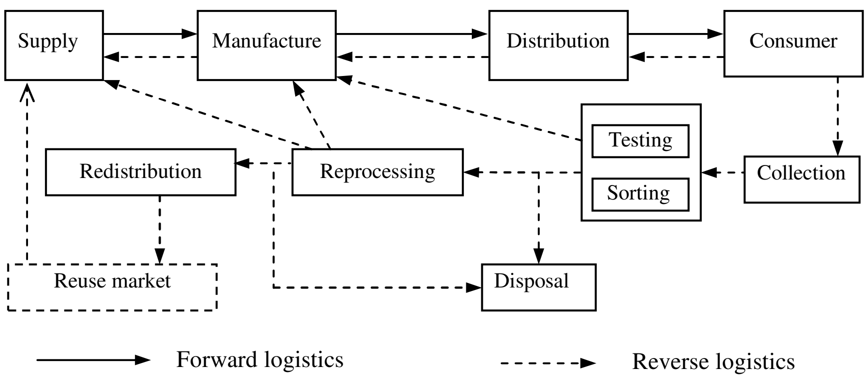


Fig. 16.1: Forward logistics chain and reverse logistics chain

16.2 Characteristics of Decision Making in the Reverse Logistics

There are several kinds of actors involved in reverse logistics activities in practice. They are independent intermediaries, specific recovery companies, reverse logistics service providers, municipalities taking care of waste collection, and public-private foundations created to take of goods recovery. The aims of different kinds of actors in a reverse logistics chain are different. For example, a *manufacture* may do recycling in order to prevent jobbers reselling its products at a lower price, while a collector may collect used products in order to establish a long-term customer relationship. These actors can also be logically differentiated into *returners*, *receivers*, *collectors*, *processors*, and *sales persons* based on the features of their roles in a reverse logistics chain. The most important type of actors is *returner* as any stage can be a returner, including customers, in the whole reverse logistics chain, hence suppliers, manufactures, wholesalers, and retailers.

Returners always need to decide how to best move current returned goods, such as to return it to a factory for repairing or disposal it locally. Returners at different stages or at the same stage but with different goods returns may have different alternatives and different selection criteria to find the best way. For example, at the stage of *collection*, the decision is mainly about planning and scheduling of recovery operations, and the transportation and the warehousing of returns have to be dealt with. At the stage of *sorting*, returners need to determine whether or not to do recovery and which type of recovery if do. The decisions for a goods return at a previous stage will become constraints given for and impact directly on the decision activities of its following stages. For example, when one product is identified to be not usable any other decisions on storage, treatment, transportation for reusing process are not considerable except transportation for disposing processed wastes. Therefore, every decision has to bear the impact on the decisions at its previous stages.

The following characteristics have been seen through the above analysis:

(1) Reverse logistics management involves decision making at multiple stages. All the stages involved in the chain are interrelated in a way that a decision made at one stage affects the performance of next stages.

(2) Decisions made at different stages are based on different alternatives and selection criteria. For example, the alternatives to deal with a goods return in a collection stage are totally different from one in a redistribute stage.

(3) At each stage, returners' business aspects, related alternatives, and evaluation criteria are dynamic changed. The change is caused by both the features of returned goods and the actions of previous functions of the reverse logistics chain. The analysis reminds a multi-stage multi-criteria decision support model to help the selection of the best way to handle a goods return in a reverse logistics chain.

(4) The importance degrees of these operational functions are different in a goods return. Some functions may play more important roles than others for a particular goods return. The degree of importance of each operational function is also variable for different goods returns. This variance is mainly dependent on the business aspect of the reverse logistics management. For example, if the company's reverse logistics management is to provide customer services in warranties, then the function of *collection* may play a more important role in the reverse logistics than the reprocessing for the disassembly of products. If the business aspect is more environmentally related such as '*reclaiming parts*,' the function of *sorting* may be more important.

From Fig. 16.1, returners can be classified into four basic types: collector, tester/sorter, processor, and redistributor, as shown in Table 16.1. The four types of returners are at four main functional stages of a reverse logistics chain respectively. For each type of returners, possible business aspects are shown in the column two of Table 16.1. Once a returner's business aspects for a particular goods return are determined, a set of alternatives can be identified. For example, two business aspects of a collector are to maximise customer relationship and to minimise customer services cost. Related alternatives are thus recycling, reconditioning, and disposal as shown in the column three of Table 16.1.

Table 16.1: Example of relationships among returners' types, business aspects, and alternatives in a reverse logistics chain

Returner types	*Business aspects (F)*	*Alternatives (A)*
Collector	Maximising customer relationship Minimising customer service cost in warranties	Replacement Local storage Customer post
Tester/Sorter	Minimising total operational cost Maximising customer relationship Maximising satisfying environmental regulation	Recycling Remanufacturing Reuse Disposal
Processor	Minimising total operational cost Maximising customer services in warranties of repair	Local remanufacturing Recycling Disposal
Redistributor	Maximising business profit Maximising reclaiming parts Minimising time	Resale Disposal Storage

To evaluate these alternatives, each business aspect can be extended to a number of criteria, which are strongly dependent on the corresponded business aspects. If necessary, each criterion can be further described by a number of items (sub-criteria). For example, when a company's business aspect for a goods return is to minimise customer services, *time* is one criterion, and its related items include *collect time*, *treatment time*, and *transportation time*, which are the assessment items for the selection of a solution from related alternatives. Table 16.2 lists the possible business aspects (F), related selection criteria (C), and involved assessment items (I) as an example.

Table 16.2: Example of relationships among business aspects, selection criteria, and related items in a reverse logistics chain

Business aspects (F)	*Selection criteria (C)*	*Related items (sub-criteria) (I)*
Minimising total operational cost	Cost	Collection cost Storage cost Treatment cost Transportation cost for reusing processed wastes Transportation cost for disposing processed wastes Repair cost
Minimising customer services in warranties	Time	Collecting time Treatment time Transportation time
Maximising customer relationship	Customer satisfaction	Product life stages (Introduction, Growth, Maturity, and Decline) Time Usability
Maximising business profit	Benefit/cost	Reusability Resale income Repair cost Transportation cost Redistribute cost

16.3 A Multi-Stage Multi-Criteria Decision Support Model

We propose a multi-stage multi-criteria decision support model for reverse logistics management as shown in Fig. 16.2. This model describes a whole decision-making process of a returner at any stage of a reverse logistics chain. In the model, when a returner's type is known, its business aspects can be identified based on the relationships shown in Table 16.1. After business aspects are determined, the returner is allowed to indicate a weight for each aspect based on individual experience and knowledge. Related alternatives are then determined based on the relationships shown in Table 16.1 as well.

As the alternatives of a goods return decision are totally related to its business aspects, when an aspect's weight is very low, its related alternatives and selection criteria won't be considered. To evaluate these alternatives, a set of selection criteria is determined based on information shown in Table 16.2. The types of returners and their preferences for business aspects may result in different sets of alternatives. Obviously, this decision process involves multiple layers of relationships: from the type of a returner to determining its business aspects, then alternatives, and finally selection criteria (and/or sub-criteria).

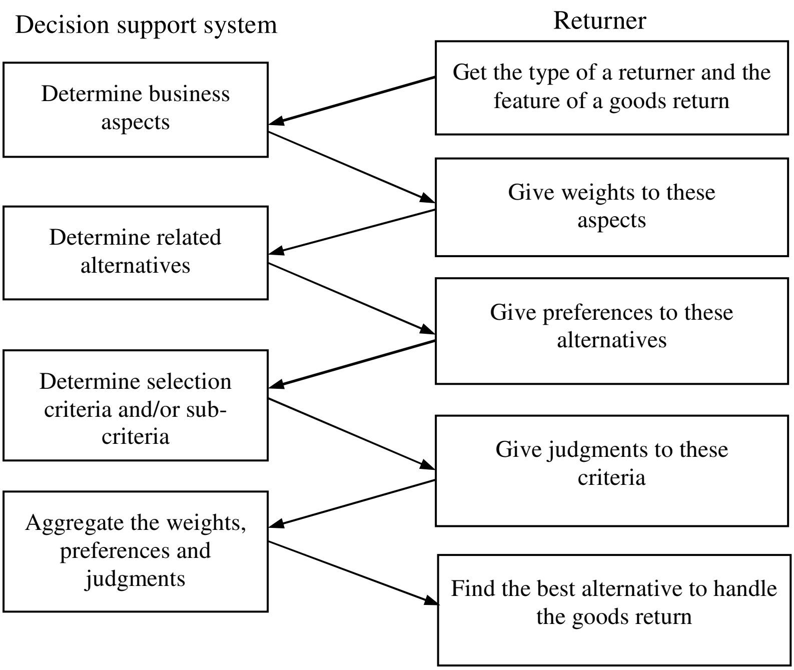


Fig. 16.2: A multi-stage multi-criteria decision support model of reverse logistics management

In practice, reverse logistics managers (returners) often imprecisely know the values of related constraints and evaluation criteria in selecting an optimal alternative. They often describe and measure the degree of weights and their preferences in linguistic terms, such as '*important*,' '*high*,' or '*low*' since a numerical evaluation is sometimes unacceptable. Each criterion may involve a number of related selection items (sub-criteria), estimation of these items' values is needed and these estimated values are often with imprecision. For example, when minimising the total operational cost is the business aspect of a goods return at an operational stage, five major time-related cost items may need to be estimated and measured: collection cost, storage cost, treatment cost, transportation cost for reusing processed wastes, and transportation cost for disposing processed wastes. All these estimations and measures often involve imprecise values.

The uncertainty and imprecision features will affect on the processing of a decision evaluation. When several layers of a goods return decision evaluation are synthesised into an aggregated result, that is, the weights of business aspects will be combined with the preferences of related criteria to selection alternatives, the uncertainty and imprecision features will be integrated into the final outcome, an optimal plan, for the particular goods to be returned.

Now we will apply the hybrid FMCDM method, presented in Section 9.4, in this decision problem. The method has been implemented into a DSS called *FMCDSS*, which can effectively handle multi-stage, multi-criteria decision making with uncertainty in the reverse logistics management.

16.4 A Case Study

A returner at the *collection* stage of a reverse logistics chain needs to make a decision for a particular goods return. The returner has currently two main business aspects to concern:

$F = (F_1, F_2)$ = {*minimise service cost, maximise customer relationship*},

and three alternatives

$A = (A_1, A_2, A_3)$ = {*replacement, taking to local store for testing (test), asking customer to post it to the collector (post)*}

for the goods return. The first aspect can be evaluated by three criteria

(C_{11}, C_{12}, C_{13}) = {*collection cost, storage/testing cost, new product cost*},

and the second one can be evaluated by two criteria

(C_{21}, C_{22}) = {*time, convenience*}.

The relationships among these business aspects, alternatives, and evaluation criteria are shown in Fig. 16.3. The aim of the decision is to get a solution from the alternatives that can maximally reach the goals of these business aspects will be selected.

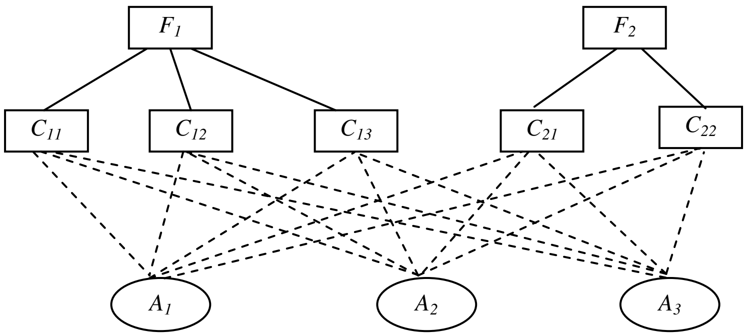


Fig. 16.3: An example of the interrelation among aspects, criteria, and alternatives

The logistics manager (returner) needs to give his/her preference and evaluation for the three ways of goods return.

Table 16.3: The relationships among the elements in logistics

				A_1	A_2	A_3
F_1	WF_1	C_{11}	WC_{11}	AC_{11}^1	AC_{11}^2	AC_{11}^3
		C_{12}	WC_{12}	AC_{12}^1	AC_{12}^2	AC_{12}^3
		C_{13}	WC_{13}	AC_{13}^1	AC_{13}^2	AC_{13}^3
F_2	WF_2	C_{21}	WC_{21}	AC_{21}^1	AC_{21}^2	AC_{21}^3
		C_{22}	WC_{22}	AC_{22}^1	AC_{22}^2	AC_{22}^3

In Table 16.3, WF_i and WC_{ij} provided by returners are the weights and can be linguistic terms, which are described by fuzzy numbers as shown in Table 16.4. The linguistic terms about AC_{ij}^k are the evaluation values described by fuzzy numbers, as shown in Tables 16.5.

Table 16.4: An example of linguistic terms for WF_i and WC_{ij} weights and related fuzzy numbers

Linguistic terms	Fuzzy numbers
Absolutely unimportant	$\bigcup_{\lambda\in[0,1]} \lambda[0, \frac{\sqrt{1-\lambda}}{10}]$
Unimportant	$\bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{\lambda}}{10}, \frac{\sqrt{9-8\lambda}}{10}]$
Less important	$\bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{8\lambda+1}}{10}, \frac{\sqrt{25-16\lambda}}{10}]$
Important	$\bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{16\lambda+9}}{10}, \frac{\sqrt{49-24\lambda}}{10}]$
More important	$\bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{24\lambda+25}}{10}, \frac{\sqrt{81-32\lambda}}{10}]$
Strongly important	$\bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{32\lambda+49}}{10}, \frac{\sqrt{100-19\lambda}}{10}]$
Absolutely important	$\bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{19\lambda+81}}{10}, 1]$

Table 16.5: An example of linguistic terms for AC_{ij}^k and related fuzzy numbers

Linguistic terms	Fuzzy numbers
Lowest	$\bigcup_{\lambda\in[0,1]} \lambda[0, \frac{\sqrt{1-\lambda}}{10}]$
Very low	$\bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{\lambda}}{10}, \frac{\sqrt{9-8\lambda}}{10}]$
Low	$\bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{8\lambda+1}}{10}, \frac{\sqrt{25-16\lambda}}{10}]$
Medium	$\bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{16\lambda+9}}{10}, \frac{\sqrt{49-24\lambda}}{10}]$
High	$\bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{24\lambda+25}}{10}, \frac{\sqrt{81-32\lambda}}{10}]$
Very high	$\bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{32\lambda+49}}{10}, \frac{\sqrt{100-19\lambda}}{10}]$
Highest	$\bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{19\lambda+81}}{10}, 1]$

Based on the hybrid FMCDM method, the details of the proposed approach for the goods return case study are described as follows.

Step 1: A returner gives weights to two business aspects: F_1 (*service cost*) and F_2 (*customer relationship*), weights of C_{11} (*collection cost*), C_{12} (*storage, testing cost*), and C_{13} (*new product cost*) for F_1, and weights of C_{21} (*time*), C_{22} (*convenience*) for F_2, respectively:

$WF = (WF_1, WF_2) = \{Unimportant, Strongly\ important\}$

$WC_1 = (WC_{11}, WC_{12}, WC_{13}) = \{Unimportant, Unimportant, Strongly\ important\}$

$WC_2 = (WC_{21}, WC_{22}) = \{Strongly\ important, Unimportant\}$

Step 2: The two aspects and their criteria are checked and finalised through applying related rules presented in the hybrid FMCDM method. Fig. 16.4 displays the finalised weights of these business aspects and criteria in the FMCDSS.

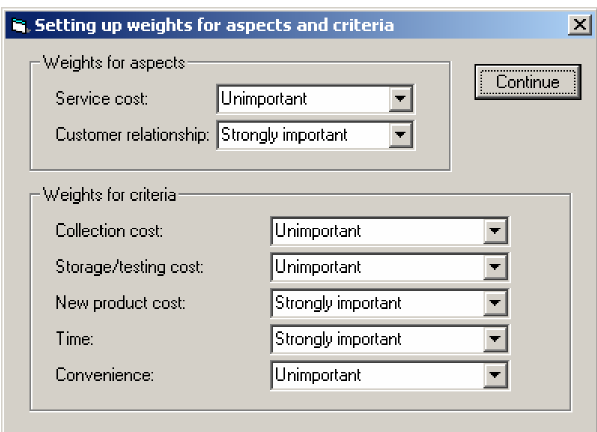


Fig. 16.4: Weights of the business aspects and criteria

Step 3: The returner provides relevant degrees (evaluation value) of A_k on C_{ij} (k=1, 2, 3) (see Fig. 16.5).

$AC_1^1 = \{AC_{11}^1, AC_{12}^1, AC_{13}^1\} = \{High, Very\ low, Very\ high\}$

$AC_2^1 = \{AC_{21}^1, AC_{22}^1\} = \{Very\ low, Very\ high\}$

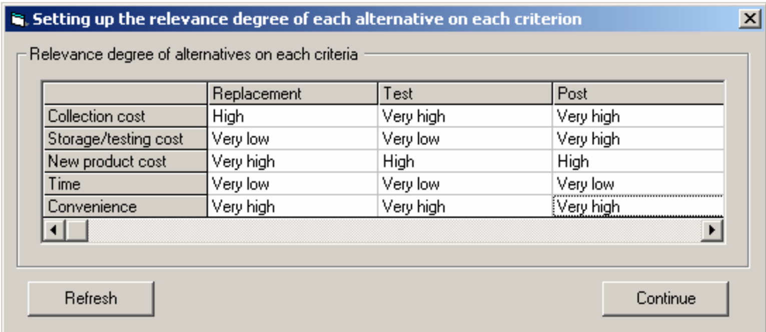
$AC_1^2 = \{AC_{11}^2, AC_{12}^2, AC_{13}^2\} = \{Very\ high, Very\ low, High\}$

$AC_2^2 = \{AC_{21}^2, AC_{22}^2\} = \{Very\ low, Very\ high\}$

$AC_1^3 = \{AC_{11}^3, AC_{12}^3, AC_{13}^3\} = \{Very\ high, Very\ high, High\}$

$AC_2^3 = \{AC_{21}^3, AC_{22}^3\} = \{Very\ low, Very\ high\}$

For example, the manager thinks '*replacement* (A_1)' has a very high '*new product cost* (C_{13})', therefore gives a value '*very high*' on it.



	Replacement	Test	Post
Collection cost	High	Very high	Very high
Storage/testing cost	Very low	Very low	Very high
New product cost	Very high	High	High
Time	Very low	Very low	Very low
Convenience	Very high	Very high	Very high

Fig. 16.5: Relevance degree of each criterion on each alternative

Step 4: The weights proposed in Step 1 are normalised.

Since $\sum_{j=1}^{3} WC_{1j_0}^{R} = 0.3+0.3+1=1.6$, $\sum_{j=1}^{2} WC_{2j_0}^{R} = 1+0.3=1.3$ we have

$$WC_{11}^{*} = WC_{12}^{*} = \bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{\lambda}}{16}, \frac{\sqrt{9-8\lambda}}{16}],$$

$$WC_{13}^{*} = \bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{32\lambda+49}}{16}, \frac{\sqrt{100-19\lambda}}{16}],$$

$$WC_{21}^{*} = \bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{\lambda}}{13}, \frac{\sqrt{9-8\lambda}}{13}],$$

$$WC_{22}^{*} = \bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{32\lambda+49}}{13}, \frac{\sqrt{100-19\lambda}}{13}].$$

Step 5: Calculating the relevance degree FA_i^k of alternatives A_k on F_i, i = 1, 2, and k = 1, 2, 3, we have

$$FA_1^1 = \bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{\lambda(24\lambda+25)}}{160} + \frac{\lambda}{160}, \frac{\sqrt{(9-8\lambda)(81-32\lambda)}}{160} + \frac{109-27\lambda}{160}]$$

$$FA_2^1 = WC_2^{*} \times AC_2^1 = \sum_{j=1}^{2} WC_{2j}^{*} \times AC_{2j}^1$$

$$= \bigcup_{\lambda\in[0,1]} \lambda[\frac{2\sqrt{\lambda(32\lambda+49)}}{130}, \frac{2\sqrt{(9-8\lambda)(100-19\lambda)}}{130}]$$

$$FA_1^2 = \bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{\lambda(32\lambda+49)}}{160} + \frac{32\lambda+49}{160} + \frac{\sqrt{(32\lambda+49)(24\lambda+25)}}{160},$$

$$\frac{\sqrt{(9-8\lambda)(100-19\lambda)}}{160}+\frac{9-8\lambda}{160}+\frac{\sqrt{(100-19\lambda)(81-32\lambda)}}{160}]$$

$$FA_2^2=\bigcup_{\lambda\in[0,1]}\lambda[\frac{2\sqrt{\lambda(32\lambda+49)}}{130},\frac{2\sqrt{(9-8\lambda)(100-19\lambda)}}{130}]$$

$$FA_1^3=\bigcup_{\lambda\in[0,1]}\lambda[\frac{2\sqrt{\lambda(32\lambda+49)}}{160}+\frac{\sqrt{(32\lambda+49)(24\lambda+25)}}{160},$$

$$\frac{2\sqrt{(9-8\lambda)(100-19\lambda)}}{160}+\frac{\sqrt{(100-19\lambda)(81-32\lambda)}}{160}]$$

$$FA_2^3=\bigcup_{\lambda\in[0,1]}\lambda[\frac{2\sqrt{\lambda(32\lambda+49)}}{130},\frac{2\sqrt{(9-8\lambda)(100-19\lambda)}}{130}]$$

Step 6: Normalising the relevance degree FA_i^k of the alternatives A_k on F_i, i = 1, 2, and k = 1, 2, 3.

$$\overline{FA}_1^1=\bigcup_{\lambda\in[0,1]}\lambda[\frac{\sqrt{\lambda(24\lambda+25)}}{160\times 1.3115}+\frac{33\lambda+49}{160\times 1.3115},\frac{\sqrt{(9-8\lambda)(81-32\lambda)}}{160\times 1.3115}+\frac{109-27\lambda}{160\times 1.3115}]$$

$$\overline{FA}_2^1=\bigcup_{\lambda\in[0,1]}\lambda[\frac{2\sqrt{\lambda(32\lambda+49)}}{130\times 1.3115},\frac{2\sqrt{(9-8\lambda)(100-19\lambda)}}{130\times 1.3115}]$$

$$\overline{FA}_1^2=\bigcup_{\lambda\in[0,1]}\lambda[\frac{\sqrt{\lambda(32\lambda+49)}}{160\times 1.2678}+\frac{\lambda}{160\times 1.2678}+\frac{\sqrt{(32\lambda+49)(24\lambda+25)}}{160\times 1.2678},$$

$$\frac{\sqrt{(9-8\lambda)(100-19\lambda)}}{160\times 1.2678}+\frac{9-8\lambda}{160\times 1.2678}+\frac{\sqrt{(100-19\lambda)(81-32\lambda)}}{160\times 1.2678}]$$

$$\overline{FA}_2^2=\bigcup_{\lambda\in[0,1]}\lambda[\frac{2\sqrt{\lambda(32\lambda+49)}}{130\times 1.2678},\frac{2\sqrt{(9-8\lambda)(100-19\lambda)}}{130\times 1.2678}]$$

$$\overline{FA}_1^3=\bigcup_{\lambda\in[0,1]}\lambda[\frac{2\sqrt{\lambda(32\lambda+49)}}{160\times 1.3990}+\frac{\sqrt{(32\lambda+49)(24\lambda+25)}}{160\times 1.3990},$$

$$\frac{2\sqrt{(9-8\lambda)(100-19\lambda)}}{160\times 1.3990}+\frac{\sqrt{(100-19\lambda)(81-32\lambda)}}{160\times 1.3990}]$$

$$\overline{FA}_2^3=\bigcup_{\lambda\in[0,1]}\lambda[\frac{2\sqrt{\lambda(32\lambda+49)}}{130\times 1.3990},\frac{2\sqrt{(9-8\lambda)(100-19\lambda)}}{130\times 1.3990}]$$

Step 7: Calculating the relevance degree S_k of the alternatives A_k on F_i by using $S_k=\overline{FA}^k\times WF=\sum_{i=1}^{2}\overline{FA}_i^k\times WF_i$, k = 1, 2, 3.

$$S_1 = \bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{\lambda}}{10}\left(\frac{\sqrt{\lambda(24\lambda+25)}}{160\times1.3115}+\frac{33\lambda+49}{160\times1.3115}\right)+\frac{\sqrt{32\lambda+49}}{10}\times\frac{2\sqrt{\lambda(32\lambda+49)}}{130\times1.3115},$$
$$\frac{\sqrt{9-8\lambda}}{10}\left(\frac{\sqrt{(9-8\lambda)(81-32\lambda)}}{160\times1.3115}+\frac{109-27\lambda}{160\times1.3115}\right)+\frac{\sqrt{100-19\lambda}}{10}\times\frac{2\sqrt{(9-8\lambda)(100-19\lambda)}}{130\times1.3115}]$$

$$S_2 = \bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{\lambda}}{10}\left(\frac{\sqrt{\lambda(32\lambda+49)}}{160\times1.2678}+\frac{\lambda}{160\times1.2678}+\frac{\sqrt{(32\lambda+49)(24\lambda+25)}}{160\times1.2678}\right)+\frac{\sqrt{32\lambda+49}}{10}\times\frac{2\sqrt{\lambda(32\lambda+49)}}{130\times1.2678},$$
$$\frac{\sqrt{100-19\lambda}}{10}\times\frac{2\sqrt{(9-8\lambda)(100-19\lambda)}}{130\times1.2678}+\frac{\sqrt{9-8\lambda}}{10}\times$$
$$\left(\frac{\sqrt{(9-8\lambda)(100-19\lambda)}}{160\times1.2678}+\frac{9-8\lambda}{160\times1.2678}+\frac{\sqrt{(100-19\lambda)(81-32\lambda)}}{160\times1.2678}\right)]$$

$$S_3 = \bigcup_{\lambda\in[0,1]} \lambda[\frac{\sqrt{\lambda}}{10}\left(\frac{2\sqrt{\lambda(32\lambda+49)}}{160\times1.3990}+\frac{\sqrt{(32\lambda+49)(24\lambda+25)}}{160\times1.3990}\right)+\frac{\sqrt{32\lambda+49}}{10}\times\frac{2\sqrt{\lambda(32\lambda+49)}}{130\times1.3990},$$
$$\frac{\sqrt{100-19\lambda}}{10}\times\frac{2\sqrt{(9-8\lambda)(100-19\lambda)}}{130\times1.3990}+\frac{\sqrt{9-8\lambda}}{10}\times$$
$$\left(\frac{2\sqrt{(9-8\lambda)(100-19\lambda)}}{160\times1.3990}+\frac{\sqrt{(100-19\lambda)(81-32\lambda)}}{160\times1.3990}\right)]$$

Step 8: The results S_k, $k=1,2,3$ are normalised to be positive fuzzy numbers, and their ranges belong to the closed interval [0, 1]. Both positive distance and negative distance are then calculated respectively by

$$d_1^* = d(S_1, S^*)$$
$$=\left(\int_0^1 \frac{1}{2}\left[\left(\frac{\sqrt{\lambda}}{10}\left(\frac{\sqrt{\lambda(24\lambda+25)}}{160\times1.3115}+\frac{33\lambda+49}{160\times1.3115}\right)+\frac{\sqrt{32\lambda+49}}{10}\times\frac{2\sqrt{\lambda(32\lambda+49)}}{130\times1.3115}-1\right)^2\right.\right.$$
$$\left.\left.+\left(\frac{\sqrt{9-8\lambda}}{10}\left(\frac{\sqrt{(9-8\lambda)(81-32\lambda)}}{160\times1.3115}+\frac{109-27\lambda}{160\times1.3115}\right)+\frac{\sqrt{100-19\lambda}}{10}\times\frac{2\sqrt{(9-8\lambda)(100-19\lambda)}}{130\times1.3115}-1\right)^2\right]d\lambda\right)^{\frac{1}{2}}$$
$$=0.80143$$
$$d_2^* = d(S_2, S^*) = 0.80233$$
$$d_3^* = d(S_3, S^*) = 0.81200$$
$$d_1^- = d(S_1, S^-)$$
$$=\left(\int_0^1 \frac{1}{2}\left[\left(\frac{\sqrt{\lambda}}{10}\left(\frac{\sqrt{\lambda(24\lambda+25)}}{160\times1.3115}+\frac{33\lambda+49}{160\times1.3115}\right)+\frac{\sqrt{32\lambda+49}}{10}\times\frac{2\sqrt{\lambda(32\lambda+49)}}{130\times1.3115}-0\right)^2\right.\right.$$

$$+\left(\frac{\sqrt{9-8\lambda}}{10}\left(\frac{\sqrt{(9-8\lambda)(81-32\lambda)}}{160\times1.3115}+\frac{109-27\lambda}{160\times1.3115}\right)+\frac{\sqrt{100-19\lambda}}{10}\times\frac{2\sqrt{(9-8\lambda)(100-19\lambda)}}{130\times1.3115}-0\right)^2\Bigg]d\lambda\Bigg)^{\frac{1}{2}}$$

$= 0.26982$

$d_2^- = d(S_2, S-) = 0.27111$

$d_3^- = d(S_3, S-) = 0.25811$

Step 9: After d_k^* and d_k^- of each alternative A_k (k = 1, 2, 3) are obtained, the closeness coefficient of each alternative is calculated as:

$$D_1 = \frac{1}{2}\left(d_1^- + (1-d_1^*)\right) = \frac{1}{2}\left(0.26982 + (1-0.80143)\right) = 0.23420$$

$$D_2 = \frac{1}{2}\left(d_2^- + (1-d_2^*)\right) = \frac{1}{2}\left(0.27111 + (1-0.80233)\right) = 0.23439$$

$$D_3 = \frac{1}{2}\left(d_3^- + (1-d_3^*)\right) = \frac{1}{2}\left(0.25811 + (1-0.81200)\right) = 0.22306$$

We have

$$\max\{D_1, D_2, D_3\} = D_2 = 0.23439.$$

As D_2 has the highest closeness coefficient value (also see Fig. 16.6), the alternative A_2, that is, '*to take it to local storage for testing*,' is the best way for the returner. That is, this option maximally satisfies the business aspects for the particular goods return in the *collection* stage of reverse logistics chain.

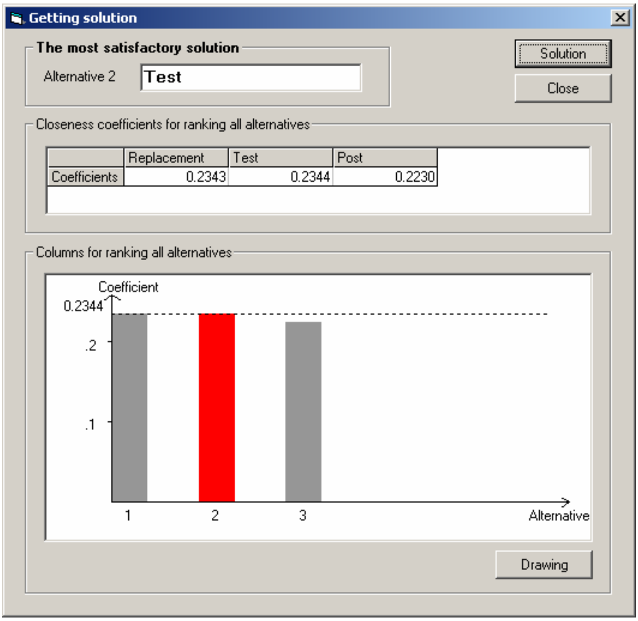


Fig. 16.6: Most satisfactory solution for the returner

16.5 Summary

There is a growing interest in exploiting logistics decision models and developing DSS to enhance logistics management. The interrelated relationship and multi-actors feature in logistics chain management require capabilities of multi-stage multi-criteria decision support. In this chapter, we analysed the characteristics of a reverse logistics chain and built a set of corresponding relationships among goods returners, business aspects, alternatives, and selection criteria. By using the hybrid FMCDM method and the FMCDSS presented from Chapter 9 within the forward logistic channel, a solution that meets maximally the business aspects under the preference of the logistics manager was proposed to handle a goods return in reverse logistics. The proposed method has potential to deal with decision problems.

This page intentionally left blank

Appendix A

User Manual on FMODSS

1. Overview

This user manual briefly describes how to use the Fuzzy Multi-Objective Decision Support System (FMODSS), which is included in the book's companion CD. The system aims to help decision makers gather the knowledge about and obtain possible solutions for the fuzzy multi-objective linear programming (FMOLP) problem.

The FMODSS includes three main components (see Fig. 1.1):

(1) Setting up an FMOLP problem (in the 'File' menu)

(2) Displaying the related information: FMOLP model (in the 'Model' menu) and the running result (in the 'Result' menu)

(3) Solving the FMOLP problem (in the 'Run' menu)

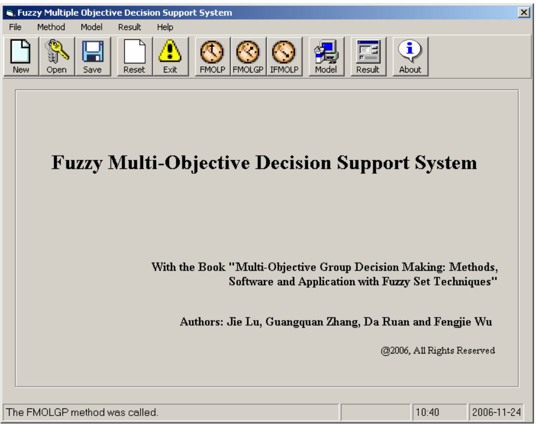


Fig. 1.1: Main interface of the FMODSS

In the 'File' menu, there are five sub-menu items:

- New FMOLP model
- Open FMOLP model
- Save FMOLP model
- Reset system
- Exit

In the 'Method' menu, there are three sub-menu items:

- Fuzzy Multiple Objective Linear Programming (FMOLP)
- Fuzzy Multiple Objective Linear Goal Programming (MFOLGP)
- Interactive Multiple Objective Linear Programming (IFMOLP)

In the 'Model' menu, there is one sub-menu item:

- FMOLP model

In the 'Result' menu, there is one sub-menu item:

- FMOLP result

2. Setting Up an FMOLP Problem

Suppose, we have a production-planning problem as follows:

A company produces two products P_1 and P_2 utilising four different materials M_1, M_2, M_3, and M_4. To produce about 1 ton of P_1 requires about 1 ton of M_1, about 5 tons of M_2, about 4 tons of M_3, and about 3 tons of M_4; while to produce about 1 ton of P_2 requires about 4 tons of M_1, about 3 tons of M_2, about 3 tons of M_3, and about 1 ton of M_4, respectively. The total amounts of available materials are limited to about 21 tons, about 27 tons, about 45 tons, and about 30 tons for M_1, M_2, M_3, and M_4, respectively. By previous experiences, P_1 yields a profit of about 4 million dollars per ton, while P_2 yields about 2 million dollars. P_1 and P_2 contribute about 2 and about 8 units to trading balance, respectively. The two objectives are to maximise the total profit and the trading balance at the same time.

This problem can be modelled as the following FMOLP problem.

$$\text{Max}\begin{pmatrix}\tilde{f}_1(x)\\ \tilde{f}_2(x)\end{pmatrix} = \text{Max}\begin{pmatrix}\tilde{c}_{11}x_1 + \tilde{c}_{12}x_2\\ \tilde{c}_{21}x_1 + \tilde{c}_{22}x_2\end{pmatrix} = \text{Max}\begin{pmatrix}\tilde{4}x_1 + \tilde{2}x_2\\ \tilde{2}x_1 + \tilde{8}x_2\end{pmatrix}$$

$$\text{s.t.}\begin{cases}\tilde{a}_{11}x_1 + \tilde{a}_{12}x_2 = \tilde{1}x_1 + \tilde{4}x_2 \precsim \tilde{b}_1 = \tilde{21}\\ \tilde{a}_{21}x_1 + \tilde{a}_{22}x_2 = \tilde{5}x_1 + \tilde{3}x_2 \precsim \tilde{b}_2 = \tilde{27}\\ \tilde{a}_{31}x_1 + \tilde{a}_{32}x_2 = \tilde{4}x_1 + \tilde{3}x_2 \precsim \tilde{b}_3 = \tilde{45}\\ \tilde{a}_{41}x_1 + \tilde{a}_{42}x_2 = \tilde{3}x_1 + \tilde{1}x_2 \precsim \tilde{b}_4 = \tilde{30}\\ x_1 \geq 0\\ x_2 \geq 0\end{cases}$$

The user can set up a new model for this problem, by clicking the *New FMOLP Model* item in the *File* menu, then a window will be shown as Fig. 2.1. The following common data need to be input for the model in sequence.

(1) The numbers of decision variables, fuzzy objective functions, and fuzzy constraints, respectively (Fig. 2.1).

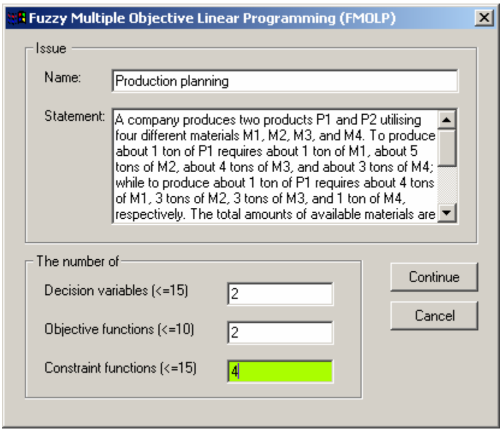


Fig. 2.1: Define an FMOLP model

(2) The names of decision variables (Fig. 2.2), fuzzy objective functions (Fig. 2.3), and fuzzy constraints (Fig. 2.4), respectively.

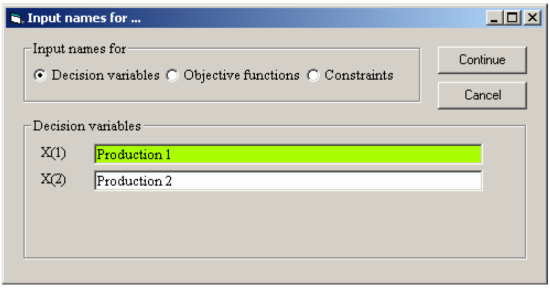


Fig. 2.2: Input the names of decision variables

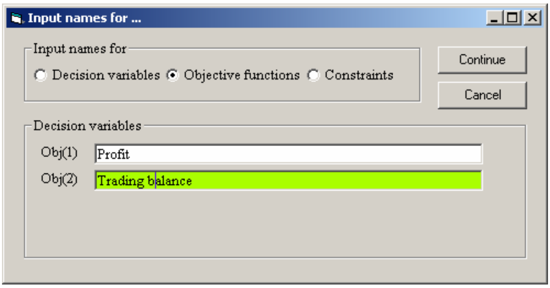


Fig. 2.3: Input the names of fuzzy objective functions

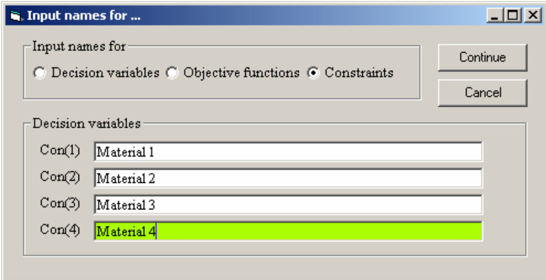


Fig. 2.4: Input the names of fuzzy constraints

(3) The parameters of and the max/min for each fuzzy objective function (Fig. 2.5).

(4) The parameters and the relation sign of each fuzzy constraint (Fig. 2.5).

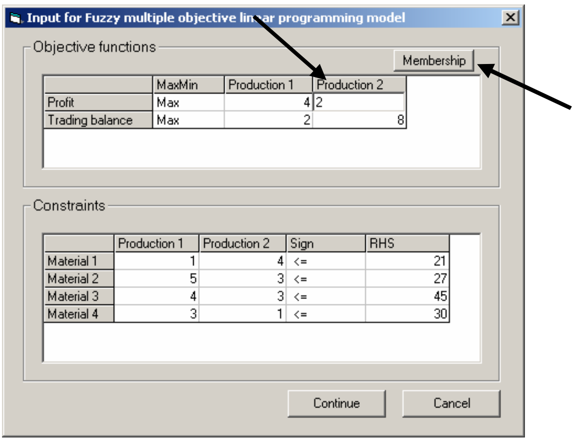


Fig. 2.5: Input fuzzy objective functions and fuzzy constraints

To input these parameters represented by fuzzy numbers includes two steps:

Step 1: Input the fuzzy parameter's value

Double click on the corresponding grid, and then a textbox will appear for the input. For example, in Fig. 2.5, the value of the fuzzy parameter $\tilde{C}_{12} = \tilde{2}$ is input as *2* in the corresponding textbox. If you do not want to use a particular form of membership function, do not go to Step 2.

Step 2: Input the membership function of the fuzzy parameter

Click on the 'Membership' button, a Dialog Box is shown as Fig. 2.6 for entering the membership function of the fuzzy parameter $\tilde{C}_{12}$. As in Fig. 2.6, the forms of both left and right functions of $\tilde{C}_{12}$ are chosen as quadratic, and four end-points of left and right functions are entered as *1*, *2*, *2*, and *4*, respectively.

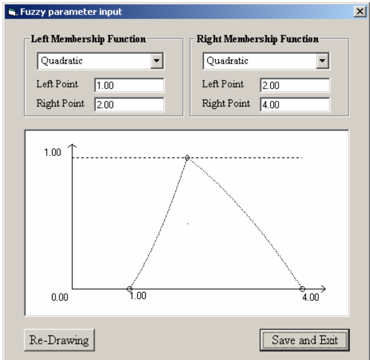


Fig. 2.6: Input the membership function of a fuzzy number

Click on the *FMOLP Model* item in the *Model* menu, the information about the FMOLP problem, will be shown as Fig. 2.7. The fuzzy parameters' values and their membership functions can also be modified in this window.

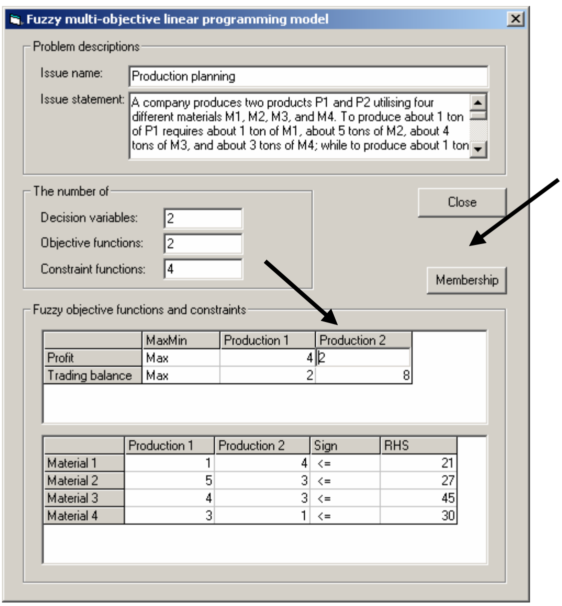


Fig. 2.7: Information about an FMOLP problem

3. Solving the FMOLP Problem

There are three FMOLP methods, FMOLP, FMOLGP, and IFMOLP, implemented in the system. You can use any of them to solve an FMOLP problem.

3.1 *By the FMOLP method*

Click on the *FMOLP* item in the *Method* menu or the *FMOLP* button in the *Toolbar*, a window is shown as Fig. 3.1, in which different weights for fuzzy objective functions can be entered, and the degree α of all membership functions of the fuzzy numbers can be set by the slider as well.

Click on the *Run* button, a solution of the problem is shown in Fig. 3.1: the output of decision variables is: 2.4006 tons for '*Production 1*' and 2.4391 tons for '*Production 2*.' To display membership functions of the fuzzy objective functions '*Profit*' and '*Trading balance*,' click on the corresponding grids and the *Membership* button one by one, new windows will be shown as Fig. 3.2 sequentially. The two figures in Fig. 3.2 show that the value of '*Profit*' is around 14.4808 and the '*Trading balance*' is around 24.3142.

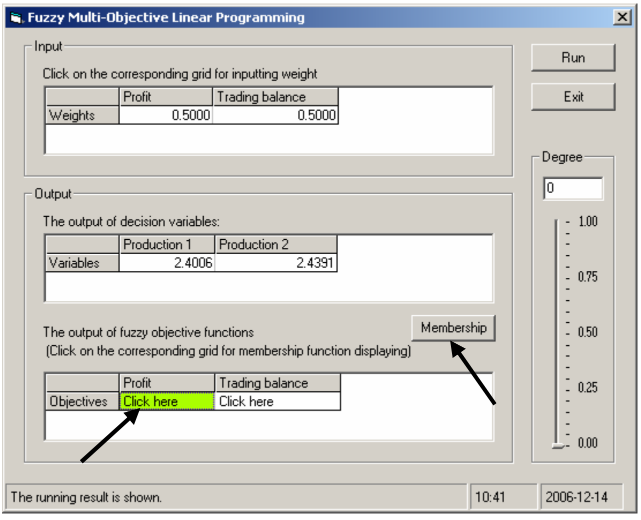


Fig. 3.1: Solving an FMOLP problem by the FMOLP method

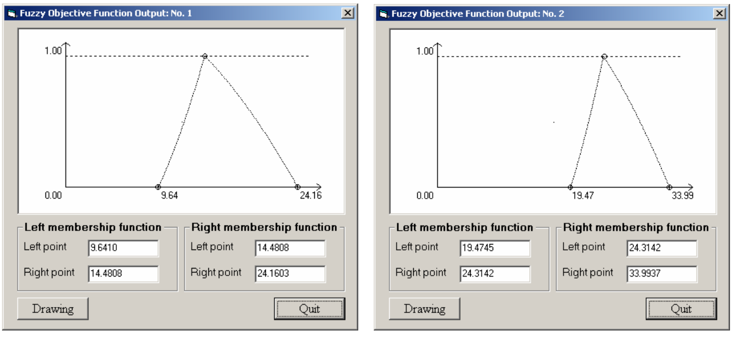


Fig. 3.2: Membership functions of the fuzzy objective results

When the degree α of membership functions is changed to 0.2 shown in Fig. 3.3, click on the *Run* button again, and then we have 2.4835 tons for '*Production 1*' and 2.5916 tons for '*Production 2*'. The membership functions of '*Profit*' and '*Trading balance*' are, in Fig. 3.4, around 15.1171 and around 25.6996, respectively.

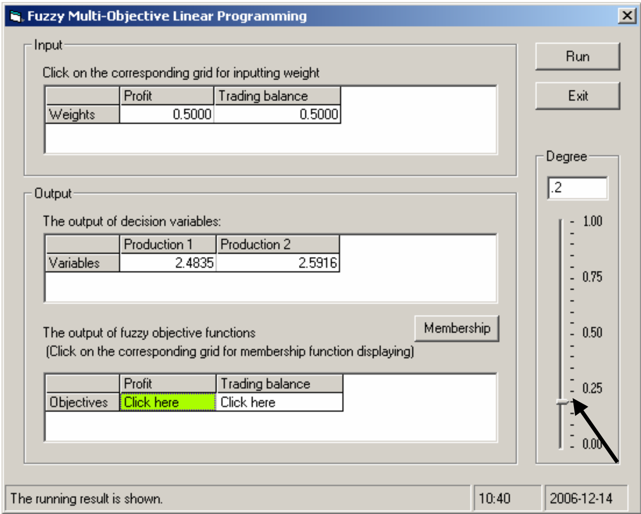


Fig. 3.3: Changing the degree α to 0.2 (w_1=w_2=0.5)

When the weights for '*Profit*' and '*Trading balance*' are changed to 0.8 and 0.2, respectively, the output is: 2.5019 tons for '*Production 1*' and 2.6263 tons for '*Production 2*' (Fig. 3.5). The '*Profit*' and '*Trading balance*', in Fig. 3.6, are around 16.5361 and around 14.1339, respectively.

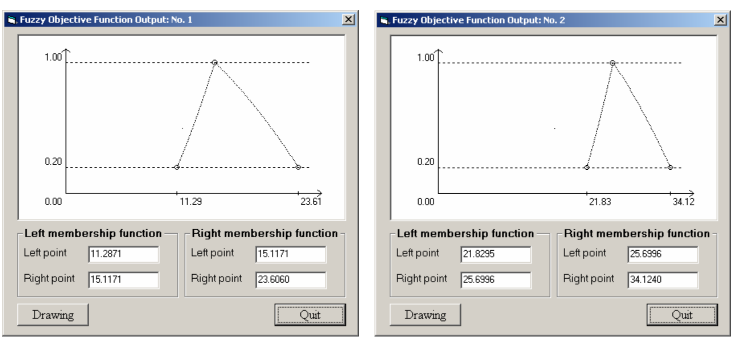


Fig. 3.4: Membership functions of the fuzzy objective results (α=0.2, w_1=w_2=0.5)

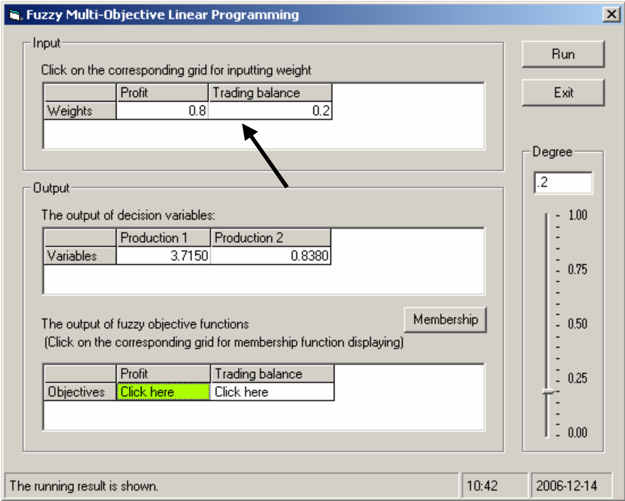


Fig. 3.5: Changing the weight of fuzzy objective functions

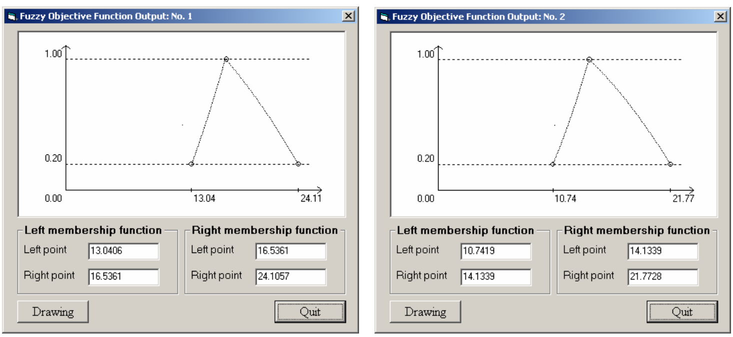


Fig. 3.6: Membership functions of the fuzzy objective results (α=0.2, w_1=0.8, w_2=0.2)

3.2 *By the FMOLGP method*

Click on the *FMOLGP* item in the *Method* menu or the *FMOLGP* button in the *Toolbar*, a window is shown in Fig. 3.7.

The initial fuzzy goals for the fuzzy objective '*Profit*' and '*Trading balance*' should be entered. As the goals are represented by fuzzy numbers, the input of them needs the following two steps:

Step 1: Input the fuzzy goal's value

Double click on the corresponding grid, and then a textbox will appear for the input. For example, in Fig. 3.7, the values of two fuzzy goals are $\tilde{10}$ and $\tilde{15}$, you can therefore input *10* and *15* in the corresponding textbox. If you do not want to use a particular form of membership function, do not go to Step 2.

Step 2: Input the membership function of the fuzzy goal

Click on the *Membership* button, a Dialog Box will be shown for entering the membership function of the fuzzy goal. For example, the two fuzzy goals' membership functions are input as Fig. 3.8, one is around 10, and the other is around 15.

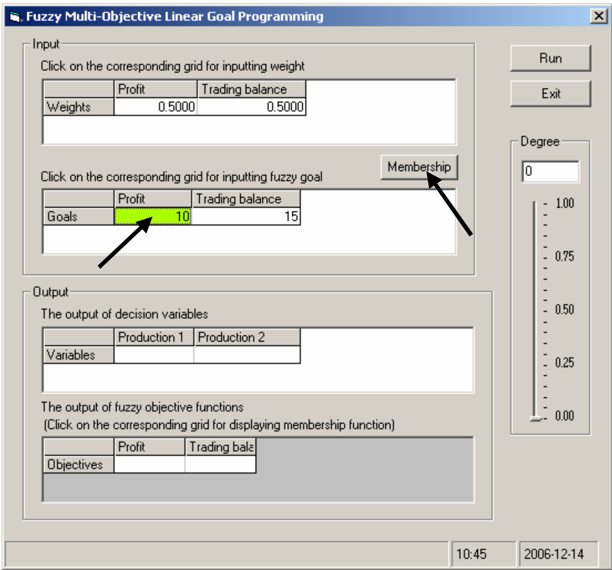


Fig. 3.7: Setting fuzzy goals

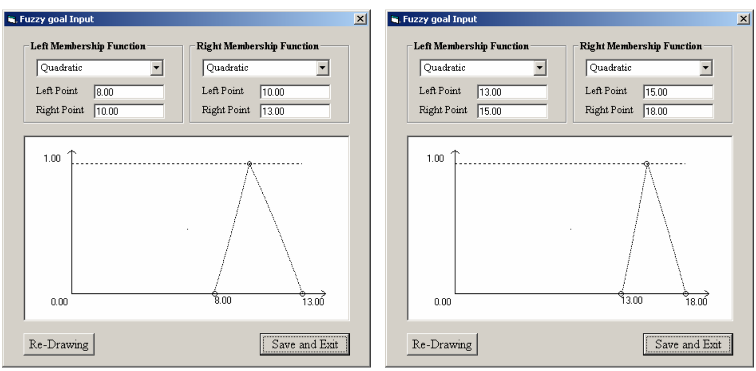


Fig. 3.8: The membership function of the fuzzy goals

Click on the *Run* button, a solution is shown in Fig. 3.9: 2.15 tons for '*Production 1*' and 1.55 tons for '*Production 2*'. To display membership functions of '*Profit*' and '*Trading balance*,' click on the corresponding grids and the *Membership* button, new windows will be displayed as Fig. 3.10, around 11.7 for ‘*Profit*’ and around 16.7 for ‘*Trading balance*.’

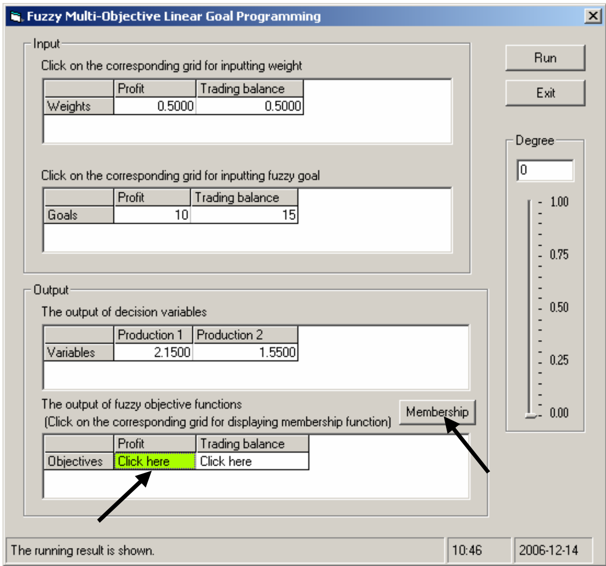


Fig. 3.9: Solving an FMOLP problem by the FMOLGP method

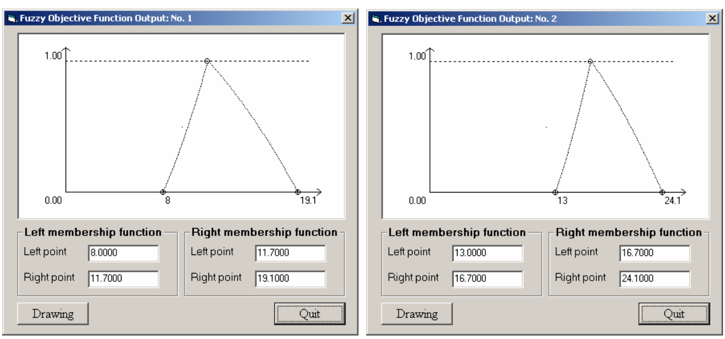


Fig. 3.10: Membership functions of the results for ‘*Profit*’ and ‘*Trading balance*’

3.3 *By the Interactive FMOLP method*

Click on the *IFMOLP* item in the *Method* menu or the *IFMOLP* button in the *Toolbar*, a window is shown as Fig. 3.11. Suppose, the initial degree α is set to *0.3*. Click on the *Initiate* button, an initial solution is: 2.5309 tons for ‘*Production 1*’ and 2.6814 tons for ‘*Production 2*’. The membership functions of ‘*Profit*’ and ‘*Trading balance*’ are shown in Fig. 3.12.

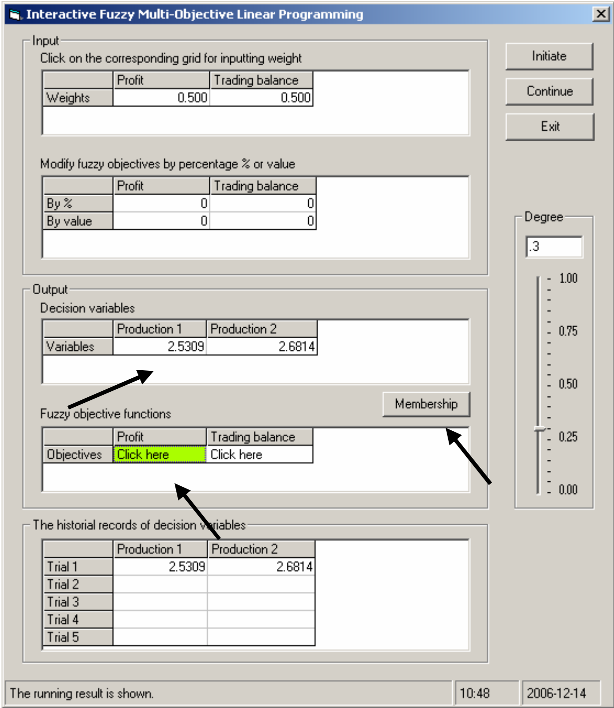


Fig. 3.11: Solving an FMOLP problem by the IFMOLP method

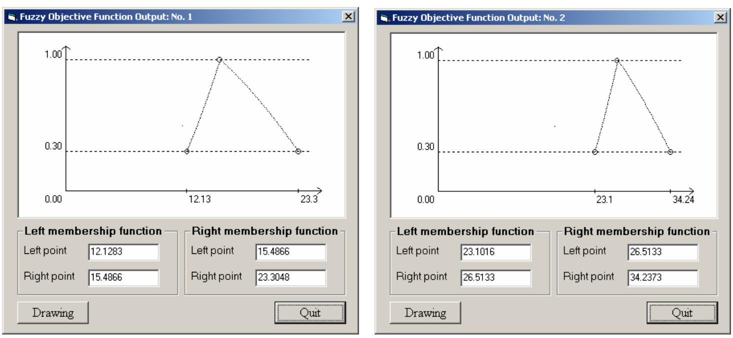


Fig. 3.12: Membership functions of the fuzzy objective function in Trial 1

Suppose, the user is not satisfied with the initial solution, he/she can assign new fuzzy goals for '*Profit*' and '*Trading balance*,' such as by decreasing the '*Profit*' result by 10% and increasing the '*Trading balance*' by 10% as new fuzzy goals based on the initial solution. By clicking the corresponding grids in the row '*By %*' in Fig. 3.13, the increasing and decreasing numbers (-10, and 10) are filled in the textboxes. Click on *Continue* button, the new solution to the problem is generated. The output is: 1.9474 tons for '*Production 1*' and 3.2522 tons for '*Production 2*'.

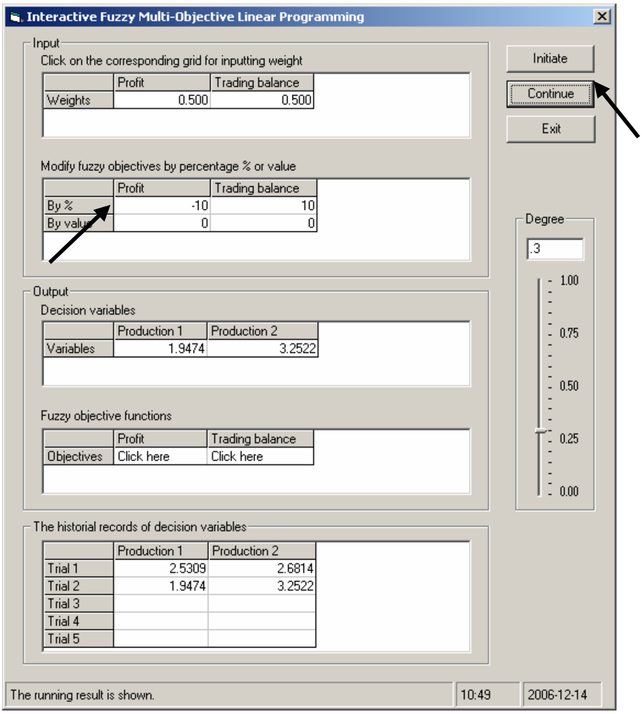


Fig. 3.13: Changing the fuzzy goals by percentage

The user can also set new fuzzy goals by values, which is the same as what we describe in the beginning of this section. If we input two fuzzy goals as *15* and *20*, we will obtain a solution shown in Fig. 3.14.

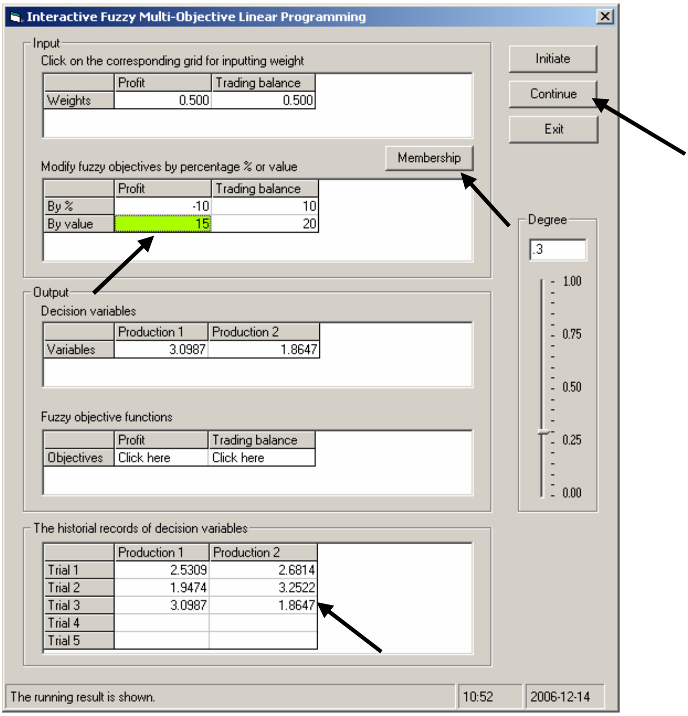


Fig. 3.14: Input new fuzzy goals

This page intentionally left blank

Appendix B

User Manual on FGDSS

This user manual briefly describes how to use the main features of the Fuzzy Group Decision Support System (FGDSS), which is included in the book's companion CD. The system aims to help decision makers manage their group decision making process through criteria generation, alternative evaluation, opinion interaction, and decision aggregation by using linguistic terms.

The FGDSS includes three main components (see Fig. 1)
(1) Generating a group and its problem (in the 'File' menu)
(2) Displaying the group and the problem (in the 'View' menu)
(3) Solving the group decision problem (in the 'Run' menu)

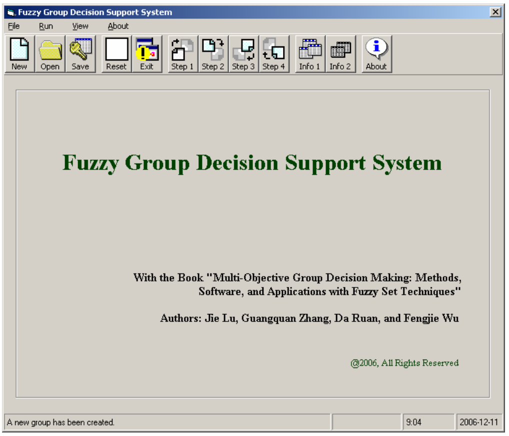


Fig. 1: The main interface of the FGDSS

In the 'File' menu, there are five menu items:

- New group
- Open group
- Save group
- Reset system
- Exit

In the 'Run' menu, there are four menu items:

- Step 1: Input individual criteria
- Step 2: Choose assessment-criteria and weights
- Step 3: Input individual preference
- Step 4: Get solution

In the 'View' menu, there are two menu items:

- Group information (Info 1)
- Alternatives information (Info 2)

As this is an off-line version, when the system is used by a group of members, these members have to use the same computer and input their commands one by one.

The working process with the FGDSS is as follows.

(1) Setting up a decision-making group

Through menu item 'File' -> 'New group' or clicking Button 'New' in the Toolbar, a window is shown Fig. 2 to input:

- The title of the group
- The issue description
- The number of group members
- The number of the alternatives

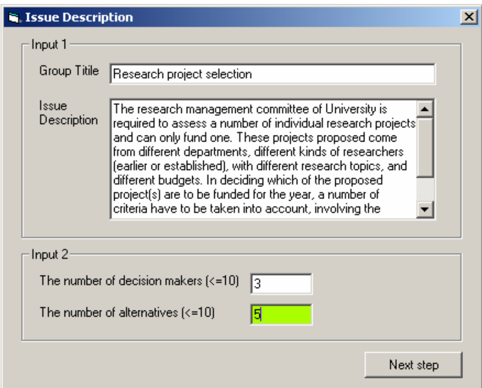


Fig. 2: Set up a group

Click Button 'Next step' (in Fig. 2) to the next window (Fig. 3) to input the names of the group members.

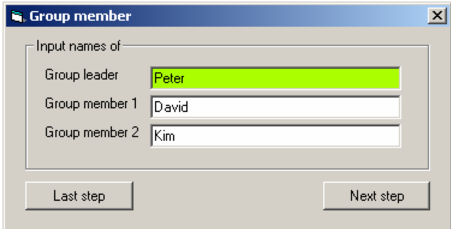


Fig. 3: Input the names of group members

Also, Click Button 'Next step' (in Fig. 3) to the next window (Fig. 4) to input the details of the alternatives.

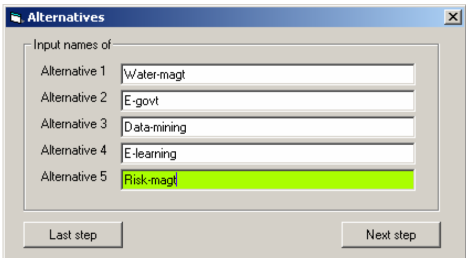


Fig. 4: Input alternatives

(2) Input criteria by all group members

Click Button 'Next step' (in Fig. 4), a window is shown as Fig. 5 for starting the input of criteria. Note: At this stage, Button 'Next step' is false to enable.

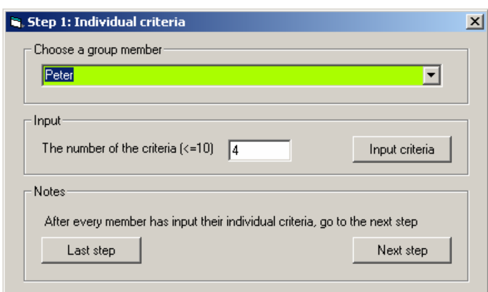


Fig. 5: Input criteria by all group members

In Fig. 5, after having input '4' for the number of the criteria, Peter clicks Button 'Input criteria' to the next window (Fig. 6) to input his four criteria. David then inputs his criteria, and so does Kim.

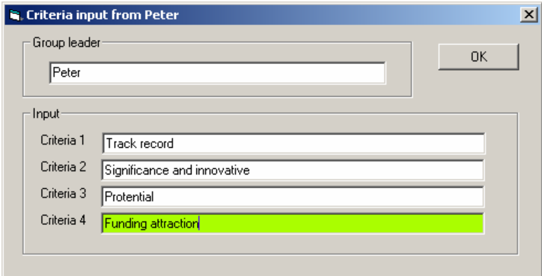


Fig. 6: Input individual criteria

After all group members have input their criteria respectively, Button 'Next step' is changed to enable (Fig. 7).

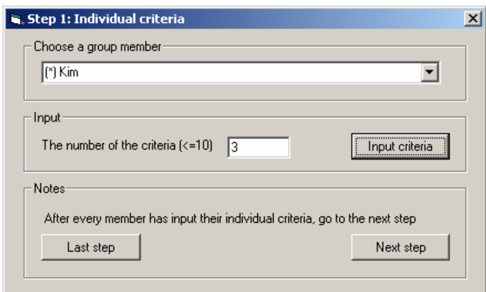


Fig. 7: The status after all group members have input their individual criteria

(3) Choose the top-t criteria and assign weights

Click Button 'Next step' (in Fig. 7), a window is shown as Fig. 8. In the window, each member is assigned with a weight that is described by a linguistic term from: 'Normal,' 'Important,' 'More important,' or 'Most important.'

In Fig. 8, there are 10 individual criteria proposed in total. You can choose some or all of them, and here four of them are chosen as assessment-criteria for the further process.

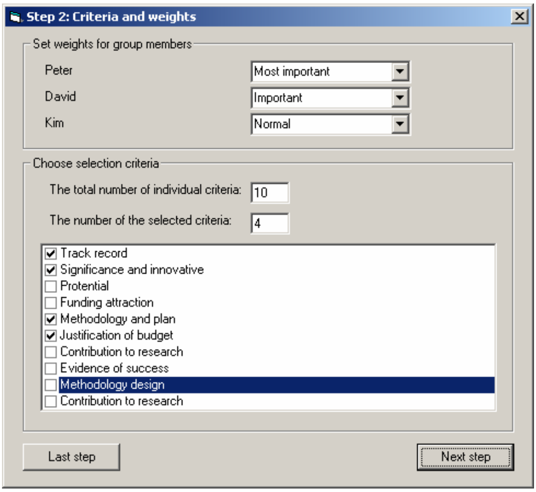


Fig. 8: Choosing the top-t criteria and assigning weights

(4) Fill the criteria comparison matrix and the belief level matrix

Click Button 'Next step' (in Fig. 8), a window is shown as Fig. 9.

Now, each group member needs to fill two matrixes: (1) a pairwise comparison matrix of the relative importance of these criteria, and (2) a belief level matrix to express the possibility of selecting a solution under some criteria.

For the first matrix, obviously, only upper triangle part of the matrix needs to be filled as the matrix is a reciprocal one. The pairwise comparison of any two assessment-criteria is expressed by linguistic terms that represent various degrees of preferences required by decision makers. These possible linguistic terms can be chosen from: 'Absolutely less important,' 'Much less important,' 'Less important,' 'Equally important,' 'More important,' 'Much more important,' 'Absolutely more important,' or 'Cannot be determined yet.'

For the second matrix, against every selection criterion, a belief level is used to express the possibility of selecting a solution under a criterion. The belief level is also expressed by linguistic terms, which can be chosen from: 'Lowest,' 'Very low,' 'Low,' 'Medium,' 'High,' 'Very high,' 'Highest,' or 'Cannot be determined yet.'

After having finished filling the two matrixes, each member must click Button 'Confirm'. After all group members have confirmed their choices, Button 'Next step' is changed to enable for proceeding.

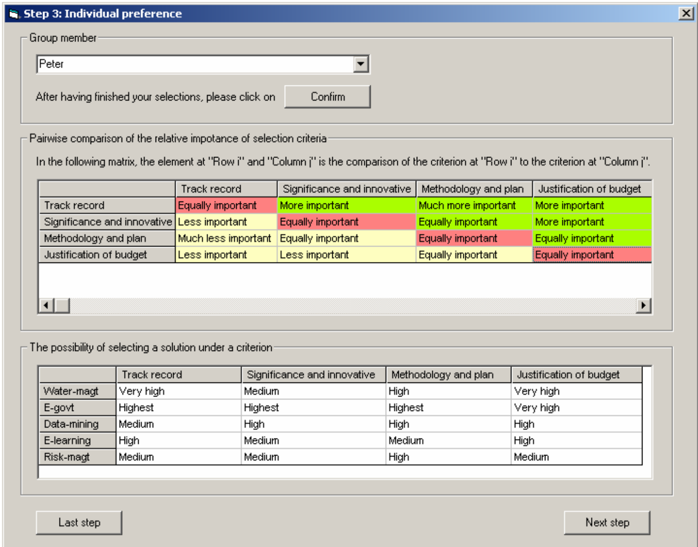


Fig. 9: Filling the criteria comparison matrix and the belief level matrix

(5) <u>Generate the final result of the group-decision making problem</u>

Click Button ‘Next step’ (in Fig. 9), a window is shown as Fig. 10.

In the top frame, the closeness coefficients of all alternative are displayed, which are used for ranking the alternatives. In Fig. 10, the second alternative ‘E-govt’ is with the maximum closeness coefficient (0.5172), and is chosen as the recommended solution to the group decision-making problem.

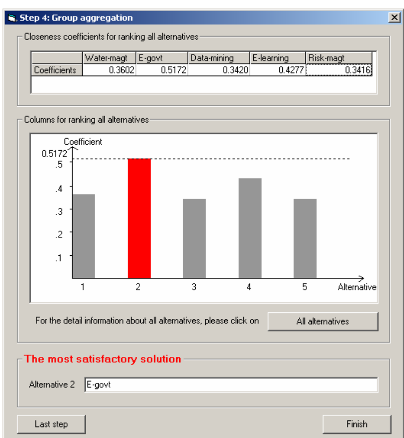


Fig. 10: Showing the result

Bibliography

Abdel-Kader, M. and Dugdale, D. (2001). Evaluating investments in advanced manufacturing technology: a fuzzy set theory approach, *British Accounting Review*, 33, pp. 455-489.

Abido, M.A. (2003). Environmental/economic power dispatch using multi-objective evolutionary algorithms, *IEEE Transactions on Power Systems*, 18, pp. 1529-1537.

Adler, M. and Ziglio, E. (1996). In Gazing into the Oracle: The Delphi Method and Its Application to Social Policy and Public Health*,* Jessica Kingsley Publishers, London.

Agrell, P.J. (1995). A multicriteria framework for inventory control, *International Journal of Production Economics*, 41, pp. 59-70.

Alavi, M. and Keen, P.G.W. (1989). Business teams in an information age, *The Information Society*, 64, pp. 179-195.

Alter, S. (1980). Decision Support Systems: Current Practice and Continuing Challenges*,* Addison-Wesley Pub., Reading, Massachusetts.

Amgoud, L., Belabbes, S. and Prade, H. (2006). Towards a formal framework for the search of a consensus between autonomous agents. *Argumentation in Multi-Agent Systems*. Rahwan, I., Parsons, S. and Maudet, N., Eds., Springer-Verlag, Netherlands, pp. 264-278.

Amgoud, L., Bonnefon, J.-F. and Prade, H. (2005). An argumentation-based approach to multiple criteria decision, *Proc. European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU'2005)*, Barcelona, pp. 269-280.

Amgoud, L. and Prade, H. (2005). Handling threats, rewards and explanatory arguments in a unified setting, *International Journal Of Intelligent Systems*, 20, pp. 1195-1218.

Amgoud, L. and Prade, H. (2006). Explaining qualitative decision under uncertainty by argumentation, *Proc. Twenty-First National Conference on Artificial Intelligence (AAAI-06)*, Boston, pp. 219-224.

Angehrn, A.A. and Jelassi, T. (1994). DSS research and practice in perspective, *Decision Support Systems*, 12, pp. 267-275.

Antunes, C.H., Almeida, L.A., Lopes, V. and Climaco, J.N. (1994). A decision support system dedicated to discrete multiple criteria problems, *Decision Support Systems*, 12, pp. 327-335.

Ba, S., Kalakota, R. and Whinston, A.B. (1997). Using client-broker-server architecture for intranet decision support, Decision Support Systems, 19, pp. 171-192.

Ballou, R. (2004). Business Logistics/Supply Chain Management*,* Pearson Prentice Hall.

Barclay, S. (1987). A User's Manual to HIVIEW, London School of Economics, London.

Barclay, S. (1988). A User's Manual to EQUITY, London School of Economics, London.

Barr, S. and Sharda, R. (1997). Effectiveness of decision support systems - development or reliance effect, *Decision Support Systems*, 21, pp. 133-146.

Bazerman, M.H. (2005). Judgment in Managerial Decision Making (6th Edition), Wiley, New York.

Belton, V. and Vickers, S.P. (1989). Visual Interactive Sensitivity Analyses for Multiple Criteria Decision Analysis: User Manual, SPV Software Products, UK.

Benayoun, R., Montogolfier, J., Tergny , J. and Larichev, O. (1971). Linear programming with multiple objective functions: step method (STEM), *Mathematical Programming*, 1, pp. 366-375.

Berry, M.J.A. and Linoff, G. (1997). Data Mining Techniques, John Wiley & Sons, New York.

Bhargava, H., Krishnan, R. and Muller, R. (1997). Decision support on demanded: emerging electronic markets for decision technologies, *Decision Support Systems*, 19, pp. 193-214.

Bidgoli, H. (1989). Decision Support Systems: Principles and Practice, West Pub. Co., St. Paul.

Bois, P.D., Brans, J.P., Cantraine, F. and Mareschal, B. (1989). Medics: an expert system for computer-aided diagnosis using the PROMETHEE multicriteria method, *European Journal of Operational Research*, 39, pp. 284-292.

Bonczek, R.H., Holsapple, C.W. and Whinston, A.B. (1981). Foundations of Decision Support Systems, Academic Press, New York.

Borenstein, D. (1998). Idssflex - an intelligent DSS for the design and evaluation of flexible manufacturing systems, *Journal of the Operational Research Society*, 49, pp. 734-744.

Bose, U., Davey, A.M. and Olson, D.L. (1997). Multi-attribute utility methods in group decision making: past applications and potential for inclusion in GDSS, *Omega*, 25, pp. 691-706.

Bouchon-Meunier, B., Rifqi, M. and Bothorol, S. (1996). Towards general measures of comparison of objects, *Fuzzy Sets and Systems*, 84, pp. 143-153.

Bouchon-Meunier, B., Yager, R.R. and Zadeh, L.A. (2000a). Information, Uncertainty, Fusion, Kluwer Academic Publishers, Norwell, MA.

Bouchon-Meunier, B., Yager, R.R. and Zadeh, L.A. (2000b). Uncertainty in Intelligent and Information Systems, World Scientific Publishers, Singapore.

Bozdag, C.E., Kahraman, C. and Ruan, D. (2003). Fuzzy group decision making for selection among computer integrated manufacturing systems, *Computers in Industry*, 51, pp. 13-29.

Brito, M.P.D. and Dekker, R. (2002). Reverse Logistics - A Framework, Econometric Institute Report 290, Erasmus University Rotterdam, Econometric Institute.

Bubnicki, Z. (2002). Uncertain Logics, Variables and Systems, Springer, Berlin, New York.

Budgen, D. and Marashi, M. (1988). MDSE advisor: knowledge-based techniques applied to software design assessment, *Knowledge-Based Systems*, 1, pp. 235-239.

Bui, T. (1989). Co-oP: A Group Decision Support System for Cooperative Multiple Criteria Group Decision-Making, Springer-Verlag, Berlin.

Bui, T.X., Jelassi, T.M. and Shakun, M.F. (1990). Group decision and negotiation support systems (GDNSS), *European Journal of Operational Research*, 46, pp. 141-142.

Calpine, H.C. and Golding, A. (1976). Some properties of Pareto-optimal choices in decision problems, *Omega*, 4, pp. 141-147.

Carlsson, C. and Fuller, R. (1996). Fuzzy multiple criteria decision making: recent developments, *Fuzzy Sets and Systems*, 78, pp. 139-153.

Cebeci, U. and Ruan, D. (2007). Appying fuzzy analytic hierarchy process to a multi-attribute comparison of quality consultants, *International Journal of Information Technology & Decision Making (IT&DM)*, in press.

Chan, F.T.S., Chan, M.H. and Tang, N.K.H. (2000). Evaluation methodologies for technology selection, *Journal of Materials Processing Technology*, 107, pp. 330-337.

Chang, D.-Y. (1996). Applications of the extent analysis method on fuzzy AHP, *European Journal of Operational Research*, 95, pp. 649-655.

Chang, P. and Chen, Y. (1994). A fuzzy multicriteria decision making method for technology transfer strategy selection in biotechnology, *Fuzzy Sets and Systems*, 63, pp. 131-139.

Chankong, V. and Haimes, Y.Y. (1983). Multiobjective Decision Making: Theory and Methodology, North-Holland, Amsterdam.

Charnes, A. and Cooper, W.W. (1961). Management Models and Industrial Applications of Linear Programming, Vols. I and II, Wiley.

Charnes, A. and Cooper, W.W. (1977). Goal programming and multiple objective optimizations, *European Journal of Operational Research*, 1, pp. 39-54.

Chaudhry, S.S., Salchenberger, L. and Beheshtian, M. (1996). A small business inventory DSS - design, development, and implementation issues, *Computers & Operations Research*, 23, pp. 63-72.

Chen, S.J. and Hwang, C.L. (1992). Fuzzy Multiple Attribute Decision Making: Methods and Applications, Springer-Verlag, Berlin.

Chen, S.M. (2000). Aggregation fuzzy opinions in the group decision-making environment, *Cybernetics and Systems*, 29, pp. 363-376.

Chen, T. (1987). Decision Analysis, Science Press, Beijing.

Chen, Z.X. (2000). Computational Intelligence Decision Support System. University of Nebraska, Omaha.

Cheng, C.-H. (1996). Evaluating naval tactical missile systems by fuzzy AHP based on the grade value of membership function, *European Journal of Operational Research*, 96, pp. 343-350.

Cheng, C.-H., Yang, K.-L. and Hwang, C.-L. (1999). Evaluating attack helicopters by AHP based on linguistic variable weight, *European Journal of Operational Research*, 116, pp. 423-435.

Chu, T.-C. and Lin, Y.-C. (2003). A fuzzy TOPSIS method for robot selection, *International Journal of Advanced Manufacturing Technology*, 21, pp. 284-290.

Chuang, T.T. and Yadav, S.B. (1998). The development of an adaptive decision support system, *Decision Support Systems*, 24, pp. 73- 87.

Cohon, J.L. (1978). Multiobjective Programming and Planning, Academic Press, New York.

Colson, G. and Mareschal, B. (1994). JUDGES: a descriptive group decision support system for the ranking of items, *Decision Support Systems*, 12, pp. 391-404.

Connell, N.A.D. and Powell, P.L. (1990). A comparison of potential applications of expert systems and decision support systems, *Journal of the Operational Research Society*, 41, pp. 431-439.

Cook, W.D. and Kress, M. (1985). Ordinal ranking with intensity of preference, *Management science*, 31, pp. 26-32.

Csaki, P., Rapcsak, T., Turchanyi, P. and Vermes, M. (1995). R and D for group decision aid in Hungary by WINGDSS, a Microsoft Windows based group decision support system, *Decision Support Systems*, 14, pp. 205-217.

Daily, B., Whatley, A., Ash, S.R. and Steiner, R.L. (1996). The effects of a group decision support system on culturally diverse and culturally homogeneous group decision making, *Information and Management*, 30, pp. 281-289.

Damousis, I.G., G., B.A. and S., D.P. (2003). Network-constrained economic dispatch using real-coded genetic algorithm, *IEEE Transactions on Power Systems*, 18, pp. 198-205.

Davey, A. and Olson, D. (1998). Multiple criteria decision making models in group decision support, *Group Decision and Negotiation*, 7, pp. 55-75.

Dawes, R.M. (1964). Social selection based on multidimentional criteria, *Journal of Abnormal and Social Psychology*, 68, pp. 104-109.

Degoulet, P., Fieschi, M. and Chatellier, G. (1995). Decision support systems from the standpoint of knowledge representation, *Methods of Information in Medicine*, 34, pp. 202-208.

Dennis, A.R. (1996). Information exchange and use in group decision making: you can lead a group to information, but you can't make it think, *MIS Quarterly*, 20, pp. 433-457.

Department of Communicable Disease (Who) (2004). WHO guidelines for the global surveillance of severe acute respiratory syndrome (SARS). http://www.who.int/csr/resources/publications/WHO_CDS_CSR_ARO_2004_1/en/index.html, pp.

Despres, S. and Rosenthal, S.C. (1992). Designing decision support systems and expert systems with a better end-use involvement: a promising approach, *European Journal of Operational Research*, 61, pp. 145-153.

Dowlatshahi, S. (2000). Developing a theory of reverse logistics, *Interfaces*, 30, pp. 143-155.

Downing, C.E. and Ringuest, J.L. (2002). Implementing and testing a complex interactive MOLP algorithm, *Decision Support Systems*, 33, pp. 363-374.

Dubois, D. (1987). An application of fuzzy arithmining processes, *Mathematical Modelling*, 9, pp. 461-475.

Dubois, D. and Fargier, H. (2004). Qualitative decision rules under uncertainty. *Planning Based on Decision Theory*. Della Riccia, G., Dubois, D., Kruse, R. and Lenz, H.-J., Eds., Springer, Wien, pp. 3-26.

Dubois, D., Fargier, H. and Perny, P. (2003). Qualitative decision theory with preference relations and comparative uncertainty: an axiomatic approach, *Artificial Intelligence*, 148, pp. 219-260.

Dubois, D., Fargier, H., Perny, P. and Prade, H. (2003). A characterization of generalized concordance rules in multicriteria decision making, *International Journal of Intelligent Systems*, 18, pp. 751-774.

Dubois, D., Fargier, H. and Prade, H. (1996). Refinements of the maximin approach to decision-making in fuzzy environment, *Fuzzy Sets and Systems*, 81, pp. 103-122.

Dubois, D., Fargier, H., Prade, H. and Perny, P. (2002). Qualitative decision theory: from savage's axioms to nonmonotonic reasoning, *Journal of the ACM*, 49, pp. 455-495.

Dubois, D. and Fortemps, P. (1999). Computing improved optimal solutions to max-min flexible constraint satisfaction problems, *European Journal of Operational Research*, 118, pp. 95-126.

Dubois, D. and Fortemps, P. (2005). Selecting preferred solutions in the minimax approach to dynamic programming problems under flexible constraints, *European Journal of Operational Research*, 160, pp. 582-598.

Dubois, D., Hüllermeier, E. and Prade, H. (2006). Fuzzy methods for case-based recommendation and decision support, *Journal of Intelligent Information Systems, Springer*, 27, pp. 95-115.

Dubois, D. and Prade, H. (1978). Operations on fuzzy numbers, *Int. J. of Systems Science*, 9, pp. 613-626.

Dubois, D. and Prade, H. (1979a). Fuzzy real algebra: some results, *Fuzzy Sets and Systems*, 2, pp. 327-348.

Dubois, D. and Prade, H. (1979b). Operations in a fuzzy-valued logic, *Information & Control*, 43 pp. 224-240.

Dubois, D. and Prade, H. (1980a). Fuzzy sets and systems: Theory and applications, Academic Press, New York.

Dubois, D. and Prade, H. (1980b). Systems of linear fuzzy constraints, *Fuzzy Sets and Systems*, 3, pp. 37-48.

Dubois, D. and Prade, H. (1981). Additions of interactive fuzzy numbers, *IEEE Trans. on Automatic Control*, 26, pp. 926-936.

Dubois, D. and Prade, H. (1983). Ranking fuzzy numbers in the setting of possibility theory, *Information Sciences*, 30, pp. 183-224.

Dubois, D. and Prade, H. (1987). The mean value of a fuzzy number, *Fuzzy Sets and Systems*, 24, pp. 279-300.

Dubois, D. and Prade, H. (1999). Qualitative possibility theory and its applications to constraint satisfaction and decision under uncertainty, *International Journal of Intelligent Systems*, 14, pp. 45-61.

Dubois, D. and Prade, H. (2000). Fundamentals of Fuzzy Sets (The Handbooks of Fuzzy Sets Series), Kluwer, Boston.

Dubois, D. and Prade, H. (2004). On the use of aggregation operations in information fusion processes, *Fuzzy Sets and Systems*, 142, pp. 143-161.

Dubois, D., Prade, H., Roubens, M., Sabbadin, R. and Marichal, J.-L. (2001). The use of the discrete Sugeno integral in decision-making: a survey. , *International Journal of Uncertainty, fuzziness and knowledge-based systems*, 9, pp. 539-561.

Dubois, D., Prade, H. and Sabbadin, R. (2001). Decision-theoretic foundations of qualitative possibility theory, *European Journal of Operational Research*, 128, pp. 459-478.

Dunham, M. (2003). Data Mining, Prentice Hall: Upper Saddle River, NJ.

Dyer, J.S. (1972). Interactive goal programming, *Management Science*, 19, pp. 62-70.

Eden, C. and Ackermann, F. (1996). Horses for courses - a stakeholder approach to the evaluation of GDSS, *Group Decision and Negotiation*, 5, pp. 501-519.

Edwards, J.S. (1992). Expert systems in management and administration - are they really different from decision support systems?, *European Journal of Operational Research*, 61, pp. 114-121.

Endsley, M.R. (1995). Toward a theory of situation awareness in dynamic systems, *Human Factors*, 37, pp. 32-64.

Endsley, M.R. and Garland, D. (2000). Situation Awareness Analysis and Measurement, Lawrence Erlbaum, Associates, Publishers, Mahway, New Jersey.

Engemann, K.J., Holmes, M.E. and Yager, R.R. (2004). Decision making with attitudinal expected values, *International Journal of Technology, Policy and Managemant*, 4, pp. 353-364.

Eom, S.B. (1998). Relationships between the decision support system subspecialties and reference disciplines: an empirical investigation, *European Journal of Operational Research*, 104, pp. 31-45.

Er, M.C. and Ng, A.C. (1995). The anonymity and proximity factors in group decision support systems, *Decision Support Systems*, 14, pp. 75-83.

Ertay, T. and Ruan, D. (2005). Data envelopment analysis based decision model for optimal operator allocation in CMS, *European Journal of Operational Research* 164 pp. 800-810.

Ertay, T., Ruan, D. and Tuzkaya, U.R. (2006). Integrating data envelopment analysis and analytic hierarchy for the facility layout design in manufacturing systems, *Information Sciences*, 176, pp. 237-262.

Expert-Choice-Inc (1992). Expert Choice: Walk-through, Pittsburgh, USA.

Farmer, T.A. (1987). Testing the robustness of multiattribute utility theory in an applied setting, *Decision Sciences*, 18, pp. 178-193.

Fetzer, J.H. (1998). People are not computers: (most) though processes are not computational procedures, *Journal of Experimental and Theoretical Artificial Intelligence*, 10, pp. 371- 391.

Finlay, P.N. (1990). Decision support systems and expert systems: a comparison of their components and design methodologies, *Computers and Operations Research*, 17, pp. 535-543.

Fullér, R. and Zimmermann, H.J. (1993). Fuzzy reasoning for solving fuzzy mathematical programming problems, *Fuzzy Sets and Systems*, 60, pp. 121-133.

Giaglis, G.M., Paul, R.J. and Doukidis, G.I. (1999). Dynamic modelling to assess the business value of electronic commerce, *International Journal of Electronic Commerce*, 3, pp. 35-51.

Giuntini, R. and Andel, T. (1995a). Advance with reverse logistics, *Transportation and Distribution, Part 1*, 36 (2), pp. 93-98.

Giuntini, R. and Andel, T. (1995b). Reverse logistics role models, *Transportation and Distribution, Part 3*, 36 (4), pp. 97-98.

Gomez-Perez, A., Fernandez-Lopez, M. and Corcho, O. (2004). Ontological Engineering, Springer, Heidelberg.

Gooley, T.B. (1998). Reverse logistics: five steps to success, *Logistics Management and Distribution Report*, 37, pp. 49-55.

Gorry, G.A. and Scott-Morton, M.S. (1971). A framework for management information system, *Sloan Management Review*, 13, pp. 55-70.

Gottinger, H.W. and Weimann, P. (1992). Intelligent decision support systems, *Decision Support Systems*, 8, pp. 317-332.

Gray, P. (1987). Group decision support systems, *Decision Support Systems*, 3, pp. 233-242.

Gray, P. and Mandviwalla, M. (1999). New direction for GDSS, *Group Decision and Negotiation*, 8, pp. 77-83.

Guerlian, S., Brown, C.E. and Mastrangelo, C. (2000). Intelligent decision support systems, *Proc. IEEE International Conference on Systems, Man, and Cybernetics*, pp. 1934-1938.

Gülbay, M., Kahraman, C. and Ruan, D. (2004). a-cut fuzzy control charts for linguistic data, *Int. J. of Intelligent Systems*, 19, pp. 1173-1195.

Guyonnet, D., Bourgine, B., Dubois, D., Fargier, H., Côme, B. and Chilès, J.-P. (2003). Hybrid approach for addressing uncertainty in risk assessments, *Journal of Environmental Engineering*, 129 pp. 68-78.

Han, J. and Kamber, M. (2001). Data Mining: Concepts and Techniques, Morgan Kaufmann, San Francisco.

Haseman, W.D., Nazareth, D.L. and Paul, S. (2005). Implementation of a group decision support system utilizing collective memory, *Information & Management*, 42, pp. 591-605.

Herrera, F. and Herrera-Viedma, E. (2000). Choice functions and mechanisms for linguistic preference relations, *European Journal of Operational Research*, 120, pp. 144-161.

Herrera, F. and Verdegay, J.L. (1994). On group decision making under linguistic preferences and fuzzy linguistic quantifiers, *Proc. 5th International Conference on Information Processing and Management of Uncertainty in Knowledge-based Systems*, Paris, pp. 418-422.

Hipel, K. (1992). Multiple objective decision making in water resources, , *Water Resources Bulletin*, 28, pp. 3-12.

Holsapple, C.W. and Whinston, A.B. (1996). Decision Support Systems : A Knowledge-Based Approach, West Pub. Co, Minneapolis, St. Paul.

Hsu, H.M. and Chen, C.T. (1996). Aggregation of fuzzy opinions under group decision making, *Fuzzy Sets and Systems*, 79, pp. 279-285.

Hu, T.L., Sheu, J.B. and Huang, K.H. (2002). A reverse logistics cost minimization model for the treatment of hazardous wastes, *Transportation Research, Part E*, 38, pp. 457-473.

Huang, C.M., Yang, H.T. and Huang, C.L. (1997). Bi-objective power dispatch using fuzzy satisfaction-maximizing decision approach, *IEEE Transactions on Power Systems*, 12, pp. 1715-1721.

Hwang, C.L., Lai, Y.J. and Liu, D.R. (1993). A new approach for multiple objective decision making, *Computer & Operations Research*, 20, pp. 889-899.

Hwang, C.L. and Masud, A.S. (1979). Multiple Objective Decision Making: Methods and Applications, Springer-Verlag, Berlin.

Hwang, C.L. and Yoon, K. (1981). Multiple Attribute Decision Making: Methods and Applications - A State-of-the-Art Survey, Springer-Verlag, Berlin.

Ignizio, J.P. (1976). Goal Programming and Extensions, Lexington Books, Massachusetts.

Ignizio, J.P. (1981). The determination of a subset of efficient solutions via goal programming, *Computer and Operations Research*, 8, pp. 9-16.

Ignizio, J.P. (1983). Generalized goal programming: an overview, *Computer & Operations Research*, 10, pp. 277-289.

Iz, P.H. (1992). Two multiple criteria group decision support systems based on mathematical programming and ranking methods, *European Journal of Operational Research*, 61, pp. 245-253.

Iz, P.H. and Gardiner, L.R. (1993). Analysis of multiple criteria decision support systems for cooperative groups, *Group Decision and Negotiation*, 2, pp. 61-79.

Jacob, V.S. and Pirkul, H. (1992). Organizational decision support systems, *International Journal of Man-Machine Studies*, 36, pp. 817-832.

Jayabarath, T., Jayaprakash, K., Jeyakumar, D.N. and Raghunathan, T. (2005). Evolutionary programming techniques for different kinds of economic dispatch problems, *Electric Power Systems Research*, 73, pp. 169–176.

Jelassi, M.T. and Beauclair, R.A. (1987). An integrated framework for group decision support design, *Information & Management*, 13, pp. 143-153.

Jelassi, M.T., Jarke, M. and Stohr, E.A. (1985). Designing a generalized multiple criteria decision support system, *Journal of Management Information Systems*, 1, pp. 24-43.

Kacprzyk, J., Fedrizzi, M. and Nurmi, H. (1992). Group decision making and consensus under fuzzy preference and fuzzy majority, *Fuzzy Sets and Systems*, 49, pp. 21-31.

Kacprzyk, J. and Yager, R.R. (1985). Management Decision Support Systems Using Fuzzy Sets and Possibility Theory, Verlag TUV Rheinland, Cologne.

Kahraman, C., Beskese, A. and Ruan, D. (2004). Measuring flexibility of computer integrated manufacturing systems using fuzzy cash flow analysis, *Information Sciences*, 168, pp. 77-94.

Kahraman, C., Bozdag, C.E., Ruan, D. and Ozok, A.F. (2004). Fuzzy sets approaches to statistical parametric and nonparametric tests, *Int. J. of Intelligent Systems*, 19, pp. 1069-1087.

Kahraman, C., Cebeci, U. and Ruan, D. (2004). Multi-attribute comparison of catering service companies using fuzzy AHP, *Int. J. of Production Economics*, 87, pp. 171-184.

Kahraman, C., Ruan, D. and Bozdag, C.E. (2003). Optimization of multilevel investments using dynamic programming based on fuzzy cash flows, *Fuzzy Optimization and Decision Making Journal*, 2, pp. 101-122.

Kahraman, C., Ruan, D. and Doğan, I. (2003). Fuzzy group decision-making for facility location selection, *Information Sciences*, 157, pp. 135-153.

Kahraman, C., Ruan, D. and Tolga, E. (2002). Capital budgeting techniques using discounted fuzzy versus probabilistic cash flows, *Information Sciences*, 142, pp. 57-76.

Karacapilidis, N. and Pappis, C. (2000). Computer-supported collaborative argumentation and fuzzy similarity measures in multiple criteria decision-making, *Computers and Operations Research*, 27, pp. 653-671.

Kasper, G.M. (1996). A theory of decision support system design for user calibration, *Information Systems Research*, 7, pp. 215-232.

Kaufmann, A. (1975). Introduction to the Theory of Fuzzy Subsets: Volume I, Academic Press, New York.

Keen, P.G.W. (1976). Interactive computer systems for managers: a modest proposal, *Sloan Management Review*, 18, pp. 1-17.

Keen, P.G.W. and Scott-Morton, M., S. (1978). Decision Support Systems: An Organizational Perspective, Addison-Wesley, Inc., Reading, MA.

Kepner, C.H. and Tregoe, B.B. (1973). Problem Analysis and Decision Making, Kepner-Tregoe Ltd, Princeton, New Jersey.

Kersten, G.E. (1985). NEGO - Group decision support system, *Information & Management*, 8, pp. 237-246.

Klein, M.R. and Methlie, L.B. (1995). Knowledge-Based Decision Support Systems with Applications in Business, John Wiley & Sons.

Klement, E.P., Mesiar, R. and Pap, E. (2000). Triangular Norms, Kluwer Academic Publishers, Dordrecht.

Klimberg, R. (1992). GRADS: A new graphical display system for visualizing multiple criteria solutions, *Computers and Operations Research*, 19, pp. 707-711.

Korhonen, P., Lewandowski, A. and Wallenius, J. (1991). Multiple Criteria Decision Support, Vol. 356 of Lecture Notes in Economics and Mathematical Systems, Springer-Verlag, Berlin, New York.

Korhonen, P. and Wallenius, J. (1990). A Multiple objective linear programming decision support system, *Decision Support Systems*, 6, pp. 243-251.

Kwok, R.C.W., Ma, J. and Zhou, D. (2002). Improving group decision making: a fuzzy GSS approach, *IEEE Transactions on Systems, Man, and Cybernatics - Part C: Applications and Reviews*, 32, pp. 54-63.

Laarhoven, P.J.M. and Pedrycz, W. (1983). A fuzzy extension of Saaty's priority theory, *Fuzzy Sets and Systems*, 11, pp. 229-241.

Lagreze, E.J. and Shakun, M.F. (1984). Decision support systems for semi-structured buying decisions, *European Journal of Operational Research*, 16, pp. 48-58.

Lahti, R.K. Group decision making within the organization: can models help? CSWT paper, from *http://www.eorkteams.unt.edu/reports/lahti.htm.*

Lai, Y.J. (1995). Imost - interactive multiple objective system technique, *Journal of the Operational Research Society*, 46, pp. 958-976.

Lai, Y.J. and Hwang, C.L. (1992). A new approach to some possibilistic linear programming problems, *Fuzzy Sets and Systems*, 49, pp. 121-133.

Lai, Y.J. and Hwang, C.L. (1994). Fuzzy Multiple Objective Decision Making: Methods and Applications, Springer-Verlag, Berlin.

Lai, Y.J., Liu, T.Y. and Hwang, C.L. (1994). Topsis for MODM, *European Journal of Operational Research*, 76, pp. 486-500.

Lambert, D.M. and Cooper, M.C. (2000). Issues in supply chain management, *Industrial marketing Management*, 29, pp. 65-83.

Lee, E.S. and Li, R.J. (1988). Comparison of fuzzy numbers based on the probability measure of fuzzy events, *Computers & Mathematics with Applications*, 15, pp. 887-896.

Lee, H. (1996). Group decision-making using fuzzy set theory for evaluating the rate of aggregative risk in software development, *Fuzzy Sets and Systems*, 80, pp. 261-271.

Lee, M., Pham, H. and Zhang, X. (1999). A methodology for priority setting with application to software development process, *European Journal of Operational Research*, 118, pp. 375-389.

Lee, S.M. (1972). Goal Programming for Decision Analysis, Auerbach Publishers, Philadelphia, Pennsylvania.

Liang, G.-S. (1999). Fuzzy MCDM based on ideal and anti-ideal concepts, *European Journal of Operational Research*, 112, pp. 682-691.

Liberatore, M. and Nydick, R. (2003). Decision Technology: Modeling, Software, and Applications, Wiley, NJ.

Lin, T.S., Yao, Y.Y. and Zadeh, L.A. (2002). Data Mining, Rough Sets and Granular Computing, Physica-Verlag, Heidelberg.

Liu, J., Ruan, D. and Carchon, R. (2002). Synthesis and evaluation analysis of the indicator information in nuclear safeguards applications by computing with words, *Int. J. Appl. Math. Comput. Sci*, 12, pp. 449-462.

Lu, J.(2000). A Framework and Prototype for Intelligent Multiple Objectives Group Decision Support Systems, PhD Thesis, Curtin University of Technology, Australia.

Lu, J. and Lu, Z. (2004). Development, distribution and evaluation for online tourism services in China, *Electronic Commerce Research Journal*, 4, pp. 221-239.

Lu, J. and Quaddus, M.A. (2000). An intelligence-based multi-objective group decision support system and its application, *Proc. Information Technology for Business Management, 16th World Computer Congress (WCC2000)*, Beijing, China, pp. 744-751.

Lu, J. and Quaddus, M.A. (2001a). Integrating a knowledge based guidance system with multiple objective decision making, *The New Zealand Journal of Applied Computer and Information Technology (NZJACIT)*, 5, pp. 53-59.

Lu, J. and Quaddus, M.A. (2001b). A prototype of multi-objective group decision support systems with a group aggregation method base, *Proc. the Twelfth Australasian Conference on Information Systems (ACIS)*, Coffs Harbour, Australia, pp. 378-394.

Lu, J. and Quaddus, M.A. (2002). Experimental evaluation of an intelligent multiple objective group decision support system (IMOGDSS), *Proc. the 9th European conference on IT evaluation*, Paris, France, pp. 273-279.

Lu, J., Ruan, D., Wu, F. and Zhang, G. (2007). An a-fuzzy goal approximate algorithm for solving fuzzy multiple objective linear programming problems, *Soft Computing*, 11, 259-276.

Lu, J., Ruan, D. and Zhang, G. (2006). E-Service Intelligence: Methodologies, Technologies and Applications, Springer, Heideberg.

Lu, J., Ruan, D., Zhang, G. and Zimmermann, H.J. (2006). Editorial Preface, *International Journal of Intelligent Systems*, Special Issue on E-Service Intelligence.

Lu, J., Tang, S. and Mccullough, G. (2001). An assessment for internet-based electronic commerce development in businesses of New Zealand, *Electronic Markets: International Journal of Electronic Commerce and Business Media*, 11, pp. 107-115.

Lu, J., Wu, F. and Zhang, G. (2007). On generalized fuzzy goal optimization for solving fuzzy multi-objective linear programming problems, *Journal of Intelligent and Fuzzy Systems*, in press.

Lu, J. and Zhang, G. (2003a). Cost benefit factor analysis in e-services, *International Journal of Service Industry Management (IJSIM)*, 14, pp. 570-595.

Lu, J. and Zhang, G. (2003b). A model for evaluating e-commerce based on cost/benefit and customer satisfaction, *Journal of Information Systems Frontiers*, 5, pp. 265-277.

Lu, J., Zhang, G. and Shi, C. (2003). Framework and implementation of a web-based WMODSS, *Proc. The Workshop on Applications, Products and Services of Web-based Support Systems, in conjunction with the 2003 IEEE/WIC International Conference on Web Intelligence*, Halifax, Canada, pp. 7-11.

Lu, J., Zhang, G. and Wu, F. (2005). Web-based multi-criteria group decision support system with linguistic term processing function, *The IEEE Intelligent Informatics Bulletin*, 5, pp. 35-43.

Lyles, M.A. and Thomas, H. (1988). Strategic problem formulation: biases and assumptions embedded in alternative decision-making models, *Journal of Management Studies*, 25, pp. 131-145.

Maeda, T. (2001). Fuzzy linear programming problems as bi-criteria optimization problems, *Applied Mathematics and Computation*, 120, pp. 109-121.

Mallach, E.G. (2000). Decision Support and Data Warehouse Systems, McGraw Hill Higher Education, New York.

Management, A.F.N. Gathering Perceptions about the Organization Evaluation of Programs, from *http://www.allianceonline.org/FAQ/strategic_planning/what_is_situation_assessment.faq.*

Marimin, M., Hatono, I. and Tamura, H. (1998). Linguistic labels for expressing fuzzy preference relations in fuzzy group decision making, *IEEE Transactions on Systems, Man, and Cybernetics*, 28, pp. 205-217.

Mccarthy, J. (1986). Applications of circumscription to formalizing common sense knowledge, *Artificial Intelligence*, 28, pp. 89-116.

Minch, R.P. and Sanders, G.L. (1986). Computerized information systems supporting multicriteria decision making, *Decision Sciences*, 17, pp. 395 - 413.

Mitra, S. and Acharya, T. (2003). Data Mining: Multimedia, Soft Computing and Bioinformatics, Wiley, New York.

Moore, C.M. (1987). Group Techniques for Ideal Building, Sage Publications, Newbury Park.

Moreau, E.M.F. (2006). The impact of intelligent decision support systems on intellectual task success: an empirical investigation, *Decision Support Systems*, 42, pp. 593-607.

Morehouse, B. Group systems corporation: corporate overview, from *http://www.groupsystems.com.*

Murofushi, T. and Sugeno, M. (2000). Fuzzy measures and fuzzy integrals. *Fuzzy Measures and Integrals*. Grabisch, M., Murofushi, T. and Sugeno, M., Eds., Physica-Verlag, Heidelberg, pp. 3-41.

Nazareth, D.L. (1993). Integrating MCDM and DSS: barriers and counter strategies, *Infor.*, 31, pp. 1-15.

Nelder, J.A. and Mead, R. (1965). A simplex method for function minimization, *The computer Journal*, 7, pp. 308-313.

Nishizaki, I. and Seo, F. (1994). Interactive support for fuzzy trade-off evaluation in group decision-making, *Fuzzy Sets and Systems*, 68, pp. 309-325.

O'Keefe, R.M. and McEachern, T. (1998). Web-based customer decision support systems, *ACM Transactions on Internet Technology*, 41, pp. 71-78.

Ogryczak, W., Studzinski, K. and Zorychta, K. (1989). A solver for the multi-objective transshipment problem with facility location, *European Journal of Operational Research*, 43, pp. 53-64.

Owens, H.D. and Philippakis, A.S. (1995). Inductive consistency in knowledge-based decision support systems, *Decision Support Systems*, 13, pp. 167-181.

Paige, G.B., Stone, J.J., Lane, L.J. and Hakonson, T.E. (1996). Calibration and testing of simulation models for evaluation of trench cap designs, *Journal of Environmental Quality*, 25, pp. 136-144.

Paige, G.B., Stone, J.J., Lane, L.J., Yakowitz, D.S. and Hakonson, T.E. (1996). Evaluation of a prototype decision support system for selecting trench cap designs, *Journal of Environmental Quality*, 25, pp. 127-135.

Pasi, G. and Yager, R.R. (2006). Modeling the concept of majority opinion in group decision making, *Information Sciences*, 176, pp. 390-414.

Pogorelec, J. (1998). Reverse logistics is doable, important, *Frontline Solutions*, 1, pp. 68-69.

Poh, K. (1998). Knowledge-based guidance system for multi-attribute decision making, *Artificial Intelligence in Engineering*, 12, pp. 315-326.

Poh, K. and Quaddus, M.A. (1990). A hybrid approach to multiobjective linear optimization, *Journal of Operational Research Society*, 41, pp. 1037-1048.

Poh, K., Quaddus, M.A. and Chin, K.L. (1995). MOLP-PC: an interactive decision support environment for multiple objective linear optimization. *OR Applications in Singapore*. Goh, M. C. T. L., Ed., Operational Research Society of Singapore, pp. 25-39.

Power, D.J. and Kaparthi, S. (2002). Building web-based decision support systems, *Studies in Informatics and Control*, 11, pp. 291-302.

Quaddus, M.A. (1997). Computer aided learning of multiple criteria decision making: an integrated system and its effectiveness, *Computers and Education*, 28, pp. 103-111.

Quaddus, M.A. and Holzman, A.G. (1986). IMOLP: an interactive method for multiple objective linear programs, *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-16, pp. 462 - 468.

Quaddus, M.A. and Klass, D. (1998). ALLOCATE: an interactive system for multi-criteria resource allocation. *Optimization Techniques and Application*. Teo, L. C. K. L., Siew, P. F., Leung, Y. H., Jennings, L. S. and Rehbock Eds, V., Eds., Perth, Australia, pp. 471-478.

Radermacher, F.J. (1994). Decision support systems - scope and potential, *Decision Support Systems*, 12, pp. 257-265.

Ramik, J. (2000). Fuzzy goals and fuzzy alternatives in goal programming problems, *Fuzzy Sets and Systems*, 111, pp. 81-86.

Ramik, J. and Rommelfanger, H. (1996). Fuzzy mathematical programming based on some new inequality relations, *Fuzzy Sets and Systems*, 81, pp. 77-87.

Rao, H.R. and Lingaraj, B.P. (1988). Expert systems in production and operations management: classification and prospects, *Interfaces*, 18, pp. 80-91.

Rao, H.R., Sridhar, R. and Narain, S. (1994). An active intelligent decision support system - architecture and simulation, *Decision Support Systems*, 12, pp. 79-91.

Reisman, S., Johnson, T.W. and Mayes, B.T. (1992). Group decision program: a videodisc-based group decision support system, *Decision Support Systems*, 8, pp. 169-180.

Reiter, R. (1980). A logic for default reasoning, *Artificial Intelligence*, 13, pp. 81-132.

Rice, D. (2002). Reverse logistics presents opportunities, *Challenges, Services News*, 1, pp. 1-13.

Richardson, H.L. (2001). Logistics in reverse, *Industry Week 4/16/2001*, 250, pp. 37.

Rogers, D.S., Lambert, D.M., Croxton, K.L. and Garcia-Dastugue, S. (2002). The returns management process, *The International Journal of Logistics Management*, 13, pp. 1-18.

Rommelfanger, H. (1989). Interactive decision making in fuzzy linear optimization problems, *European Journal of Operational Research*, 41, pp. 210-217.

Rommelfanger, H. (1990). FULPAL - an interactive method for solving (multiobjective) fuzzy linear programming problems. *Stochastic Versus Fuzzy Approaches to Multiobjective Mathematical Programming under Uncertainty*. Slowinski, R. and Teghem, J., Eds., Kluwer Academic Publishers, London, pp. 279-299.

Rommelfanger, H. and Slowinski, R. (1998). Fuzzy linear programming with single and multiple objective functions. *Fuzzy Sets in Decision Analysis Operations Research and Statistics - The Handbook of Fuzzy Sets Series*. Slowinski, R., Ed., Kluwer Academic Publishers, London, pp. 179-207.

Rowley, J. (2000). The reverse supply-chain impact of current trends, *Logistics and Transport Focus*, 2, pp. 27-31.

Roy, R. (1971). Problems and methods with multiple objective functions, *Mathematical Programming*, 1, pp. 239-266.

Ruan, D. (1996). Fuzzy Logic Foundations and Industrial Applications, Kluwer, Norwell.

Ruan, D. (2002). Aggregation operators for engineering applications, *Int. J. of Computer Research*, 11, pp. 145-157.

Ruan, D., Liu, J. and Carchon, R. (2003). Linguistic assessment approach for managing nuclear safeguards indicator information, *Int. J. of Logistics Information Management*, 16, pp. 401-419.

Rughooputh, H.C.S. and Ah King, R.T.F. (2003). Environmental/economic dispatch of thermal units using an elitist multi-objective evolutionary algorithm, *Industrial Technology*, 1, pp. 48-53.

Saaty, T. (1980). The Analytic Hierarchy Process, McGraw-Hill, New York.

Sage, A.P. (1991). Decision support systems engineering, Wiley, New York.

Sakawa, M. (1982). Interactive multiobjective decision making by the sequential proxy optimization technique: SPOT, *European Journal of Operational Research*, 9, pp. 386-396.

Sakawa, M. (1993). Fuzzy sets and interactive multiobjective optimization, Plenum Press, New York.

Sakawa, M. and Yano, H. (1990). Interactive decision making for multiobjective programming problems with fuzzy parameters. *Stochastic Versus Fuzzy Approaches to Multiobjective Mathematical Programming under Uncertainty*. Slowinski, R. and Teghem, J., Eds., Kluwer Academic Publishers, Dordrecht, Boston, London, pp. 191-229.

Sakawa, M. and Kato, K. (2000). An interactive fuzzy satisficing method for multiobjective block angular linear programming problems with fuzzy parameters, *Fuzzy Sets and Systems*, 111, pp. 55-69.

Salas, E., Prince, C., Baker, D.P. and Shrestha, L. (1995). Situation awareness in team performance: implications for measurements and training, *Human Factors*, 37, pp. 123-136.

Salukvadze, M. (1974). On the extension of solutions in problems of optimization under vector valued criteria, *Journal of Optimization Theory and Application*, 13, pp. 203-217.

Sauter, V.L. (1997). Decision Support Systems: An Applied Managerial Approach, John Wiley, New York.

Schubert, P. and Dettling, W. (2002). Extended web assessment method (EWAM) - evaluation of e-commerce applications from the customer's view-point, *Proc. the 35th Annual Hawaii International Conference on System Sciences (HICSS'02)*, pp. 175-184.

Schwartz, A.E. (1994). Group decision-making, *The CPA Journal*, 64, pp. 60-64.

Shafer, G. (1976). A Mathematical Theory of Evidence, Princeton University Press, Princeton, N.J.

Shan, F. and Xu, L.D. (1996). A hybrid knowledge-based system for urban development, *Expert Systems with Applications*, 10, pp. 157-163.

Shenoi, S. and Melton, A. (1989). Proximity relations in fuzzy relational databases, *Fuzzy Sets and Systems*, 31, pp. 287-298.

Shim, J.P., Warkentin, M., Courtney, J.F., Power, D.J., Sharda, R. and Carlsson, C. (2002). Past, present, and future of decision support technology, *Decision Support Systems*, 33, pp. 111-126.

Shin, W. and Ravindran, A. (1991). Interactive multiple objective optimization: survey I - continuous case, *Computers and Operations Research*, 18, pp. 97-114.

Sikder, I.U. and Gangopadhyay, A. (2002). Design and implementation of a web-based collaborative spatial decision support system: organizational and managerial implications, *Information Resources Management Journal*, 15, pp. 33-47.

Simon, H.A. (1977). The New Science of Management Decision, Prentice-Hail, Englewood Cliffs, New Jersey, USA.

Slowinski, R. (1990). 'FLIP': an interactive method for multiobjective linear programming with fuzzy coefficients. *Stochastic Versus Fuzzy Approaches to Multiobjective Mathematical Programming under Uncertainty*. Slowinski, R. and Teghem, J., Eds., Kluwer Academic Publishers, Dordrecht, Boston, London, pp. 249-262.

Sol, H.G. (1985). DSS: buzz word or OR challenge, *European Journal of Operational Research*, 22, pp. 1-8.

Sol, H.G. (1987). Conflicting experiences with DSS, *Decision Support Systems*, 3, pp. 203-211.

Sonnenwald, D.H., Maglaughlin, K.L. and Whitton, M.C. (2004). Designing to support situation awareness across distances: an example from a scientific collaboratory, *Information Processing & Management*, 40, pp. 989-1011.

Spangler, W. (1991). The role of artificial intelligence in understanding the strategic decision-making process, *IEEE Transaction on Knowledge and Data Engineering*, 3, pp. 145-159.

Sprague, R.H. (1980). A framework for the development of decision support systems, *Management Information System Quarterly*, 4, pp. 1-26.

Sprague, R.H. (1987). DSS in context, *Decision Support Systems*, 3, pp. 197-202.

Sprague, R.H. and Watson, H.J. (1993). Decision Support Systems: Putting Theory into Practice, 3rd Edition, Prentice-Hall, London.

Sridhar, S. (1998). Decision support using the intranet, *Decision Support Systems*, 23, pp. 19-28.

Steuer, R.E. (1977). An interactive multiple objective linear programming procedure, *TIMS Studies in the Management Sciences*, 6, pp. 225-239.

Steuer, R.E. (1986). Multiple Criteria Optimization: Theory, Computation, and Application, John Willy and Sons, New York.

Steuer, R.E. and Choo, E.U. (1983). An interactive weighted Tchebycheff procedure for multiple objective programming, *Mathematical Programming*, 26, pp. 326-344.

Stewart, T.J. (1991). A multi-criteria decision support system for R&D project selection, *Journal of the Operational Research Society*, 42, pp. 17-26.

Sun, M. and Steuer, R.E. (1996). InterQuad: an interactive quad tree based procedure for solving the discrete alternative multiple criteria, *European Journal of Operational Research*, 89, pp. 462-472.

Sygenex Inc. (1989). CRITERIUM: The Complete Decision Making System, Redmond, USA.

Tanino, T. (1984). Fuzzy preference orderings in group decision making, *Fuzzy Sets and Systems*, 12, pp. 117-132.

Te'eni, D. and Ginzberg, M.J. (1991). Human-computer decision systems: the multiple roles of DSS, *European Journal of Operational Research*, 50, pp. 127-139.

Tecle, A. and Duckstein, L. (1992). A procedure for selecting MCDM techniques for forest resource management, Springer-Verlag, New York.

Tecle, A., Shrestha, B.P. and Duckstein, L. (1998). A multiobjective decision support system for multiresource forest management, *Group Decision and Negotiation*, 7, pp. 23-40.

Teodorescu, H.N., Mlynek, D., Kandel, A. and Zimmermann, H.J. (2000). Intelligent Systems and Interfaces, Springer, Boston

Travica, B. and Cronin, B. (1995). The ARgo: a strategic information system for group decision making, *International Journal of Information Management*, 15, pp. 223-236.

Turban, E. (1996). Implementing decision support system: a survey, *Proc. The 1996 IEEE International Conference on Systems, Man and Cybernetics*, Beijing, China, pp.

Turban, E., Aronson, J.E. and Liang, T.P. (2005). Decision Support Systems and Intelligent Systems (Seventh Edition), Pearson prentice Hall, New Jersey.

Venkatachalam, A.R. and Sohl, J.E. (1999). An intelligent model selection and forecasting system, *Journal of Forecasting*, 18, pp. 167-180.

Venkatesh, P., Gnanadass, R. and Padhy, N.P. (2003). Comparison and application of evolutionary programming techniques to combined economic emission dispatch with line flow constraints, *IEEE Transactions on Power Systems*, 18, pp. 688-697.

Vetschera, R. (1994). MCView: an integrated graphical system to support multi-attribute decision, *Decision Support Systems*, 11, pp. 363-371.

Vogel, D.R. and Nunamaker, J.F. (1990). Group decision support system impact: multi-methodological exploration, *Information and Management*, 18, pp. 15-28.

Voogd, H. (1983). Multicriteria Evaluation for Urban and Regional Planning, Pion, London.

Wang, H.F. (2000). Fuzzy multicriteria decision making - an overview, *Journal of Intelligent and Fuzzy Systems*, 9, pp. 61-83.

Wang, H.F. and Shen, S.Y. (1989). Group decision support with MOLP applications, *IEEE Transactions on Systems, Man, and Cybernatics*, 19, pp. 143-153.

Wang, J. and Lin, Y.I. (2003). A fuzzy multicriteria group decision making approach to select configuration items for software development, *Fuzzy Sets and Systems*, 134, pp. 343-363.

Wang, K.J. and Chien, C.F. (2003). Designing an internet-based group decision support system, *Robotics and Computer-Integrated Manufacturing*, 19, pp. 65-77.

Watson, S.R. and Buede, D.M. (1987). Decision Synthesis: The Principles and Practice of Decision Analysis, Cambridge University Press, Cambridge.

Weck, M., Klocke, F., Schell, H. and Rüenauver, E. (1997). Evaluating alternative production cycles using the extended fuzzy AHP method, *European Journal of Operational Research*, 100, pp. 351-366.

Weistroffer, H.R. (1983). An interactive goal-programming method for nonlinear multiple-criteria decision-making problems, *Computers & Operations Research*, 10, pp. 311-320.

Werners, B. (1987). An interactive fuzzy programming system, *Fuzzy Sets and Systems*, 23, pp. 131-147.

Wong, Z. and Aiken, M. (2003). Automated facilitation of electronic meetings, *Information & Management*, 41, pp. 125-134.

Wu, F., Lu, J. and Zhang, G. (2004a). A decision support system for multiple objective linear programming with fuzzy parameters, *Proc. The 2004 IEEE International Conference on e-Technology, e-Commerce and e-Service(EEE-04)*, Taipei, Taiwan, pp. 425-428.

Wu, F., Lu, J. and Zhang, G. (2004b). FMODSS: A decision support system for solving multiple objective linear programming problem with fuzzy parameters, *Proc. The 2004 IFIP International Conference on Decision Support Systems (DSS2004)*, Prato, Tuscany, Italy, pp. 836-846.

Wu, F., Lu, J. and Zhang, G. (2005a). A decision support system prototype for fuzzy multiple objective optimization, *Proc. 4th Conference of the European Society for Fuzzy Logic and Technology*, Barcelona, Spain, pp. 985-990.

Wu, F., Lu, J. and Zhang, G. (2005b). Development and implementation on a fuzzy multiple objective decision support system, *Proc. 9th International Conference on Knowledge-Based, Intelligent Information & Engineering Systems*, Melbourne, Australia, pp. 261-267.

Wu, F., Lu, J. and Zhang, G. (2005c). A new approximate algorithm for solving multiple objective linear programming problems with fuzzy parameters, *Applied Mathematics and Computation*, 174, pp. 524-544.

Wu, F., Lu, J. and Zhang, G. (2007). The development of a fuzzy multi-objective group decision support system, *Proc. IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2007)*, London, UK, pp. Submitted.

Yager, R.R. (1992a). Higher structures in multi-criteria decision making, *International Journal of Man-Machine Studies*, 36, pp. 553-570.

Yager, R.R. (1992b). On the inclusion of importances in multi-criteria decision making in the fuzzy set framework, *International Journal of Expert Systems: Research and Applications*, 5, pp. 211-228.

Yager, R.R. (1993). Non-numeric multi-criteria multi-person decision making, *International Journal of Group Decision Making and Negotiation*, 2, pp. 81-93.

Yager, R.R. (1999a). Decision making under uncertainty with ordinal information, *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 7, pp. 483-500.

Yager, R.R. (1999b). Decision making with fuzzy probability assessments, *IEEE Transactions on Fuzzy Systems*, 7, pp. 462-467.

Yager, R.R. (1999c). A game theoretic approach to decision making under uncertainty, *International Journal of Intelligent Systems in Accounting, Finance and Management*, 8, pp. 131-143.

Yager, R.R. (1999d). Including decision attitude in probabilistic decision making, *International Journal of Approximate Reasoning*, 21, pp. 1-21.

Yager, R.R. (2000). Fuzzy modeling for intelligent decision making under uncertainty, *IEEE Transactions on Systems, Man and Cybernetics Part B: Cybernetics*, 30, pp. 60-70.

Yager, R.R. (2002a). On the valuation of alternatives for decision making under uncertainty, *International Journal of Intelligent Systems*, 17, pp. 687-707.

Yager, R.R. (2002b). Uncertainty representation using fuzzy measures, *IEEE Transaction on Systems, Man and Cybernetics* 32, pp. 13-20.

Yager, R.R. (2002c). Using a notion of acceptable in uncertain ordinal decision making, *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10, pp. 241-256.

Yager, R.R. (2002d). Using a notion of acceptable in uncertain ordinal decision making, *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10, pp. 241-256.

Yager, R.R. (2003). Monitored heavy fuzzy measures and their role in decision making under uncertainty, *Fuzzy Sets and Systems*, 139, pp. 491-513.

Yager, R.R. (2004a). A framework for multi-source data fusion, *Information Sciences*, 163, pp. 175-200.

Yager, R.R. (2004b). Generalized OWA aggregation operators, *Fuzzy Optimization and Decision Making*, 3, pp. 93-107.

Yager, R.R. (2004c). Modeling prioritized multi-criteria decision making, *IEEE Transactions on Systems, Man and Cybernetics Part B*, 34, pp. 2386-2404.

Yager, R.R. (2004d). On the determination of strength of belief for decision support under uncertainty. Part II: Fusing Strengths of Belief, *Fuzzy Sets and Systems*, 142, pp. 129-142.

Yager, R.R. (2004e). OWA aggregation over a continuous interval argument with applications to decision making, *IEEE Transactions on Systems, Man and Cybernetics Part B*, 34, pp. 1952-1963.

Yager, R.R. (2004f). Uncertainty modeling and decision support, *Reliability Engineering and Systems Safety*, 85, pp. 341-354.

Yager, R.R. (2005). Extending multicriteria decision making by mixing tnorms and OWA operators, *International Journal of Intelligent Systems*, 20, pp. 453-474.

Yager, R.R. and Kacprzyk, J. (1997). The Ordered Weighted Averaging Operation: Theory, Methodology and Applications*,* Kluwer, Norwell, MA.

Yager, R.R., Kacprzyk, J. and Fedrizzi, M. (1994). Advances in the Dempster-Shafer Theory of Evidence*,* John Wiley & Sons, New York.

Yager, R.R. and Kelman, A. (1995). Decision making under various types of uncertainties, *Journal of Intelligent and Fuzzy Systems*, 3, pp. 317-323.

Yager, R.R. and Kikuchi, S. (2004). On the role of anxiety in decisions under possibilistic uncertainty, *IEEE Transactions on Systems, Man and Cybernetics. PartB: Cybernetics* 34, pp. 1224-1234.

Yager, R.R. and Rybalov, A. (1996). Uninorm aggregation operators, *Fuzzy Sets and Systems*, 80, pp. 111-120.

Yager, R.R. and Rybalov, A. (1997). Noncommutative self-identity aggregation, *Fuzzy Sets and Systems*, 85, pp. 73-82.

Yager, R.R. and Zadeh, L.A. (1992). An Introduction to Fuzzy Logic Applications in Intelligent Systems, Kluwer Academic Publisher, Norwell, MA.

Yakowitz, D.S. and Hipel, K.W. (1997). Multiple objective decision making for lokahi (balance) in environmental management, *Applied Mathematics and Computation*, 83, pp. 97-115.

Yakowitz, D.S., Stone, J.J., Lane, L.J., Heilman, P., Masterson, J., Abolt, J. and Imam, B. (1993). A decision support system for evaluating the effects of alternative farm management systems on water quality and economics, *Water Science & Technology*, 28, pp. 47-54.

Yau, C. and Davis, T. (1994). Using multi-criteria analysis for tenant selection, *Decision Support Systems*, 12, pp. 233-244.

Yoon, K. and Hwang, C.L. (1995). Multiple Attribute Decision Making: An Introduction, Sage Publications, London.

You, Y. and Pekkola, S. (2001). Meeting others - supporting situation awareness on the WWW, *Decision Support Systems*, 32, pp. 71-82.

Zadeh, L.A. (1953). A contribution to the theory of nonlinear systems, *J. Franklin Institute*, 255, pp. 387-408.

Zadeh, L.A. (1965). Fuzzy sets, *Information and Control*, 8, pp. 338-353.

Zadeh, L.A. (1968). Probability measures of fuzzy events, *Jour. Math. Analysis and Appl.*, 23, pp. 421-427.

Zadeh, L.A. (1971). Similarity relations and fuzzy orderings, *Information Sciences*, 3, pp. 177-200.

Zadeh, L.A. (1975a). The concept of a linguistic variable and its applications to approximate reasoning. Part I, *Information Sciences*, 8, pp. 199-249.

Zadeh, L.A. (1975b). The concept of a linguistic variable and its applications to approximate reasoning. Part II, *Information Sciences*, 8, pp. 301-357.

Zadeh, L.A. (1975c). The concept of a linguistic variable and its applications to approximate reasoning. Part III, *Information Sciences*, 9, pp. 43-80.

Zadeh, L.A. (1976). A fuzzy-algorithmic approach to the definition of complex or imprecise concepts, *Int. Jour. Man-Machine Studies*, 8, pp. 249-291.

Zadeh, L.A. (1978a). Fuzzy sets as a basis for a theory of possibility, *Fuzzy Sets and Systems* 1, pp. 3-28.

Zadeh, L.A. (1978b). PRUF-a meaning representation language for natural languages *Int. Jour. Man-Machine Studies*, 10, pp. 395-460.

Zadeh, L.A. (1979a). Fuzzy sets and information granularity. *Advances in Fuzzy Set Theory and Applications*. Gupta, M., Ragade, R. and Yager, R. R., Eds., North-Holland Publishing Co., Amsterdam, pp. 3-18.

Zadeh, L.A. (1979b). A theory of approximate reasoning. *Machine Intelligence 9*. Hayes, J., Michie, D. and Mikulich, L. I., Eds., Halstead Press, New York, pp. 149-194.

Zadeh, L.A. (1983). The role of fuzzy logic in the management of uncertainty in expert systems, *Fuzzy Sets and Systems*, 11, pp. 199-227.

Zadeh, L.A. (1989). Knowledge representation in fuzzy logic, *IEEE Trans. on Knowledge and Data Engineering*, 1, pp. 89-100.

Zadeh, L.A. (1994). Fuzzy logic, neural networks and soft computing, *Communications of the ACM*, 37, pp. 77-84.

Zadeh, L.A. (1996). Fuzzy logic = computing with words, *IEEE Transactions on Fuzzy Systems*, 2, pp. 103-111.

Zadeh, L.A. (1997). Toward a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic, *Fuzzy Sets and Systems*, 90, pp. 111-127.

Zadeh, L.A. (1999a). From computing with numbers to computing with words -- from manipulation of measurements to manipulation of perceptions, *IEEE Transactions on Circuits and Systems*, 45, pp. 105-119.

Zadeh, L.A. (1999b). Outline of a computational theory of perceptions based on computing with words. *Soft Computing and Intelligent Systems*. Sinha, N. K., Gupta, M. M. and Zadeh, L. A., Eds., Academic Press, Boston, pp. 3-22.

Zadeh, L.A. (2001). Toward a logic of perceptions based on fuzzy logic. *Discovering the World with Fuzzy Logic*. Novak, W. and Perfilieva, I., Eds., Physica-Verlag, Heidelberg, pp. 4-28.

Zadeh, L.A. and Desoer, C.A. (1963). Linear System Theory-The State Space Approach, McGraw-Hill Book Co., New York.

Zadeh, L.A. and Kacprzyk, J. (1992). Fuzzy Logic for the Management of Uncertainty, John Wiley & Sons, New York.

Zeleny, M. (1982). Multiple Criteria Decision Making, McGraw-Hill, New York.

Zhang, B., Zhang, G. and Lu, J. (2003). A system for solving fuzzy linear programming problems by multi-objective linear programming, *Proc. International Conference on Fuzzy Information Processing*, Beijing, China, pp. 675-680.

Zhang, G. (1998). Fuzzy Number-Valued Measure Theory, Tsinghua University Press, Beijing.

Zhang, G. and Lu, J. (2002). Group decision-making method with fuzzy preference of decision makers, *Proc. The International Conference of Group Decision and Negotiation*, Perth, Australia, pp. 443-454.

Zhang, G. and Lu, J. (2003a). A group decision making method with fuzzy weights for decision makers, fuzzy preferences for alternatives and fuzzy judgments for selection criteria, *Proc. International Conference on Fuzzy Information Processing*, Beijing, China, pp. 655-661.

Zhang, G. and Lu, J. (2003b). An integrated group decision making method with fuzzy preference for alternatives and individual judgments for selection criteria, *Group Decision and Negotiation*, 12, pp. 501-515.

Zhang, G. and Lu, J. (2003c). Using general fuzzy number to handle fuzziness in group decision making, *Proc. 7th Joint Conference on Information Sciences*, Cary, North Carolina, USA, pp. 75-179.

Zhang, G. and Lu, J. (2004). Using general fuzzy numbers to handle uncertainty and imprecision in group decision-making. *Intelligent Sensory Evaluation: Methodologies and Applications*. Ruan and Zeng, Eds., Springer, pp. 51-70.

Zhang, G. and Lu, J. (2007). A linguistic intelligent guide for method selection in multi-objective decision support systems, *Special issue on Linguistic Decision Making: Tools and Applications, Information Science*, in press.

Zhang, G., Lu, J. and Wu, Y. (2003). Formulation of linear programming problems with fuzzy coefficients of objective functions and constraints, *Asian Information-Science-Life*, 2, pp. 57-68.

Zhang, G., Wu, Y., Remias, M. and Lu, J. (2003). Formulation of fuzzy linear programming problems as four-objective constrained optimization problems, *Applied Mathematics and Computation*, 39, pp. 383-399.

Zhang, W. and Hill, R. (2000). A template-based and pattern-driven approach to situation awareness and assessment in virtual humans, *Proc. the Fourth International Conference on Autonomous Agents*, Spain, pp. 116-123.

Zhou, C. and Ruan, D. (1999). Integration of linguistic and numerical information for biped control, Robotics and Autonomous Systems, *Robotics and Autonomous Systems*, 28, pp. 53-70.

Zhou, C. and Ruan, D. (2002). Fuzzy control rules extraction from perception-based information using computing with words, *Information Sciences*, 142, pp. 2002.

Zhu, K.-J., Jing, Y. and Chang, D.-Y. (1999). A discussion on extent analysis method and applications of fuzzy AHP, *European Journal of Operational Research*, 116, pp. 450-456.

Zimmermann, H.J. (1983). Fuzzy mathematical programming, *Computer & Operations Research*, 10, pp. 291-298.

Zimmermann, H.J. (1986). Multi-criteria decision making in crisp and fuzzy Environments. *Fuzzy Set Theory and Appliations*. Jones, A., Kaufmann, A. and Zimmermann, H. J., Eds., Reidel Publ. Comp., Dordrecht, Boston, Lancaster, pp. 233-256.

Zimmermann, H.J. (1987). Fuzzy Sets, Decision Making, and Expert Systems, Kluwer Academic Publishers, Boston.

Zimmermann, H.J. (1988). Interactive decision support for semi-structured mathematical programming models. *Mathematical Models for Decision Support*. Mitra, G., Ed., Springer-Verlag, Berlin, Heidelberg, pp. 307-319.

Zimmermann, H.J. (2000). Fuzzy Sets and Operations Research for Decision Support, Beijing Normal University Press, Beijing, China.

Zimmermann, H.J. (2001). Fuzzy Set Theory and its Applications (Fourth Edition), Kluwer, Boston.

Zimmermann, H.J. and Witte, E. (1986). Empirical Research on Organisational Decision Making, North Holland, Amsterdam.

Zimmermann, H.J., Zadeh, L.A. and Gaines, B.R. (1984). Fuzzy Sets and Decision Analysis, North Holland, Amsterdam, New York.

Zimmermann, H.J. and Zysno, P. (1983). Decisions and evaluations by hierarchical aggregation of information, *Fuzzy Sets and Systems*, 10, pp. 243-260.

Ziont, S. and Wallenius, J. (1975). An interactive programming method for solving the multiple criteria problem, *Management Science*, 22, pp. 652-663.

This page intentionally left blank

Abbreviation

AHP: Analytic Hierarchy Process
AI: Artificial Intelligence
ASM: Average Solution Method
ASP: Application Service Providers
CMOLP: Constrained Multi-Objective Linear Programming
CSCW: Computer-Supported Cooperative Work
DBMS: Database Management Systems
DINAS: Dynamic Interactive Network Analysis Systems
DSS: Decision Support Systems
EMS: Electronic Meeting Systems
ESGP: Efficient Solution via Goal Programming
FDEELD: Fuzzy Dynamic Environmental Economic Load Dispatch
FGDSS: Fuzzy Group Decision Support Systems
FGP: Fuzzy Goal Programming
FMCDM: Fuzzy Multi-Criteria Decision Making
FMCDSS: Fuzzy Multi-Criteria Decision Support Systems
FMCGDM: Fuzzy Multi-Criteria Group Decision Making
FMCGDSS: Fuzzy Multi-Criteria Group Decision Support Systems
FMODM: Fuzzy Multi-Objective Decision Making
FMODSS: Fuzzy Multi-Objective Decision Support Systems
FMOGDM: Fuzzy Multi-Objective Group Decision Making
FMOGDSS: Fuzzy Multi-Objective Group Decision Support Systems
FMOLGP: Fuzzy Multi-Objective Linear Goal Programming
FMOLP: Fuzzy Multi-Objective Linear Programming
GA: Genetic Algorithms
GDM: Group Decision Making
GDSS: Group Decision Support Systems
GP: Goal Programming

GSS: Group Support Systems

GUI: Graphical User Interface

IFMOLP: Interactive Fuzzy Multi-Objective Linear Programming

IMOGDSS: Intelligent Multi-Objective Group Decision Support Systems

IMOLP: Interactive Multi-Objective Linear Program

ISGP: Interactive Sequential Goal Programming

ISM: Ideal Solution Method

KBS/ES: Knowledge-Based Systems/ Expert Systems

LGP: Linear Goal Programming

LP: Linear Programming

MADM: Multi-Attribute Decision Making

MADSS: Multi-Attributes Decision Support Systems

MAGDM: Multi-Attribute Group Decision Making

MCDM: Multi-Criteria Decision Making

MCDSS: Multi-Criteria Decision Support Systems

MODM: Multi-Objective Decision Making

MODSS: Multi-Objective Decision Support Systems

MOGDM: Multi-Objective Group Decision Making

MOGDSS: Multi-Objective Group Decision Support Systems

MOLP: Multi-Objective Linear Programming

OLAP: Online Analytical Processing

SA: Situation Awareness

SAM: Solution Analysis Method

TOPSIS: Technique for Order Preference by Similarity to Ideal Solution

VIG: Visual Interactive Goal Programming

WFGDSS: Web-Based Fuzzy Group Decision Support Systems

WIPM: Weighting Ideal Point Method

WMLP: Weighted Maximum Linear Programming

WMM: Weighting Member Method

WOM: Weighting Objective Method

Index

G

H

I

K

L

M

N

O

P

Q

R

S

T

U

V

W

Z