FUZZY PROJECT-NETWORK SCHEDULING UNDER RESOURCE CONSTRAINTS

Pasit Lorterapong

A Thesis

in

The Centre for Building Studies

Presented in Partial Fulfilment of the Requirements
for the Degree of Doctor of Philosophy at
Concordia University
Montreal, Quebec, Canada

April 1995

ISBN 0-612-05067-X

ABSTRACT

Fuzzy Project Network Scheduling
Under Resource Constraints

Pasit Lorterapong, Ph.D.
Concordia University, 1995

Resource constraints are frequently encountered in most engineering and construction organizations. This problem has traditionally been treated using optimization and heuristic techniques. Optimization techniques aim at providing the best solution. However, they require excessive computational efforts for solving large-scale problems. Heuristic solutions, on the other hand, do not guarantee the optimal solution, but are applicable to real-life problems. Heuristic solutions, however, have often been criticized for their inconsistency. Both optimization and heuristic techniques have been developed for deterministic-type problems. In the project environment where uncertainties exist, these deterministic solutions may prove inadequate.

To schedule projects in such an environment, a flexible heuristic method has been developed. The method integrates a newly developed resource allocation model with a suitable technique for modeling uncertainties in construction scheduling. The proposed resource allocation model incorporates a decomposition technique that generates partial schedule alternatives and examines the negative impact of each on the overall project duration. The performance of the model is tested against those of the widely used heuristic techniques, considering three objectives: project durations, resource utilization, and resource interruptions. The results indicate that the proposed model statistically outperforms the traditional heuristic techniques in all of the three objectives measured. In addition, the proposed model tends to generate schedules that satisfy, simultaneously, the

three objectives more than any of the traditional heuristic rules.

The present method employs fuzzy set theory for modeling the uncertainties associated with the durations of project activities and the resource availabilities. Various degrees of uncertainties associated with construction scheduling can be represented using fuzzy numbers with simple triangular and trapezoidal approximations. These fuzzy numbers are expressed in a quadruple format that facilitates efficient calculations. Fuzzy bounds are used to overcome a major limitation of the fuzzy subtraction which, in the past, hindered its application to complete network scheduling.

A prototype software system has been developed to demonstrate the use of the proposed method and to illustrate its capabilities. Considerable attentions have been focused on facilitating user interactions with the system. The system acts as an assistant to the scheduler in resolving resource conflicts by allowing the scheduler to generate and examine partial scheduling alternatives before allocating resources. One of the interesting features of the present system is that it supports both automatic and interactive resource allocation operations. Two numerical examples are presented to demonstrate the effectiveness and practicality of the system features. It has been shown that the system can be used as a decision support tool for improving and facilitating the generation of schedules under nondeterministic resource constraints in single and multiple project environments.

ACKNOWLEDGEMENTS

I wish to express my deepest gratitude and appreciation to my supervisor Dr. Osama Moselhi for his support, encouragement, and guidance throughout the course of this study.

I am thankful to the faculty, staff, and my colleagues at the Centre for Building Studies for their help and hospitality during my studies at Concordia.

I would like to acknowledge the Canadian International Development Agency for providing financial support for this research.

I want to thank my wife, Lek, and my children, Oum & Ohm, for being my strength and my will, and for their invaluable love, care, and patience during all the pleasant and difficult times, while being graduate students at Concordia.

Finally, I wish to express my special thanks to my father and mother, to whom I dedicate this thesis, for their continuous love and support.

Table of Contents

NOMENCLATURE

A,B,M,N, etc.	=	trapezoidal fuzzy numbers;
AI(A,H)	=	agreement index of A with respect to H;
AI	=	agreement index;
CF	=	contiguous float;
CP	=	cumulative probability;
CT	=	current time;
C^A	=	geometric centroid of a trapezoidal fuzzy number A;
F	=	degree of availabilities or demands;
ED	=	expected delay;
EFT_x	=	early finish time of activity x;
EST_x	=	early start time of activity x;
Fd_x	=	fuzzy duration of activity x;
FES_x, FEF_x	=	fuzzy early (start/finish), and
FLS_x, FLF_x	=	fuzzy late (start/finish) for activity x;
FLF^u_x	=	upper-bound values for fuzzy late finish;
FS_x	=	fuzzy start for activity x;
FF_x	=	fuzzy finish for activity x;
IF	=	intermediate float;
LF	=	logic float;
LFT_x	=	late finish time of activity x;
NTF	=	next time frame;
NI_X	=	negative impact for activity set X;
ni_X	=	negative impact for activity x;
P	=	set of in-progress activities;
PM	=	possibility measure;
R	=	amount of resource availability;
r	=	amount of resource demand;
RU	=	amount of resource in-use
RTF	=	remaining total float;

S(A)	=	support of fuzzy set **A**;
SST	=	scheduled start;
SFT	=	scheduled finish;
T	=	universe of discourse;
t	=	element of fuzzy set A;
PLF	=	preliminary Late finish;
Q	=	resource quantity;
$R_{j,n}$	=	fuzzy resource quantity type j at time interval n;
T_{proj}	=	fuzzy project duration;
TF_x	=	fuzzy total float of activity x;
w	=	weight factor;
$t \in A$	=	t is a member of fuzzy subset A;
$\mu_A(t)$	=	membership function for the element t with respect to the fuzzy subset A;
α	=	alpha-cut level;
$\neg F$	=	opposite event of fuzzy number F
$\cap$	=	intersection;
$\wedge$, $\vee$	=	minimum (maximum);
$\ominus$, $\oplus$	=	Fuzzy subtraction (addition);
$\tilde{max}$, $\tilde{min}$	=	Fuzzy maximum (minimum);
ni	=	negative impact;
Poss {X is F}	=	Possibility that fuzzy subset X is F;
Sup	=	Supremum;

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Resource Scheduling

Scheduling can generally be described as the process of estimating the duration of project activities, and allocating resources to those activities so as to achieve organizational objectives and satisfy any constraints that may exist (Baker 1974). Scheduling objectives typically include, for example, minimizing project completion times, maximizing resource utilizations, minimizing the fluctuation of resource demands, etc. Constraints, in general, are restrictions imposed on the project execution which often exist in various types (e.g. technical, safety, time, resources constraints, etc). Among those constraints, time and resource limitations have been reported as the top two challenges that are most frequently encountered by project managers (Thamhain and Wilemon 1986).

Time constraints occur in the form of milestone events, contractual dates, etc. Resource constraints, on the other hand, relate to the availability in the resource pool at any given time-periods. The main objective of scheduling a time-constrained project is to minimize the variation of resource demands over a period of time (Easa 1989, Harris 1990, Savin et al. 1994), assuming that resources are sufficiently available. The process employed to achieve this objective is referred to as resource leveling or resource smoothing (Moselhi and Lorterapong 1993a).

On the other hand, if the amount of resources are limited, the unconstrained project duration obtained using a well-known technique such as the critical path method (CPM) might not be achievable. The revised objective would then be to minimize the unavoidable project delay due to any arisen resource conflicts. This latter process, which is the main focus of this study, is called resource-constrained scheduling or simply resource allocation. The outcome of scheduling is the established project time frame (i.e. the start and finish times of each activity) and the resources required to complete the project. The quality of a schedule can be measured by the degrees to which project objectives are achieved, and the degrees to which project constraints are satisfied.

The problem of allocating an adequate amount of resources to the appropriate activity at the right time is not new. It has received a considerable attention from researchers and practitioners for the past several decades (Davis and Heidorn 1971, Davis 1973, Davis and Patterson 1975, Kurtulus and Davis 1982, Lawrence 1985, Khashenas and Haber 1990). The main reason, perhaps, is the technological advances which continue to open new and ever promising avenues thereby allowing this increasingly complicated problem to be dealt with in a more effective manner.

1.2 Complexity of Resource-Constrained Scheduling

The resource-constrained scheduling problem varies in type and in intensity, depending upon the nature of the project. In some projects, there could be only one type of resource (e.g. cranes, skilled workers) that causes a bottle-neck in scheduling. Others may involve in a more complicated situation such as multiple projects and multiple resource conflict environments. Scheduling the project activities in such an environment so as to satisfy both precedence and resource constraints, while attempting to meet other objectives simultaneously represents a difficult task (Weist and Levy 1977, Kim and Schniederjans 1989).

Resource-constrained scheduling problems have been challenged using both optimization and heuristic methods. The optimization methods aim at providing the best solutions using the well known optimization techniques such as integer programming (Pritsker et al. 1969, Talbot and Patterson 1978, Khashenas and Haber 1990) and various enumeration procedures (Davis and Heidorn 1971, Stinson et al. 1978, Talbot and Patterson 1979, Patterson 1984, and Christofides et al. 1987). These traditional optimization techniques have constantly been criticized for their excessive computational requirements to handle most real-life problems (Weist and Levy 1977, Kurtulus and Davis 1982, Allam 1988, Hadavi et al. 1992, and Dumond 1992).

To overcome these shortcomings, a number of heuristic resource allocation techniques have been developed (Panwalkar and Iskander 1977, and Lawrence 1985). These techniques do not guarantee optimal solutions. Instead, they provide feasible solutions to practical problems. Heuristic solutions also have some deficiencies. For example, they are designed to achieve a single scheduling objective at a time (e.g. the minimum slack rule is effective for providing minimal project delays). Another major limitation associated with most heuristic rules is that different rules perform with varying degrees of effectiveness when applied to different networks (Davis and Patterson 1975, Allam 1988, Shanmuganayagum 1989). In light of these findings, it is apparent that there is a significant room for improving the performance of traditional heuristic rules.

Another important aspect which has often been disregarded in most resource-constrained scheduling techniques is the uncertainties that are inherent in most construction projects. This uncertainty can have a negative impact either directly or indirectly on project activities and their required resources. This stems from the fact that activities and resources usually exhibit some degree of relationship among each other. As such, uncertainties affect not only the entity that they are directly associated with, but also some related entities. The integrity of a schedule can be compromised if the present uncertainty is not properly considered.

As far as the literature is concerned, there have been few attempts to incorporate uncertainties in resource-constrained scheduling (Ahuja and Nandakumar 1986, Chang et al. 1991, and Padilla and Carr 1991). Except for the study done by Chang et al. (1990), previous studies base their solutions primarily on probability theory. In probability theory, it is generally assumed that uncertainties occur due to randomness (Bellman and Zadeh 1970). In construction industry, however, there are other types of uncertainty (e.g. vagueness or imprecision, lack of information, etc.) which do not lend themselves to the assumption of randomness. These types of uncertainties are usually found in the statements used in estimating activity durations by experts. A more appropriate technique that can handle vagueness or fuzziness is therefore needed.

The size and complexity of today's construction projects have made computers increasingly popular tools to facilitate scheduling (Tavakoli and Riachi 1990). This has given rise to the development of numerous project management software systems. The majority of these systems, however, perform resource allocation in an automatic mode, utilizing a set of heuristic rules. The user of such systems is often excluded from the resource allocation process. As a result, the knowledge pertaining to the project environment is not fully utilized. Therefore, schedules generated using these systems may prove inadequate and far from practical solutions.

1.3 Desirable Features

As mentioned earlier, optimization techniques for resource allocation are feasible only for a small scheduling problem that assumes idealized conditions such as a small number of activities, few types of resources, and definitive temporal information. The size of a construction project, the multiplicity of project goals and constraints, and the need to integrate human expertise into scheduling have made heuristic-based solutions more attractive. In addition, an adequate resource allocation model should be able to handle the types of uncertainties that are inherent in most construction environments. Fuzzy set theory is developed as a tool to handle uncertainties that are nonstatistical in nature (Zadeh 1965). The theory is designed particularly to model imprecise linguistic terms such as "approximately", "about", or "long", etc. normally found in human expert statements (Zadeh 1978, Dubois and Prade 1988, Zimmermann 1991, and Chen and Hwang 1992). This has made the theory intuitively attractive to be used for modeling uncertainties in the resource allocation process.

To improve the quality and workability of generated schedules, it is necessary that project specific knowledge be utilized in the scheduling process. Developing another purely number-crunching computer system would not lead to a proper solution. Instead, a system that allows the user to actively participate in the scheduling process appears to be more of a promising solution. Such a system should encourage and integrate efforts between the user and the computer.

1.4 Research Objectives

The objectives of this study are:

1) To develop a new resource allocation technique capable of resolving multiple

resource conflicts considering multiple objectives.

2) To compare the performance of the newly developed technique with the most commonly used heuristic rules.

3) To study the suitability of fuzzy set theory for modeling uncertainties in the project-network scheduling, and possibly to extend its application to resource-constrained scheduling.

4) To demonstrate the advantages and effectiveness of the newly developed technique against the probabilistic scheduling methods.

5) To develop a prototype decision support system (DSS) to assist the project team in allocating resources to project activities utilizing the newly developed techniques. The capabilities of the developed DSS is demonstrated through the use of two numerical examples.

1.5 Thesis Organization

The remainder of this thesis is organized into the following chapters:

Chapter 2 presents a literature review of previous work related to three areas of construction scheduling: 1) the techniques used to resolve resource constraints, 2) the modeling of uncertainty, and 3) the capabilities of computerized planning and scheduling systems.

Chapter 3 starts by identifying the limitations of traditional heuristic-based scheduling methods. Next, it provides a detailed description of the proposed resource allocation technique. A test conducted to study the performance of the proposed technique against a number of widely used heuristic rules is then presented, and the results of the test are discussed.

Chapter 4 first discusses the fundamental concepts and the arithmetic operations used in fuzzy set theory. Following this discussion, various degrees of temporal imprecision and their associated membership functions are described. A method for unconstrained, fuzzy network analysis called FNET is then introduced. The last section of this chapter describes the extension of FNET to resource-constrained scheduling domain.

Chapter 5 presents a thorough comparison between the results obtained using FNET and those generated by PERT and Monte Carlo simulation. The advantages and disadvantages of the three methods are then described considering the following aspects: 1) the suitability of the method, 2) the assumptions made, and 3) the scheduling information obtained upon the completion of the analysis.

Chapter 6 is devoted to the development of a prototype decision support system. The chapter starts with the overall description of the system together with its components and their respective functions. The capabilities of the proposed system are then illustrated through the use of two example applications. The performance of the prototype is discussed based on the results obtained.

In chapter 7, the results of the study are summarized and the main contributions are stated. The chapter also includes some recommendations for future work.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

Scheduling problems represent a class to which a considerable amount of research effort has been devoted. The reason for this interest is that this class encompasses many and very different problems, with respect to characteristics, application fields and solution techniques. The objective of this chapter is to present a review of the literature related to project scheduling in general, and to construction scheduling in particular. Issues that are frequently addressed in the scheduling literature can mainly be grouped into four main areas: 1) unconstrained network scheduling, 2) constrained network scheduling, 3) uncertainty modeling, and 4) computerized scheduling systems. Each of these areas is discussed in turn in the following sections.

2.2 Unconstrained Network Scheduling

In the unconstrained network scheduling, the only type of constraints that is considered is the precedence or technological constraints. In this regard, two lines of research: deterministic and nondeterministic scheduling will be discussed.

2.2.1 Deterministic Scheduling

In the deterministic type problem, the duration of project activities are assumed to be precisely known. The simplest form of scheduling in this environment is the

Gantt chart (barchart). A barchart consists of a list of activities placed on a horizontal time scale. Each activity is graphically represented by a bar where its ends indicate respectively, the start and finish times, and the length of the bar represents the duration of the corresponding activity. Barcharts are very popular in practice. They are simple and easy to understand by all levels of management, and can be revised in a timely manner. Traditional barcharts have been criticized for the lack of information related to the logical relationships among activities (Barrie and Paulson 1984). Although a number of enhanced barcharts have been proposed to overcome such a limitation (Busch 1991), still, they become very cumbersome as the number of activities and relationships increases. In practice, barcharts are simply used for presenting the outputs of a more sophisticated scheduling model for the purpose of project monitoring and control.

A more refined model for deterministic project scheduling is to formulate activities, their durations, and relationships in the form of networks. The technique such as the Critical Path Method (CPM) (Kelley and Walker 1959; Harris 1978; and Busch 1991) has been used to determine, for example, the critical paths, the slack time for noncritical activities, and the start and finish times for each activity. In CPM, project activities are modeled using techniques such as Activity-on-Arrow (AOA) or Activity-on-Node (AON), alternatively known as the Precedence Diagraming Method (PDM). There are, however, some limitations associated with AOA. For example, it allows for only one type of relationships (finish-to-start) between two activities. In addition to the actual project activities, dummy activities are often required to establish the correct logical relationships, increasing the size of the network. Each activity in AOA is identified by two numbers (the head and tail of the arrow), thereby rendering the construction of the model complicated. PDM was introduced in 1961 to overcome the shortcomings associated with AOA (Harris 1978). In this method, several types of relationships (e.g. finish-to-start, start-to-start, start-to-finish, etc.) between project activities are allowed. Only one identification number is required for an activity. This helps reducing the size of project networks and eliminating the need for dummy activities.

Both AOA and PDM are readily available in most commercial project management software packages. These two techniques, however, has been reported to be ineffective for scheduling repetitive projects such as housing and highway construction (Selinger 1980; Rahbar and Rowing 1992). The main concern is that they do not take into consideration the effective use of resources. To circumvent this limitation, the line-of-balance (LOB) technique has alternatively been used (Lumsden 1968, Arditi and Albulak 1986). The fundamental principle of the LOB technique is to maintain the work continuity, thereby maximizing resource utilization. This technique was originally designed for the manufacturing industry where the domain activities are highly repetitive in nature. Most construction projects, however, contain both repetitive and nonrepetitive activities, rendering the LOB technique in its original form inadequate (Al Sarraj 1991). To enable the scheduling of the project that contains repetitive and nonrepetitive activities, a scheduling system that integrates LOB with network techniques has been developed (Russell and Wang 1993).

The scheduling techniques including CPM, PDM and LOB discussed thus far have been developed for deterministic-type problems. These techniques assume that all the input parameters such as activity's duration are definitive. As such, they may not be adequate scheduling in the situations where the durations of project activities cannot be precisely estimated. Considerable research efforts have been made during the past few decades to overcome such a drawback. Previous studies have employed primarily the probability and fuzzy set theories to model uncertainties associated with temporal variables in the project network. The following subsections describe the developments along these two lines of research.

2.2.2 Probabilistic Network Scheduling

In practice, construction operations are susceptible to many factors such as technologies, work content, equipment, labor, working conditions, weather, efficiency of management, etc. (Baxendale 1984, Ahuja and Nandakumar 1986, Wu and Hadipriono 1994). These factors are the major sources of uncertainties inherent in the construction environment. During the past few decades, probability has been a dominant theory for modeling uncertainty in the scheduling process. The theory has been employed primarily to: 1) estimate activity durations, and 2) analyze project networks.

The main objective in representing the uncertainty in activity duration using probability theory is to construct its probability density function. To accomplish that, four main steps are theoretically required. These steps involve: 1) collecting past performance of similar activities, 2) identifying an appropriate distribution, 3) solving for the selected distribution's parameters, and 4) testing for the goodness of the fit (AbouRizk and Halpin 1990, AbouRizk and Halpin 1992, Badiru 1993, AbouRizk et al. 1994). Various elementary techniques exist for identifying appropriate distributions. The simplest technique is to plot the frequency histogram and select the suitable distribution based on visual examination. There is, however, a major shortcoming associated with this technique. That is, a number of histograms can be produced from the same set of data, by varying the number of cells and its width (AbouRizk and Halpin 1990).

A more objective approach is to use Sturges' rule to calculate the number of cells and its width from the sample size, the highest and lowest observations (AbouRizk and Halpin 1990). Another means to select the appropriate distribution is to calculate the coefficient of skewness and kurtosis of the sample data and compare these two coefficients with those of known distributions. AbouRizk and Halpin (1992) apply this technique to a set of construction activity duration data involving earth-moving operations. They report that beta distributions generally fit the durations better than such distributions as lognormal, uniform, and triangular.

Once the appropriate distribution is selected, the next step is to compute its distribution's parameters. For the beta distribution, various techniques such as moment matching, maximum likelihood, least square, etc., have been used to calculate the lowest/highest and the two shape parameters (i.e. **a** and **b**) (AbouRizk et al. 1994). After these parameters are estimated, the next step is to assess for the goodness of fit. The main idea is to compare the fitted distribution with the empirical one. Techniques such as the chi-square, Komogorov-Smirnov, or visual assessment have been used (AbouRizk and Halpin 1990).

The techniques described earlier are useful only when historical data are available. In the situation where sample durations are not available, some of those statistical parameters would have to be estimated. AbouRizk et al. (1991) develop a computer program to assist the estimator to fit beta distributions to the subjective activity durations. Inputs to their system are the end points of the target distribution, and two of the following parameters: the mode, the mean, the variance, or selected percentile. The outputs are the shape parameters, mode, mean, standard deviation, variance, and percentile points. These probabilistic density functions are then used as input to probabilistic network analysis methods.

The other application of probability theory involves network calculations.

Probabilistic scheduling methods such as the Program Evaluation Review Technique (PERT) and Monte Carlo Simulation have traditionally been used to model uncertainty in network scheduling (Elmaghraby 1977, Hendrickson and Au 1989. Diaz and Hadipriono 1993). These two techniques consider activity durations in the network as random variables, expressed in the form of statistical parameters of probability density functions (i.e. means and variances). If historical data are available, the probability distribution functions can be determined using the techniques described previously. In the situation where the data are not available, the scheduler is required to provide three estimated durations namely: optimistic, pessimistic and most likely as inputs to PERT.

The Beta distribution can generally be used to describe the behavior of those data because it has finite end points and can assume a variety of shapes based on different shape parameters (Elmaghraby 1977, Badiru 1993). The precise calculations for the mean and variance of the beta distribution are quite complicated. In practice, an approximation of the expressions for these two parameters are often used instead to simplify the calculations (Wiest and Levy 1977, Badiru 1993). By applying the central limit theorem, the mean and variance of the project duration can be computed from the summation of their respective values of the critical activities. Disregarding the distributions assumed for activity durations, the project duration calculated based on the central limit theorem can approximately have a normal distribution (Badiru 1993).

A major limitation associated with PERT can be found in the situation where the project network exhibits multiple critical or nearly critical paths (Hendrickson and Au 1989, Diaz and Hadipriono 1993). When this happens, PERT produces optimistic results. A number of research effort has been made to overcome this limitation. Ang et al. (1975) developed a probabilistic evaluation technique (PNET) for predicting the completion times of a project. In their method, several paths that are highly correlated are represented by the longest path in the group. The coefficient of correlation between any two paths is the function of the number of shared activities between the two. A subjective threshold value must be established for determining whether the two paths are correlated. Those paths that are found to be correlated are represented by the most critical path in that group. Once all the major paths and their respective project durations are identified, the probability of having the project completed in less or greater than a certain time is the product of those probabilities of the major paths. The results obtained with PNET were found to agree more with those obtained with Monte Carlo Simulation than those with PERT (Diaz and Hadipriono 1993).

In PNET, defining what should be the threshold value for "high correlation" is subjective and left to the scheduler. Different threshold values can significantly alter the outcome probabilities. Gong (1993) developed a more objective technique called the "back-forward uncertainty estimation" (BFUE) to overcome the limitation associated with PNET. The backward pass is performed in order to locate the relative start events, accounting for the correlation influence. He proposed a method which can be used to calculate the expected time and variance of the joint events. The results obtained using the BFUE were reported to be close to those produced by PNET and Monte Carlo simulation (Gong and Hugsted 1993).

Both PNET and BFUE are direct solutions which, to certain degrees of accuracy, can account for the correlations among different network paths. In the case where project activities exhibit some sort of dependencies among each other, Monte Carlo simulation appears to be the most suitable technique to model such correlations. This technique takes as input the expected mean, variances, types of distributions, and correlations for each scheduling variables. In each simulation, the realizations (i.e. synthetic outcome) of each scheduling variable are generated according to the provided distribution and correlations (Hendrickson and Au 1989). The mean and standard deviation of the project duration in each simulation is calculated using the same procedure as employed in PERT.

As far as the literature is concerned, there has been very limited practical applications of Monte Carlo simulation in construction scneduling. Those reported, have experienced some difficulties. For example, Baxendale (1984) uses Monte Carlo simulation technique to forecast the time required to perform a concrete operation based on the data of the same operation collected on the same site. The forecast obtained from the simulation was reported to be 25% longer than the actual time. He points out that the discrepancy could, perhaps, be attributed to the insufficient data, and the correlations between the activities involved in that operation.

2.2.3 Fuzzy Network Scheduling

Another line of research employs fuzzy set theory to model uncertainties in project network scheduling. Researchers in this area share a common view that uncertainties inherent in the construction environment can sometimes be best described qualitatively using linguistic terms such as "good" or "bad" (Ayyub and Haldar 1984, AbouRizk and Sawhney 1993, Wu and Hadipriono, 1994). These linguistic terms are subjectively quantified and used. It is believed that these types of data lend themselves well to the analysis using fuzzy set theory (Zadeh 1965, Zadeh 1978). Similar to the previous line of research, fuzzy set theory has also been used in the process of estimating activity durations and in performing network calculations.

With regard to the estimation of activity durations, fuzzy set operations have been used to model the relationships between those durations and the factors such as site condition, weather, labor performance, etc. (Ayyub and Haldar 1984, AbouRizk and Sawhney 1993, Wu and Hadipriono, 1994). For example, Ayyub and Haldar (1984) employ fuzzy set theory to model the combined effect of adverse weather conditions and labor experience to the durations of project activities. In their method, fuzzy relation was used to establish two relationships: between the frequency of occurrence and its consequences, and between the consequences and the durations. Fuzzy relation and compositions are then used to establish the crossed-relation between the frequency of occurrence and the activity durations. The output of this method is a set of activity durations and their membership values.

The method proposed by Ayyub and Haldar (1984) has been adopted in the development of a computer system called "SIDES" (AbouRizk and Sawhney 1993). In SIDES, the input data include the factors that affect the duration of the activity being considered, their likelihood of occurrence and adverse consequences on this activity. The ouputs of the system are the mean, variance, and the shape parameters of the beta distribution from subjective data provided by the users. Wu and Hadipriono (1994), alternatively, use a technique called the fuzzy modus ponens deduction (FMPD) to assess the impacts of duration factors on activity durations. Given the most likely activity durations, it calculates the optimistic and pessimistic durations as required in PERT. A concept called the angular fuzzy set is used to interpolate the resulting fuzzy membership function between the available information and the expected situation (Hadipriono and Sun 1990). The calculated activity durations are then used in probabilistic scheduling techniques such as PERT.

In addition to the above application, fuzzy set theory has also been employed to represent the uncertainties network calculations. Previous studies have demonstrated the feasibility in using fuzzy arithmetic operations to calculate the early start and finish times along with fuzzy project durations (Chanas and Kamburowski 1981, Dubois and Prade 1988, McCahon 1987). Due to a major limitation associated with the traditional fuzzy subtraction, backward pass calculations cannot be performed in a similar manner as that in CPM. To circumvent this limitation, McCahon (1993) use the mode and the generalized mean value of fuzzy variable in the forward and backward pass calculations. Her technique can be used to provide approximate results for fuzzy calculations. A method that realistically propagates uncertainties in the backward pass calculation is not available.

2.3 Resource-Constrained Scheduling

The unconstrained network scheduling techniques described in the previous sections essentially concern themselves only with precedence or technological constraints. Resources required to perform project activities are, however, assumed to be available when they are needed. In practice, however, project teams have often been confronted also with resource constraints (Thamhain and Wilemon 1986). To produce a realistic schedule, it is necessary that resources are incorporated into scheduling. Failure to do so can cause, for example, unexpected project delays which are sometimes unrecoverable. An attempt to bring the project back on track often requires the use of overtime, resulting in potential project cost overruns. Successful resource-constrained scheduling can provide not only realistic schedules but also can serve as an early warning to the project team so that necessary precautions can be made at the outset. Research in the area of resource-constrained scheduling can be classified into two groups: deterministic and nondeterministic type problems. The following subsections summarize the review of the research in these two areas.

2.3.1 Deterministic Resource-Constrained Scheduling

The term resource-constrained scheduling and resource allocation are often used interchangeably in literature (Hasting 1972, Chang et al. 1990, Johnson 1992, Badiru 1993). In this study, both are used to describe the process of scheduling with resource constraints while minimizing the unavoidable project delays. Deterministic resource-constrained scheduling have been approached using both optimization and heuristic methods. The optimization methods aim at providing the best (i.e. optimum) solutions using the well known optimization techniques such as integer programming (Brand et al. 1964, Pritsker et al. 1969, Talbot and Patterson 1979, Khashenas and Haber 1990) and various enumeration procedures (Davis and Heidorn 1971, Stinson et al. 1978, Talbot and Patterson 1978, Patterson 1984, and Christofides et al. 1987). These traditional optimization techniques have constantly been criticized for their excessive computational requirements to handle most large-scale problems (Weist and Levy 1977, Kurtulus and Davis 1982, Allam

1988, Hadavi et al. 1992, and Dumond 1992).

Linear and integer programming are used in the past to minimize project delays (Brand et al. 1964, Talbot and Patterson 1979, Teixeira and Bezelga 1990). Linear and integer programming involve minimizing or maximizing an objective function while satisfying a set of project constraints. The most common objective is the determination of a minimum duration schedule for the projects subject to the predetermined precedence constraints for the activities and the resource availability (Lee et al. 1978). Alternatively one can form the objective function in terms of project cost. For example, Khashenas and Harber (1990) used binary variables in formulating a linear integer model for minimizing project costs. All resources including times are transformed into cost components. Limitations of resources are imposed by assigning extremely high costs for the use of resources above desirable limits. A four-activities network example was used to illustrate their model. The authors claim that the schedule obtained using their model would have an optimal duration and the resource use is leveled "economically". Similar approach was used in the model proposed by Teixeira and Bezelga (1990).

Early attempts in applying integer programming in order to obtain optimal solutions, were reported to be unsuccessful (Davis 1973, Patterson 1984). Researchers have alternatively approached this problem using different search techniques. The examples include bound enumeration (Davis and Heidorn 1971), branch and bound (Hasting 1972, Stinson et al. 1978), and implicit enumeration (Talbot and Patterson 1979). These techniques basically generate partial solutions (solution trees) and search for the shortest path. The main differences among the three techniques are the way in which the solution trees are generated, and the manner in which inferior solutions are eliminated (Patterson 1984). Enumeration techniques can solve relatively larger size problems when compared to integer programming. In the situation where degrees of resource tightness are high, however, there is no guarantee that the optimal solution will be obtained (Patterson 1984).

Optimization techniques have constantly been criticized for their excessive computational requirements to solve real-life problems because of the many possible scheduling combinations (Adrian 1973; Kurtulus and Davis 1982; Allam 1988; Johnson 1992; Alkayyali etal. 1994). In an effort to overcome this limitation, a number of heuristic network-based methods have been introduced. Heuristic methods do not guarantee the optimal solution, but produce good solutions to practical problems. Over the past few decades, several heuristic models have been developed. These models assign priorities to the activities that are experiencing resource conflicts based on certain rules. Panwalker and Iskander (1977), and Lawrance (1985) provide examples of the heuristic rules that are used in various scheduling domains. The existence of numerous rules has prompted researchers to study the performance of those rules.

The performance of heuristic rules has been the source of interest to many researchers ever since the early 1960s (Brand et al. 1964, Davis and Patterson 1975, Lawrence 1985, Allam 1988, Mohanty and Siddiq 1989). Most heuristic models are CPM-based. Their operations range from simple dispatching rules to complicated resource allocation algorithms. The simplest heuristic models utilize scheduling parameters such as the total float (slack), the early/late start/finish times, the activity duration, etc. to assign priorities to the activities experiencing resource conflicts. For example the minimum slack rule (MINSLK) gives higher priorities to the activities that have smaller slack time (more critical activities). A more sophisticated method such as the Resource Scheduling method (RSM) developed at the university of Illinois (Brand et al. 1964) resolves resource conflicts by examining in a consecutive fashion, a pair of conflicting activities at a time. Necessary changes in the sequence of the conflicting activities are made until the resource constraints are satisfied.

The main objective of applying heuristic models is to minimize the possible project delays caused by resource conflicts. The performance of the widely used heuristic rules was examined and reported in many papers. Davis and Patterson (1975), for example, compare the number of optimal solutions generated using eight different heuristic rules. They found that the first three best performance rules are: the MINSLK, the minimum late finish (MINLFT), and the RSM respectively. The superiority of the MINSLK rule has been reconfirmed by the study done by Allam

(1988).

To effectively use the MINSLK rule, it is necessary that the total floats of the unscheduled activities are continuously updated. For a large-scale project network, this could be a time-consuming task for the scheduler. To avoid such a laborious task, Shanmuganayagum (1990) introduces a new parameter called the "current float". This current float is identical to the updated total float, and can be calculated directly when needed. The scheduling decisions obtained using the current float, however, have been proven to be identical to those obtained using the minimum late start rule (Touran 1991).

Most of the project management software packages that facilitate resource allocation adopt heuristic models to resolving arisen resource conflicts (Suarez 1987, Johnson 1992). Suarez (1987) compares resource allocation performance of four commercially available software packages using a small network. Priorities are established using the default options of each system. The only conclusive finding from this study is that the four packages produce different results for the same network. Johnson (1992) examines resource constrained scheduling capabilities of seven commercial project management software packages. The project completion times produced by each package were recorded and compared to the optimal schedules. He reports that no package consistently produces the minimal project duration. The average percentages increased above the optimal solutions range from 5.03 to 25.6.

With respect to the performance of heuristic models, a common conclusive finding of previous studies has indicated that traditional heuristic models perform with varying degrees of effectiveness when applied to different networks. Therefore, it is not possible to identify, in-advance, the rule(s) that will produce best for the problem at hand. This major shortcoming could be attributed to the fact that resources are allocated to the higher priority activities without examining their consequences. Like most enumeration-based techniques, it will be computationally intractable, however, if all or nearly all possible consequences are examined. A model that provides a reasonable trade-off between the quality of schedules produced and computational efforts seems to be a practical solution approach for this type of problem.

2.3.2 Nondeterministic Resource-Constrained Scheduling

Resource-constrained scheduling methods discussed thus far are applicable to the deterministic-type problems. There have been relatively few efforts to incorporate uncertainty into resource-constrained scheduling, as far as the literature is concerned. In the area of construction scheduling, the work of Ahuja and Arunachalam (1984) is perhaps one of the earliest attempts to model the uncertainty associated with resource availabilities. Their model is capable of generating several scheduling alternatives having varying project duration, cost, and performance probability. The period of resource availability is considered to be uncertain. Resources that are selected from more certain sources are first allocated to project activities. The model is interfaced with a project management software package, utilizing the incorporated heuristic rules to resolve any conflicts. A major limitation associated with their model is that only resource availabilities are assumed to be uncertain, the durations of project activities are, however, assumed to be deterministic.

Padilla and Carr (1990) developed a Monte Carlo simulation-based program called the "DYNASTRAT" for resource allocation. In DYNASTRAT, activity durations and costs are assumed to be random variables and are modeled using a beta distribution. They proposed a formula incorporating 15 scheduling parameters for calculating the priority for project activities. Some of these parameters are: activity priority points, criticality factor, criticality priority, global resource priorities, resource priority, etc. The values of all fifteen parameters are to be input by the user.

There are major problems associated with resource-constrained scheduling using Monte Carlo simulation. First, while it is true that the schedule generated in each simulation is a resource conflict-free one, it is not necessarily true for the final schedule (i.e. compiled from all simulations). This is due mainly to the changing network-logic after resource allocation in each run. The results of applying certain heuristic rules (e.g. the shortest activity duration, the minimum slack rule, etc.) in each simulation could be different, given the same set of conflicting activities. The results obtained using this technique are useful for assessing global project risk. They are, however inadequate for implementation at the activity level. A more practical solution is therefore needed.

There has been an attempt to apply fuzzy logic to the process of resource allocation for construction projects (Chang et al. 1990). In their study, the impacts of external factors such as precipitation are incorporated in the activity priority establishment process. An expert system called "PRIORITY RANKING" has been developed at the University of California at Berkeley. The system uses fuzzy logic and possibility theory which are extensions of fuzzy set theory to draw conclusions related to the importance for each criteria and the succeptibility of an activity to certain criteria (Chang and Ibbs 1990). For example, suppose that a criterion being severity of rain has been established, and an activity placing concrete in an open area is to be considered. Assume also that it rains a lot in the site. The priority of this activity as calculated using this method would be low. The use of the system is, however, limited to a single-resource-type project. In addition, resource availabilities as well as activity durations are deterministic.

2.4 Computerized Planning and Scheduling Systems

The advent of inexpensive microcomputers over the last decade has significantly played an important role in the current project management practice, as well as the research directions. Among various project management functions, scheduling is considered to be one of the very first to be computerized. Today, there are numerous commercial scheduling software packages that are readily available in the market (Johnson 1992). The use of these packages, though exhibiting an increasing trend, is still limited (Tavakoli and Riachi 1990). Most of the software tools are, however, domain independent. They are not specifically designed to accommodate the unique characteristics of construction projects. As such, they have often been criticized for their vast demands for inputs and fail to incorporate experiential knowledge and rich environments for scheduling in this domain.

In view of these deficiencies, there have been two primary categories of research: those attempting to automate network generation (i.e. planning) and those dealing with scheduling. With respect to the first category, the majority of construction planning research has utilized the advanced computer technology called artificial intelligence (AI) to identify project activities and their relationships. Most AI planning systems start generating project networks by first breaking down a project into basic components such as beams, columns, slabs, walls, etc using construction knowledge stored in knowledge bases, or extracting them from the 3-D graphic data of the facility. The basic activities necessary to assemble these components are usually stored in the system's knowledge bases. Technological relationships among these activities are either predetermined such as those used in CONSTRUCTION PLANEX (Hendrickson et al. 1987) and ESCHEDULER (Moselhi and Nicholas 1990), or established by physical relationships between the components such as "covered by", "supported by", "set on", or "side by side" used in GHOST (Navichandra et al. 1988), OARPLAN (Darwiche et al. 1988), and KNOW-PLAN (Morad and Beliveau 1991).

Once a plan is developed, some systems continue to generate a schedule using the conventional techniques such as CPM and PDM (Zozoya 1988; and Nicholas 1990) or the combinations of LOB and CPM such as the CONSCHED system developed by Shaked and Warszawski (1992). Schedules generated by these systems can serve as a preliminary schedule since they do not satisfy other types of scheduling constraints such as those involving resources and times.

The other category of research deals with various functions in scheduling such as estimating activity durations, time-cost tradeoff, and resource scheduling. PLATFORM is perhaps one of the first AI-based prototypes developed by Levitt and Kunz (1985). It combines the benefits of AI techniques including the frame-based and rule-based knowledge representation for encoding and reasoning the planner's knowledge about risk factors (e.g. types of soil, etc.) that might have both positive and negative impacts on activity durations. The system is used for updating activity networks for the construction of offshore drilling platforms project. The system, however, does not incorporate resources into scheduling. JANUS is another example of AI-based system developed by Axworthy and Levitt (1989). The system utilizes object-oriented approach for representing scheduling entities as well as the knowledge pertaining to project acceleration. It does not, however, address a number of important issues related to resources.

2.7 Summary

This chapter has reviewed previous work related to various aspects in project scheduling. In general, it can be stated that most of the existing procedures provide solutions that satisfy certain aspects of construction scheduling problems. In respect to the resource-constrained scheduling, for example, it has generally been found that heuristic approaches are more appropriate for practical problems than the optimization ones. Traditional heuristic rules, however, exhibit various degree of inconsistencies in their performance. The technique that combines the benefits of the two approaches is needed.

With regard to uncertainty modeling, the traditional probabilistic networking methods require excessive efforts for acquiring historical data. The complexity of probability theory (i.e. selecting the probability density functions, calculating the distribution parameters, testing for the goodness of fit, incorporating correlations between variables, etc.) has been a major barrier for the theory to be fully adopted in practice. More importantly, there are other types of uncertainty such as those inherent in expert's judgment which do not lend themselves for the analysis using probability theory. Fuzzy set theory appears to be an attractive solution approach in this regard. Existing scheduling systems both commercially available and research prototypes are designed to behave like a "black box". The users of those systems are often excluded from the scheduling process. To produce or maintain realistic schedules, it is necessary that the user be part of the decision making processes. The system that supports joint performance between the user and computer is, therefore, intuitively appealing.

CHAPTER 3

PROPOSED RESOURCE ALLOCATION MODEL

3.1 Introduction

Ever since the early 1960s, resource-constrained scheduling has challenged many practitioners and researchers in both manufacturing and construction industries. Due to the unsuccessful application of the optimization-based techniques as described in Chapter 2, considerable research effort has been directed toward the use of heuristic rules to resolve resource conflicts that may arise among project activities. The main objective of applying these rules is to minimize any unavoidable project delays that may occur beyond the unconstrained CPM project duration. Today, there are numerous heuristic rules developed and used by both researchers and practitioners (Panwalkar and Iskander 1977, and Lawrence 1985). The performance of those heuristic rules has been a source of interest to researchers for the past few decades. The following section summarizes some of those findings.

3.2 Performance of Traditional Heuristic Rules

Davis and Patterson (1975) studied the performance of eight commonly used heuristic rules. These rules include: the minimum slack (MINSLK), the resource scheduling method (RSM), the minimum late finish (MINLFT), the greatest resource demand, the greatest resource utilization, the shortest duration, and the random rules. A comparison was made between the results generated by these rules and the optimal solutions. According to the results of their studies, the three best performance rules were the MINSLK, the MINLFT, and the RSM. Table 3.1 shows the relative performance among the three rules.

Davis and Patterson (1975) measured the performance of these rules by comparing the frequency of obtaining the shortest project duration, and the percent increase above the respective optimal solution generated in each case. It can be seen from Table 3.1 that the MINSLK outperforms the MINLFT, while the RSM is the worst among the three in both measures. The poor performance was observed from the number of longest project durations obtained using each rule. The higher the number of longest projects obtained using any rule, the poorer the performance of that rule is. Considering for example the total number of the longest project durations, the worst performance can be attributed to the MINLFT rule, while the MINSLK and the RSM were equally good. Regarding the number of exclusive longest project durations, however, the RSM is the worst, while the MINLFT rule performed best. The findings of Davis and Patterson (see Table 3.1) are apparently inconclusive in this regard. In general, however, it can be stated that the MINSLK generally performs best, while the RSM is the worst among the three rules.

Table 3.1: Relative Performance of traditional Heuristic Rules*.

Summary Measures	Heuristic Scheduling Rule		
	MINSLK	MINLFT	RSM
Average percent increase above optimum	5.6	6.7	6.8
Number of problems with optimum solutions	24	17	12
No. of times shortest heuristic-based duration produced	50	38	28
No. of times unique shortest duration produced	15	5	5
No. of times longest duration produced	2	4	2
No. of times unique longest duration produced	1	0	2

* adopted from Davis and Patterson (1975)

3.2.1 Similarity between the RSM and MINLFT rules

The similarity between the solutions generated by the RSM and those of the MINLFT rule can be observed from the results of the study performed by Davis and Patterson (1975). An attempt is made in this study to prove this similarity. In the RSM, resource conflicts are resolved by examining, on a consecutive fashion, a pair of conflicting activities at a time. Then, necessary changes in the sequence of the two activities are made to satisfy all resource constraints (Brand et al. 1964). The comparison between the activities within a pair is made based on the calculation of the increase in project duration resulting when activity **k** follows activity **j** (D_{jk}).

$$D_{jk} = \text{Max}\ [0\ ; (EFT_j - LST_k)] \qquad (3.1)$$

or activity j follows activity k (D_{kj}).

$$D_{kj} = \text{Max}\ [0\ ; (EFT_k - LST_j)] \qquad (3.2)$$

For the RSM to be applicable, at least one of the above equations must produce a positive value. In the situation where both equations produce positive values, the smallest **D**, by definition, is the one that will establish the priorities for allocating resources. In order to prove the similarity between the RSM and the MINLFT rules, the two equations (Eqs. 3.1 and 3.2) forming the basis for establishing the priorities in the RSM will be expressed in terms of the late finish time (LFT).

For the valid case (i.e. $D_{jk} > 0$), the expression **EFT - LST** in Eqs. 3.1 and 3.2 must be greater than zero. That is $EFT_j > LST_k$.

Suppose that the final decision is that activity **j** precedes activity **k**. That is, D_{jk} is smaller than D_{kj}, or:

$$D_{jk} < D_{kj} \tag{3.3}$$

Substituting Eqs. 3.1 and 3.2 into the above inequality, and considering only the expression **EFT - LST**, yields:

$$EFT_j - LST_k < EFT_k - LST_j \tag{3.4}$$

From the CPM analysis:

$$EFT = EST + d \tag{3.5}$$

$$LST = LFT - d \tag{3.6}$$

where **EST** and **EFT** are the early start and the early finish time, **LST** and **LFT** are the late start and the finish time, and **d** is the duration of the activity in question. Substituting Eqs. 3.5 and 3.6 into the inequality 3.4 yields:

$$(EST_j + d_j) - (LFT_k - d_k) < (EST_k + d_k) - (LFT_j - d_j) \tag{3.7}$$

which could be expressed as:

$$(EST_j + d_j - LFT_k + d_k) < (EST_k + d_k - LFT_j + d_j) \tag{3.8}$$

The durations of both activities are positive values. Inequality 3.8, therefore becomes:

$$\mathbf{EST_j - LFT_k < EST_k - LFT_j} \tag{3.9}$$

Considering the expression $\mathbf{EST_k - LFT_j}$ which appears on the right-hand-side of the above inequality, the outcome of this expression must be greater than zero. Therefore,

$$\mathbf{EST_k > LFT_j} \tag{3.10}$$

From CPM analysis, $\mathbf{LFT_k}$ is greater than $\mathbf{EST_k}$. From inequality 3.10, it can be stated that $\mathbf{LFT_k > LFT_j}$. In other words, if activity **j** is to precede activity **k**, the **LFT** of activity **j** must be smaller than that of activity **k**.

Therefore, it can be stated that the scheduling decisions obtained using the RSM and those of the MINLFT rules are identical. This proof is also valid for the case where an updated early finish time of an activity is to be used instead of the original **EFT** obtained from CPM analysis. This can be demonstrated simply by replacing the term **EFT** in Inequality 3.8 by **CT + d** (current time + activity duration), the following inequality can be obtained:

$$\mathbf{(CT + d_j - LFT_k + d_k) < (CT + d_k - LFT_j + d_j)} \tag{3.11}$$

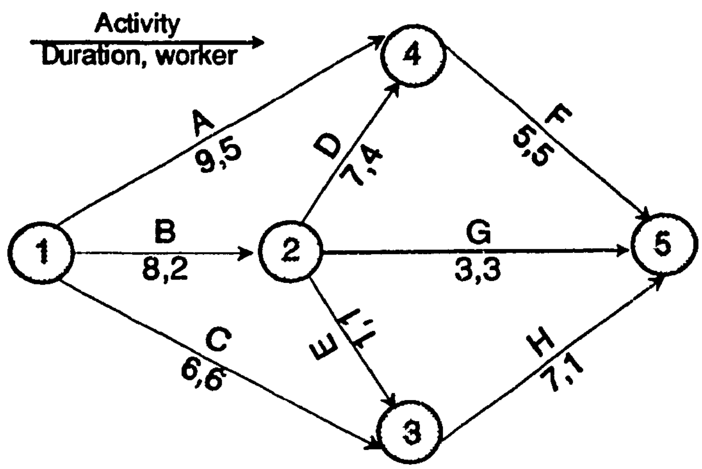
Eliminating the terms that appear on both sides of Inequality 3.11, and multiplying both sides by -1, results:

$$LFT_k > LFT_j \tag{3.12}$$

In order to illustrate the proof of the equality, made above, between the RSM and the MINLFT, a CPM network example used by Adrain (1973) is analyzed. The project network and its CPM information are shown in Fig. 3.1. The RSM and the MINLFT rules were used for resource allocation. The results are summarized in Table 3.2. For the cases that are applicable to the RSM, it can be seen that the decisions obtained from using the two rules are identical. It is clear, however, that the RSM requires more computational efforts than those of the MINLFT rule.

3.2.2 Shortcomings of Current Heuristic Rules

An analysis was made in this study to identify the possible limitations associated with traditional heuristic rules. With the exception to RSM, these rules assign priorities to activities according to pre-established criteria without examining the downstream effects of such allocations on the overall project duration. In certain circumstances, some of the postponed activities could impose even more tightness (diminishing the remaining total float) to the project network. Moreover, resources are allocated to individual project activities sequentially. This prevents the opportunities to investigate the effects of group-allocation which is often the actual situation, disregarding the rules being applied. This, perhaps, could be the main source of the inconsistency inherent in current heuristic-based models as reported



Activities	ES	EF	LS	LF
A	0	9	6	15
B	0	8	0	8
C	0	6	7	13
D	8	15	8	15
E	8	9	12	13
F	15	20	15	20
G	8	11	17	20
H	9	16	13	20

Fig. 3.1: Example Network No. 1.

Table 3.2: Similarity Between the RSM and MINLFT Rule.

CT	Conflicting activities	EFT	LST	LFT	D_{jk}	RSM Decisions	MINLFT Decisions
0	A B C	9 8 6	6 0 7	15 8 13	$D_{AB} = 9$ $D_{BA} = 2$ $D_{BC} = 1$ $D_{CB} = 6$ $D_{AC} = 2$ $D_{CA} = 0$	B -> A B -> C C -> A	E -> C -> A
8	D E G	15 9 11	8 12 17	15 13 20	$D_{DE} = 3$ $D_{ED} = 1$ $D_{EG} = -8$ $D_{GE} = -1$ $D_{DG} = -2$ $D_{GD} = 3$	E -> D E ? G D -> G	E -> D -> G
9	D G H	16 12 16	9 18 14	15 20 20	$D_{DG} = -2$ $D_{GD} = 3$ $D_{GH} = -2$ $D_{HG} = -2$ $D_{DH} = 2$ $D_{HD} = 7$	D -> G G <-> H D -> H	D -> G G <-> H D -> H

X -> Y = Schedule X before Y; X <-> Y = Schedule X or Y first makes no difference;
X ? Y = Inconclusive results

by researchers (Allam 1988, Davis and Patterson 1975).

Although the RSM can be viewed as an improvement to traditional heuristic rules in this regard, it has a major limitation. According to Equations 3.1 and 3.2, the RSM does not provide an explicit solution when both, $\mathbf{D}_{jk}$ and $\mathbf{D}_{kl}$, have negative values. This stems from the fact that the positive value of **D** indicates the increase in the project duration resulting from forcing one activity to follow another. The negative value of **D**, by definition, has no physical meaning. The MINLFT rule, on the other hand does not have such a limitation. In this respect, the RSM can be viewed as a special case of the MINLFT rule.

3.3 Proposed Resource Allocation Model

A new heuristic-based resource allocation model has been developed to overcome the limitations of both optimization techniques as well as those of traditional heuristic rules identified earlier. The model is intended to be used for resolving multiple resource conflicts that may occur in a project schedule. The proposed model is based on a decomposition technique in which the analysis is carried out locally on small segments of the network using an evaluation function. This function measures the negative impact each alternative has on the overall project duration. The proposed resource allocation procedure is called least impact model (LIM). Unlike the conventional heuristic rules where a single scheduling objective (e.g. minimize project delays) is normally considered, LIM resolves resource conflicts taking into consideration: 1) project delays, 2) resource utilization, and 3) resource interruptions. The following subsection describes the problem definition and assumptions made in LIM.

3.3.1 Problem Definition

A set of projects **P** is given, where project **P** consists of a set of activities **I** = {1,2,....**n**} and a set of resources **R** = {1,2,....,**j** }. Each resource **r** $\in$ **R** is a renewable resource (e.g. machines, workers, tools). Each resource can be used by, at most, one activity at a time. The time intervals during which each resource is available are given. For each resource **r** $\in$ **R**, several periods of availability can be defined. This means that between two availability periods the resource **r** is not available, and no activity which requires this resource can be scheduled.

LIM permits activity splitting. The splitting can be done prior to or during the allocation process. For some activities in construction projects, splitting is physically possible only at pre-specified times. As a result, those activities should be divided into a number of sequential sub-activities, according to splitting time intervals specified. The precedence relationships among these sub-activities as well as their calculated scheduling parameters are accordingly generated. Each individual sub-activity is treated in the same fashion as a non-splitting activity which remains to be scheduled. For the activity in which splitting is allowed during the allocation process, the splitting-point can be specified as a percentage of its duration (e.g. must be executed for least 50% of the duration before being interrupted).

The proposed model is different from traditional heuristic rules in two main aspects. First, evaluations are made to groups rather than to an individual activity. Second, the negative impact of each group on the overall project duration is assessed prior to the allocation. Accordingly, LIM is expected to enhance the performance of the outstanding heuristic rules including the MINSLK and the MINLFT rules. The details of these two aspects are described in the following subsections.

3.3.2 Group Formation

In LIM, resources are allocated to a group or a set of activities rather than to individual activities, one at a time, in a sequential manner (Moselhi and Lorterapong 1993a). At any time when a resource conflict is encountered, activities that are involved in the conflict are used to form all possible activity sets. Considering for example, a set $\mathbf{X}_1$ containing 4 conflicting activities (e.g. activities A, B, C, and D) as shown in Fig. 3.2. These activities are used to generate 4 different possible subsets ($\mathbf{X}_{11}$, $\mathbf{X}_{12}$, $\mathbf{X}_{13}$, and $\mathbf{X}_{14}$) each consists of 3 activities as depicted in level 2 of Fig. 3.2. These four new subsets are then checked for their feasibility.

A feasible subset is defined as the subset that can be scheduled without creating

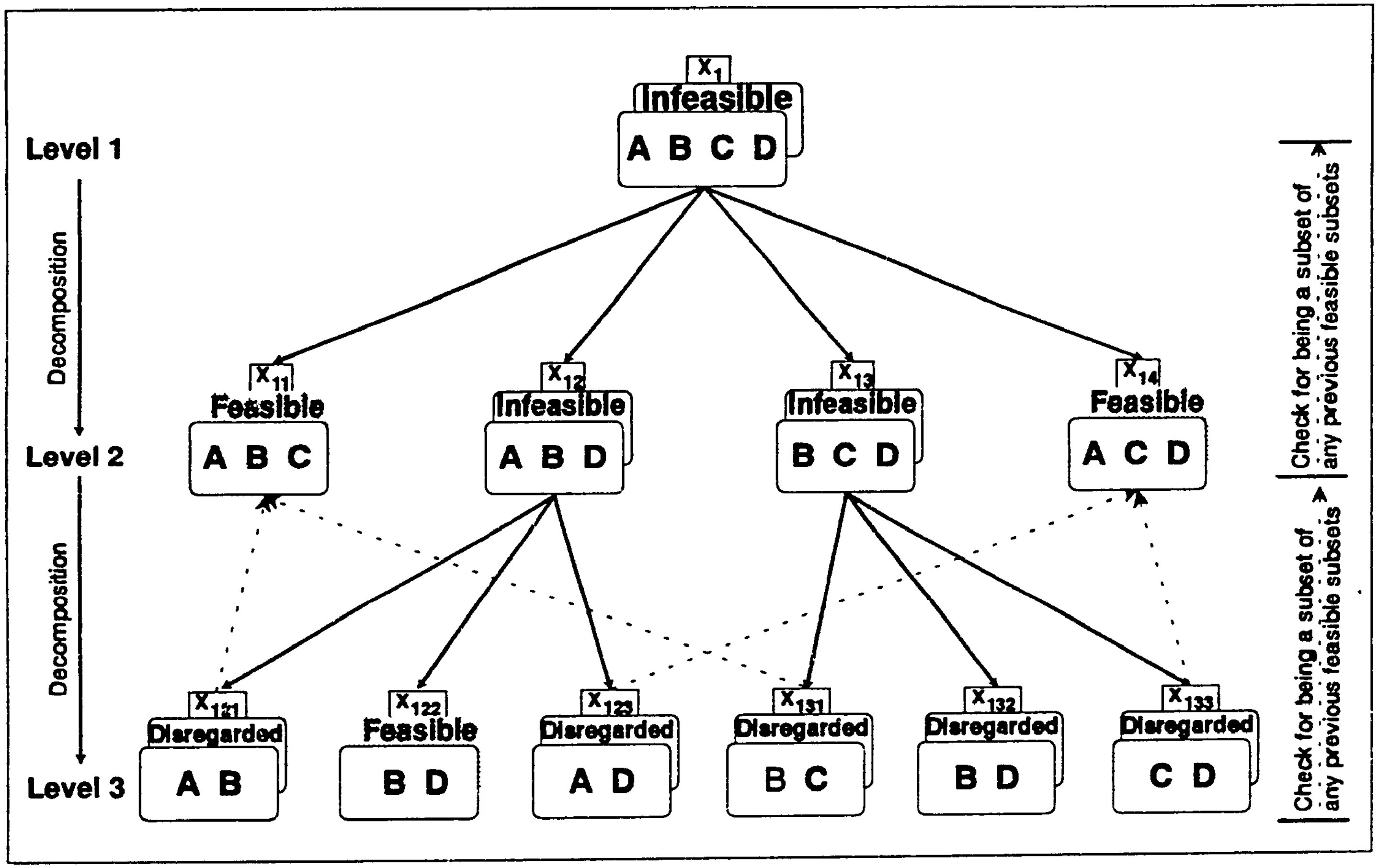


Fig: 3.2 Group Formation.

any resource conflict, given a number of resources available at that moment. Let us assume that the subsets $\mathbf{X}_{11}$ and $\mathbf{X}_{14}$ are feasible and the subsets $\mathbf{X}_{12}$ and $\mathbf{X}_{13}$ are infeasible (as illustrated in Level 2 of Fig. 3.2). For the subsets that are feasible (e.g. the subsets $\mathbf{X}_{11}$ and $\mathbf{X}_{14}$ in level 2 of Fig. 3.2) no further decomposition is needed. For those subsets that are not feasible (e.g. the subsets $\mathbf{X}_{12}$ and $\mathbf{X}_{13}$ in level 2 of Fig. 3.2), the activities in each subset are used to generate smaller subsets such as those in level 3 of Fig. 3.2. These newly generated subsets are disregarded if they constitute parts of the feasible subsets generated in upper levels (e.g. $\mathbf{X}_{121} \subset \mathbf{X}_{11}$, therefore, $\mathbf{X}_{121}$ is dismissed). Similarly, $\mathbf{X}_{132}$, is eliminated as it is the same as $\mathbf{X}_{122}$. The newly generated subsets, in turn, are treated using the same procedure. The main reason for disregarding these smaller feasible subsets is to possibly maximize the use of resources to their full capacities. It is expected that this model will enhance the quality of schedules with regard to resource-based scheduling objectives such as resource utilization and interruptions.

3.3.3 Group Evaluation

The following procedure is used for evaluating the merit of each and every feasible activity subset identified in the previous section. A delay function incorporating three new terms namely the current time (**CT**), the next time frame (**NTF**) and the remaining total float (**RTF**) is used to facilitate this evaluation. The **CT** is the time-frame at which the current examination is being made or the time when a resource conflict is encountered. The **NTF** for each subset is the projected earliest time located after **CT** where a new decision pertaining to resource allocation is required. Figs. 3.3 graphically demonstrate an example for the determination of the **NTF** for subset X_{11}. In this subset activities A,B, and C are assumed to be scheduled, and activity D is assumed to be postponed. Therefore, the **NTF** for this subset would be the finish time of activity B (see Fig. 3.3). Generally there could be more than one activity being in-progress. As such, it is necessary to establish a set of in-progress activities **P** consisting of all activities **p** in which their scheduled finish time is greater than **CT**. Accordingly, the **NTF** can be calculated as follows:

$$NTF = Min[CT + d_1, CT + d_2, .., CT + d_x, SFT_1, SFT_2, .., SFT_p] \qquad (3.13)$$

where **x** is the activity that belongs to a feasible subset **X**; d_x is the duration of activity **x**; $p \in P$; and SFT_p is the scheduled finish time of activity **p**.

The expected project delay resulting from scheduling this subset is determined from the **RTF** of activities excluded from this subset (i.e. activity D in this case). Generally, there could be several activities being excluded from the feasible subset **X** being considered (e.g. subset X_{122} in the third level of Fig.3.2). Let **Z** symbolizes the set of activities being excluded from **X**. The remaining total float of any activity **z** that is excluded from activity subset **X** can be calculated using the following equation:

$$RTF_z = LST_z - NTF_X \qquad (3.14)$$

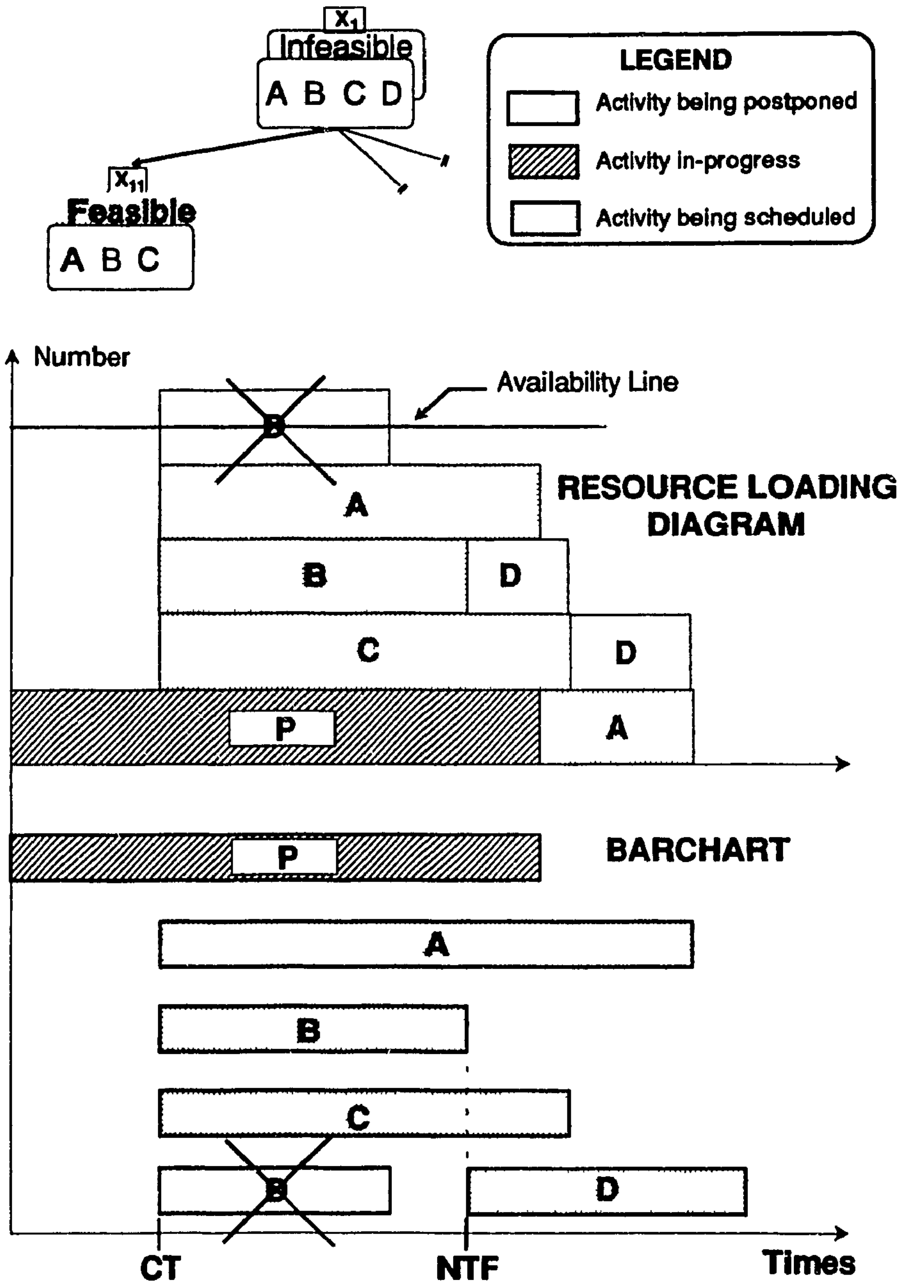


Fig. 3.3: Graphical Presentation of Next Time Frame.

where **z** is the postponed activity that belongs to **Z**; $\mathbf{LST_z}$ is the late start time of activity **z** (from the CPM analysis).

The **RTF** calculated using Eq. 3.14 can have either a positive or a negative value. The negative **RTF** indicates that there will be an extension to the original project duration as a result of postponing the activity in question. The positive **RTF**, on the other hand, suggests that no delay has yet occurred as a result of postponing this particular activity. However, certain amount of available total float will be consumed if the activity in question is postponed. The amount of the total float consumed is equal to the original total float less the calculated **RTF**.

At this stage, one could introduce a weight factor (**w**) which can be used to account for the significance of the activity being postponed. The significance of an activity can be extracted directly from the project network (e.g. the bottle-neck activity, etc.) or it could be subjectively defined by the scheduler. If resource allocation is performed in a multi-project environment, this weight factor could be the priority that the scheduler assigns to each project. Once established, the weight factor can then be used for adjusting the **RTF** values. The adjusted **RTF** is called the negative impact (**nI**) which can be calculated as follows:

$$\mathbf{ni_z = w_z \times RTF_z;} \quad \text{when } RTF_z < 0 \qquad \mathbf{(3.15)}$$

and

$$ni_z = \frac{RTF_z}{w_z}; \quad \text{when } RTF_z \geq 0 \tag{3.16}$$

The overall negative impact generated by scheduling the subset **X** (NI_X) takes it value from the largest negative impact created by the activities that are postponed from the subset as:

$$NI_x = Min[ni_1, ni_2, ..., ni_z] \tag{3.17}$$

The above negative impact is calculated for each and every feasible subset identified earlier. Once the negative impacts for all feasible subsets are computed, the next step is to select the best subset. The most favorable subset is the one that has, numerically, the largest **NI**. For example if the NI_{11} and NI_{14} are calculated to be -9 and -4 respectively, the subset X_{14} is more favorable.

In addition, it is possible also to calculate the expected project delay (**ED**) caused by scheduling a particular subset. The expected delay due to scheduling subset **X** is calculated from the **RTFs** (before adjusting by the weight factor) of its postponing activities as follows:

$$ED_x = Max[|RTF_1|, |RTF_2|, |RTF_3|, ..., |RTF_z|] \tag{3.18}$$

It should be noted however that only those **RTFs** that have a negative value are relevant.

3.3.4 Allocation Procedure

The schedule is generated incrementally by iteratively selecting a set of activities and assigning a scheduled start time and resources to them. The time when a new decision is required (e.g. a resource conflict arise) is referred to as current time. The cycle starts by first initializing the following control parameters:

Set the current time (**CT**) = 0

Set the amount of resources in-use ($\mathbf{RU_j}$) = 0; where **j** is a resource type, and then performing the following 13 steps:

1. Identify activities where their predecessors had finished, select only those that would individually satisfy resource availabilities at that time (see the following condition), and include them in the eligible activity set **E**.

If $\mathbf{r_{ij} < R_j - RU_j}$ then $\mathbf{i \in E}$

where

$\mathbf{R_j}$	=	amount of resource **j** available.
$\mathbf{RU_j}$	=	amount of resource **j** in-use.
E	=	{the set of activity **i** where **i** is eligible to be scheduled at this current time}.

2. If all eligible activities identified in step 1 are to be scheduled simultaneously, check whether there is (are) any resource conflict(s). This can be performed as follows:

$$\sum_{\forall i \in E} r_{ij} < R_j - RU_j \tag{3.19}$$

If the above condition is satisfied for all types of resources (i.e. no resource conflict is encountered), allocate required resources to all eligible activities, assign the current time (**CT**) as their scheduled start time (**SST**), and calculate the next time frame (**NTF**) using Eq. 3.13. Then, identify which activity is in-progress by checking the following condition.

If $SFT_i > NTF$ then activity **i** is in-progress

where

SFT_i = Scheduled finish time of activity **i**.

= $SST_i + d_i$

Update the resource pool, and move the time to the next time frame, and repeat the above procedure starting from step 1.

3. If, however, a resource conflict is encountered, identify the conflicting resources using the following condition.

Resource **j** causes a conflict if:

$$\sum_{i \in E} r_{ij} > R_j - RU_j \tag{3.20}$$

4. Allocate the required resources to those activities that do not require the conflicting resources, and exclude them from the eligible set **E**. This procedure will significantly reduce the number of combinations (subsets) which will be generated in the following steps.

5. Generate smaller (reduced) activity subsets (X_t) using the remaining eligible activities that are involved in the conflict(s).

6. Identify feasible subsets where all the activities contained in each subset can be scheduled simultaneously without creating a resource conflict. In other word, X_t is feasible if, for all types of resources:

$$\sum_{\forall i \in X_t} r_{ij} < R_j - Ru_j \tag{3.21}$$

X_t is unfeasible otherwise.

If all subsets are feasible, then execute step 7.

7. Decompose each unfeasible subset into smaller subsets. The same procedure (Step 6 and 7) is used again to treat the newly generated subset(s). This process is repeated until all necessary feasible subsets are identified.

8. Calculate the next time frame and the remaining total floats for each and every feasible subset using Eqs. 3.13 and 3.14, respectively.

9. For each subset, calculate the negative impact (**nI**) for each postponing activity **z** using either Eqs. 3.15 or 3.16, depending on the **RTF** calculated from the previous step.

10. Compute the negative impact (**NI**) caused by each feasible subset **X** using Eq. 3.17.

11. Select the best feasible subset (i.e. the subset which has numerically the largest **NI** value), and assign the required resources to all activities in this subset. Then, advance to next time frame of the best subset.

12. Identify activities in-progress following the same procedure as described in step 2), update the resource pool by allowing activities in-progress retain all required resources, and calculate the amount of resource in-use (RU_j) for all types of resources using:

$$RU_j = \Sigma r_{ij}, \text{ where } i \in P$$

r_{ij} = amount of resource **j** required by activity **i**.

13. Repeat steps 1 to 12 until all project activities are scheduled.

The above resource allocation cycle is depicted in Fig. 3.4.

A CPM network extracted from the literature (Shanmuganayagum 1989) is used to demonstrate the calculations involved in LIM. The network diagram, its input

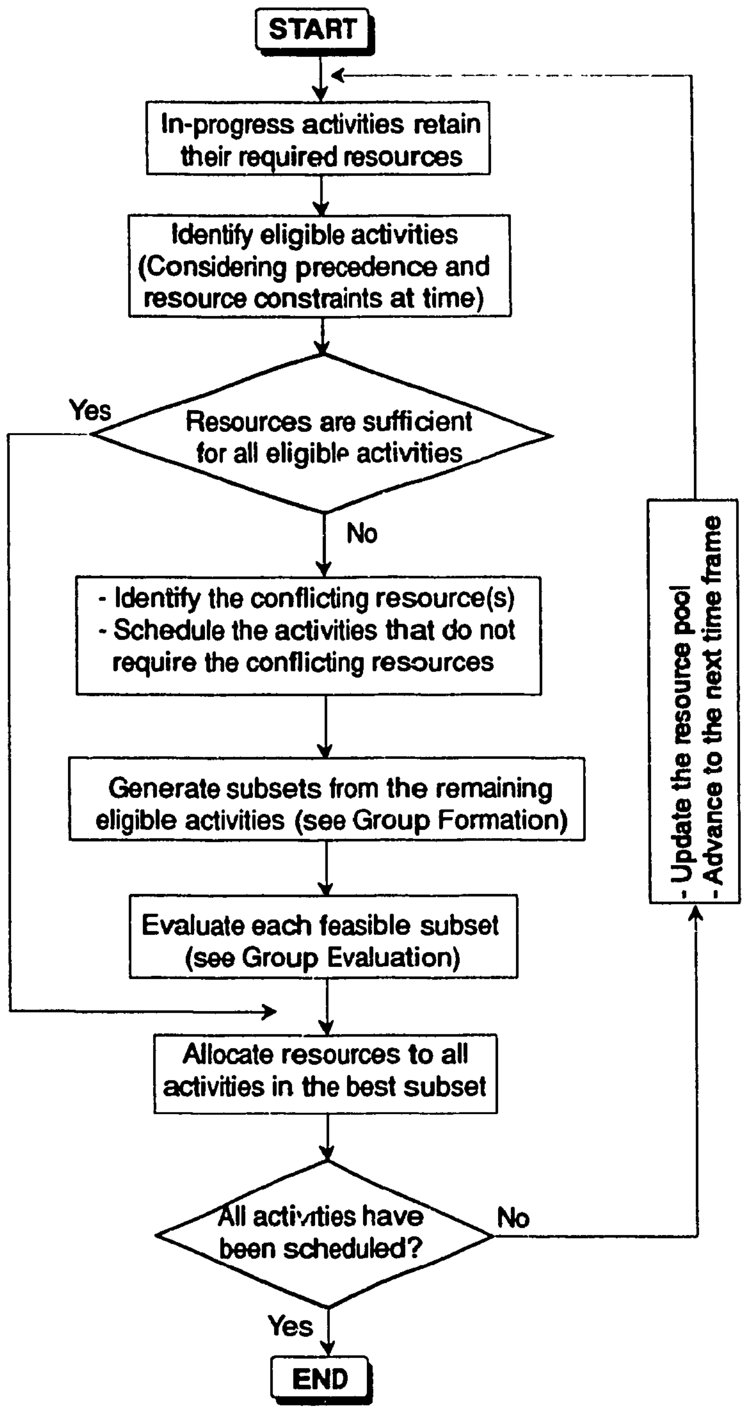


Fig. 3.4: Proposed Resource Allocation Cycle.

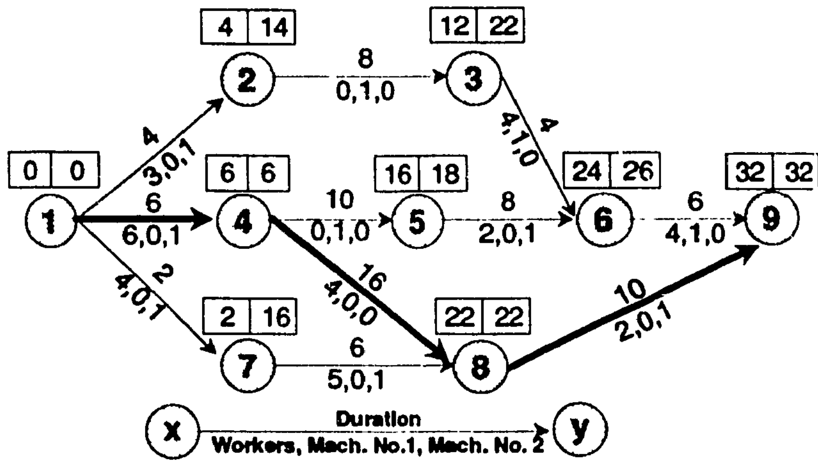
data, and CPM output are depicted in Fig. 3.5. The network consists of eleven activities, each activity can consume up to three types of resources. For simplicity, it is assumed that the weight factors (**w**) for all activities are identical and equal to 1.0. Therefore the negative impact (**nI**) and the remaining total float (**RTF**) of each postponing activity are identical. Table 3.3 shows the step-by-step calculations performed using LIM. The overall project duration after resource allocation is 40 days which is eight days beyond the original CPM duration.

3.3.5 Identification of Critical Activities

In the CPM, the criticality of project activities is determined from their associated total float values established from the forward and backward passes calculations. Traditionally, the total float associated with each activity signifies the amount of time allowable for that activity to be postponed without extending the unconstrained project completion time. The critical activity is the one that has a zero total float. In resource-constrained project scheduling, however, different classification for critical activities could be considered. Ideally, the criticality of a project activity should indicate not only the negative impact on the overall project duration but also the possible resource conflict that might occur as a result of postponing those activities. Such critical activities are resource-based critical activities and will be referred to here, for simplicity, as critical activities.

To enable a proper identification of critical activities, three new terms namely, the

Resource Availability
Workers: 8
Machine No. 1: 1
Machine No. 2: 1



Activity	ES	EF	LS	LF
1-4	0	6	0	6
4-5	6	16	8	18
5-6	16	24	18	26
6-9	24	30	26	32
3-6	12	16	22	26
2-3	4	12	14	22
1-2	0	4	10	14
4-8	6	22	6	22
1-7	0	2	14	16
7-8	2	8	16	22
8-9	22	32	22	32

Fig. 3.5: Example Network No.2.

Table 3.3: Demonstration of Resource Allocation using LIM.

CT	Eligible Activities	Possible Combinations	Feasible	NTF	Postponing Activities	RTF	NI	Decision
		1-2, 1-4, 1-7	No	-	-	-	-	No
		1-2, 1-4	No	-	-	-	-	No
	1-2	1-2, 1-7	No	-	-	-	-	No
0	1-4	1-4, 1-7	No	-	-	-	-	No
	1-7	1-2	Yes	4	1-4, 1-7	2,-4	-4	No
		1-4	Yes	6	1-2, 1-7	4,8	4	Yes
		1-7	Yes	2	1-2, 1-4	8,-2	-2	No
		1-2, 1-7, 4-5, 4-8	No	-	-	-	-	No
	1-2	1-2, 1-7, 4-5,	No	-	-	-	-	No
6	1-7	1-2, 1-7, 4-8	No	-	-	-	-	No
	4-5	1-2, 4-5, 4-8	Yes	10	1-7	4	4	Yes
	4-8	1-7, 4-5, 4-8	Yes	8	1-2	2	2	No
		1-2, 1-7	No	-	-	-	-	No
10	1-7	1-7	Yes	12	-	-	-	Yes
16	2-3	2-3, 5-6	Yes	24	-	-	-	Yes
	5-6							
24	7-8	7-8, 3-6	No	-	-	-	-	No
	3-6	7-8	Yes	30	3-6	-8	-8	Yes
		3-6	Yes	28	7-8	-12	-12	Yes
30	3-6	3-6, 8-9	Yes	34	-	-	-	Yes
	8-9							
34	6-9	6-9	Yes	40	-	-	-	Yes

Logic Float (**LF**), the Contiguous Float (**CF**), and the Intermediate Float (**IF**) are introduced. After resource allocation, each project activity is assigned a new scheduled start (**SCS**) and a new scheduled finish (**SCF**) times. The **LF** is the time that will appear as a typical float in the traditional CPM but may or may not be utilized for further resource adjustment. The **LF** is the difference between the scheduled finish of the activity in question and the earliest scheduled start of its successor. The **CF** of an activity, on the other hand, is the resource-conflict-free time zone adjacent to its **SCF**. The **IF** float is the resource-conflict-free time zone that is located within the logic float, but outside the **CF**. The **IF** can be described in terms of an interval of the next feasible scheduled start times, bounded by the earliest and the latest scheduled start times. The critical activities are those that contain zero **CF** and **IF**.

3.3.6 Application of the new floats

The same network example shown in Fig. 3.5 is used to illustrate the application of the proposed new floats. The scheduled start and finish times obtained using LIM, and the calculated new floats are summarized in Table 3.4. The schedule after allocation, illustrated using a proposed barchart, is depicted in Fig. 3.6. It can be seen from this figure that the critical activities are 1-4, 4-5, 5-6, 6-9, 3-6, 7-8, and 8-9. Activities 1-4, 4-5, 6-9, 3-6, 7-8, 8-9 are critical because they do not have **LF**. Activity 5-6, although has an **LF** of 10 days, but there is no flexibility in rescheduling this activity to benefit from that float since, both, the **CF** and **IF** of this

Table 3.4: Results Produced by LIM.

Activities	Duration (Days)	SCS (Working days)	SCF (Working days)	LF (Days)	CF (Days)	IF (Working Dyas)		Critical
						ESI	LSI	
1-2	4	6	10	6	0	12	12	No
1-4	6	0	6	0	0	-	-	Yes
1-7	2	10	12	12	4	-	-	No
2-3	8	16	24	6	6	-	-	No
3-6	4	30	34	0	0	-	-	Yes
4-5	10	6	16	0	0	-	-	Yes
4-8	16	6	22	8	2	-	-	No
5-6	8	16	24	10	0	-	-	Yes
6-9	6	34	40	0	0	-	-	Yes
7-8	6	24	30	0	0	-	-	Yes
8-9	10	30	40	0	0	-	-	Yes

SCS = Scheduled start time; SCF = Scheduled finish time; LF = Logic float; CF = Contiguous float; IF = Intermediate float
ESI = Earliest start due of the intermediate float; LSI = Latest due of the intermediate float.

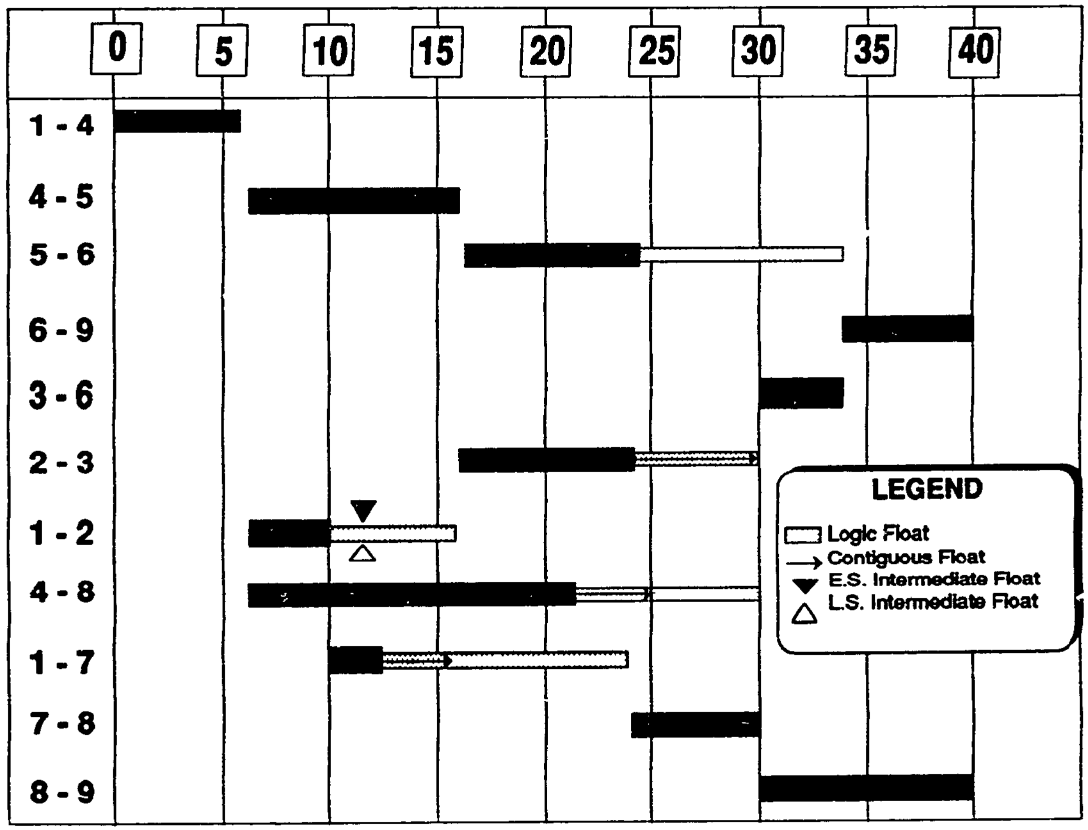


Fig. 3.6: Critical Activities in Resource-Constrained Scheduling.

activity are nil. An attempt to utilize the **LF** will result in a new resource conflict, invalidating the rest of the schedule.

Activities 2-3, 4-8, and 1-7 are not critical since their **CFs** are not zero. This provides the scheduler with a flexibility to start these activities late or allow them to be delayed within their respective **CFs**. Activity 1-2 contains an **LF** of 5 days, zero **CF**, and an interval of one day of intermediate float. The zero **CF** indicates that if this activity is to be started as scheduled, the extension of the duration will cause a new resource conflict. However, there is another time-slot available to start this activity. That time-slot is marked by the next earliest and latest possible start times located within its **LF**. For activity 1-2, the next earliest possible start and the next latest possible start times coincide. If these two start times do not coincide, there would be more flexibility to alter the rescheduled start times of the activity.

The proposed new floats can also provide information regarding the resource availabilities. For example, in the situation where there is a change in the quantity of work for an activity, the duration of that activity can increase. The scheduler can know immediately whether the rest of the schedule is affected by such a change without having to re-calculate the entire schedule. These new floats can easily be used to indicate whether additional resources are required or the available resources can be used to perform the additional work without creating any new resource conflict.

3.4 Performance Test

The performance of LIM, with regard to the project durations produced, resource utilization, and resource interruptions, is investigated. A two-step testing procedure (i.e. the preliminary, and expand tests) is designed to facilitate the study. The preliminary test involves the identification of the three best heuristic rules, considering only the resulting project durations (Moselhi and Lorterapong 1993b). The main purpose of conducting the preliminary test is to select the best performance rules that produce near optimal project durations, and to compare them with the study done by Davis and Patterson (1975). The expand test is then used to test the performance of the selected rules, considering all the combinations of the three criteria.

A computer program is developed specifically to facilitate these two tests. The program has been coded in BASIC and run on IBM PC and compatible environments. The program prompts the user for input data including the traditional CPM data, resource demands for each activity, and resource availabilities. It then generates the traditional CPM output including the early and late times, the total float of each activity, and the project duration. Upon the completion of the CPM analysis, the program performs resource allocation using the five heuristic rules: LIM, the minimum late finish rule (MINLFT), the shortest duration (SHD), the greatest resource demand (GRD), and the minimum slack (MINSLK) rules.

3.4.1 Preliminary Test

Nine networks adopted from the literature are used in the preliminary test. The sources and the characteristics of these networks are summarized in Table 3.5. The number of activities in these examples range from 11 to 49, the number of resource types consumed vary from 2 to 6, and the CPM durations vary from 18 to 78 days. Each of the nine network example is imposed with different resource constraints, resulting in a total of 31 different cases. The performance of the proposed model is compared with four highly regarded heuristic rules including the MINLSK, MINLFT, GRD, and SHD. In addition, six available optimal solutions and three RSM solutions are utilized for comparison with solutions obtained using the four rules.

3.4.2. Analysis of Preliminary Test Results

The project durations obtained using each heuristic rule, together with available RSM and optimal solutions are listed in Table 3.6. The results summarized in this table are used for measuring the performances of the proposed model against other heuristic rules based on the following objectives:

1) Average percent increase over the shortest duration encountered using the five heuristic rules.

2) Number of shortest project durations obtained.

Table 3.5: Characteristics of Example Networks Used In The Test.

Group No.	Network Number	Author	No. of Activities	Maximum Types of Resources/Activity	CPM Duration
1	1-3	Brand et al.(1964)	13	2	36
2	4-6	Brand et al.(1964)	39	3	78
3	7-9	Davis & Patterson (1971)	35	3	31
4	10-12	Willis & Hasting (1976)	38	6	33
5	13-16	Suarez & Lybrand (1984)	15	3	18
6	17-20	Mohanty & Siddiq (1989)	26	3	23
7	21-24	Mohanty & Siddiq (1989)	23	3	35
8	25-28	Mohanty & Siddiq (1989)	49	3	42
9	29-31	Shanmuganayagam (1989)	11	3	32

Table 3.6: Project Durations Produced by Different Models (Preliminary Test).

Network No.	OPT	RSM	MINSLK	MINLFT	GRD	SHD	LIM
1	39	N/A	43	47	43	47	40
2	N/A	N/A	49	48	49	49	47
3	N/A	N/A	40	40	40	40	37
4	N/A	91	91	91	91	93	91
5	N/A	100	94	94	94	94	93
6	N/A	N/A	85	85	85	88	84
7	64	74	74	74	77	77	70
8	N/A	N/A	74	74	83	80	73
9	N/A	N/A	59	57	74	60	57
10	35	N/A	39	37	39	36	36
11	N/A	N/A	43	45	43	47	45
12	N/A	N/A	58	[illegible]	57	64	58
13	20	N/A	22	24	24	26	22
14	N/A	N/A	23	21	25	25	23
15	N/A	N/A	22	22	22	22	22
16	N/A	N/A	32	32	38	36	36
17	N/A	N/A	35	31	31	30	30
18	N/A	N/A	35	31	31	30	30
19	N/A	N/A	32	29	29	29	27
20	23	N/A	24	24	24	24	23
21	N/A	N/A	41	41	44	45	39
22	N/A	N/A	41	41	44	45	37
23	N/A	N/A	37	37	44	45	37
24	N/A	N/A	37	37	44	45	37
25	N/A	N/A	55	54	68	59	51
26	N/A	N/A	50	47	65	49	51
27	N/A	N/A	48	45	58	45	47
28	42	N/A	43	43	43	43	42
29	N/A	N/A	40	40	40	52	40
30	N/A	N/A	44	44	50	48	44
31	N/A	N/A	46	44	46	48	46

OPT: Optimum duration
RSM: Resource Scheduling Method
SHD: Shortest project duration
MINSLK: Minimum slack rule
GRD: Greatest resource demand
MINLFT: Minimum late finish rule
LIM: Least Impact model

3) Number of shortest project durations obtained only by the heuristic rule being evaluated.

4) Number of longest project durations obtained.

5) Number of longest project durations obtained only by the heuristic rule being evaluated.

The above stated objectives are applied, and the results are summarized in Table 3.6. The desirable features of any heuristic rule can be measured by the smallest average percentage increase above the shortest project durations and the largest number of the shortest duration obtained. From Table 3.7, it can be seen that among the five models compared, LIM has the smallest average percent increase above the shortest duration encountered, followed by those of MINLFT and MINSLK respectively. With respect to the frequency of obtaining shortest durations, LIM also outperforms the other four rules by producing 24 out of 31 shortest project durations, of which, 15 are exclusive solutions.

The undesirable features are measured by the number of longest project durations produced. The results indicate that the proposed model does not produce any of the longest project durations encountered. For the examples where the RSM solutions are available (examples 4, 5, and 7) the proposed model outperforms all heuristic rules considered including the RSM method. For the examples in which the optimal solutions are available (i.e. examples 1, 7, 10, 13, 20, and 28), LIM

Table 3.7: Preliminary Test Results.

Criteria	LIM	MINLFT	MINSLK	GRD	SHD
Average percent increase above the shortest duration	1.18	3.29	4.28	9.56	10.79
Frequency of obtaining the shortest project duration	24	11	9	6	5
Frequency of obtaining the shortest project exclusively by this method	15	3	0	0	1
Frequency of obtaining the longest project duration	0	5	9	18	15
Frequency of obtaining the longest project duration exclusively by this method	0	0	3	11	8

LIM: Least Impact Model
GRD: Greatest Resource Demand
SHD: Shortest Project Duration
MINSLK: Minimum Slack
MINLFT: Minimum Late Finish

also produce solutions closer to the optimal solutions than those obtained from the other heuristic rules.

In sum, out of 31 tested examples, the proposed model outperforms both the MINSLK and the MINLFT in 58% of the cases and was found equal or better in 91% (against the MINSLK) and 83% (against the MINLFT). In comparison with the six available optimal solutions, the proposed model also outperforms the MINSLK and the MINLFT rules in all of the cases examined. Considering the three available RSM solutions, the proposed model also proved superior to the RSM. Out of the three examples, the proposed model produces better results in two and identical results in the third. The results indicate that the least impact model is superior to the other four heuristic rules in all objectives stated above. The second and the third best rules are the MINLFT and the MINSLK rules respectively. With regard to the effectiveness of the MIN LFT and the MINSLK rules, the findings of this study agree with those of Davis and Patterson (1975).

3.4.3 Expanded Test

The three best models including LIM, MINLFT, and MINSLK are further examined. This test considers: 1) project durations, 2) resource utilizations, and 3) resource interruptions. Fifty one networks are used in the expanded test. Each network is analyzed using the computer program described earlier. Each time the program calculates, in addition to the scheduling start and finish times for each activity, a project duration, an average resource utilization index, and an average resource interruption index. Both, resource utilization and resource continuity are of crucial, particularly, to those that are expensive, scarce, or acquired from elsewhere. Therefore, it is worthwhile to investigate the performance of the heuristic-based models in this regard.

In this study, the utilization factor for each type of resource (U_i) and the average resource utilization indexes (U_{avg}) are determined from the following equations:

$$U_i = \frac{\text{Amount of resource i used (Man-day)}}{\text{Amount of resource i available (Man-day)}} \tag{3.23}$$

$$U_{avg} = \frac{1}{n}\sum_{i=1}^{n} U_i \tag{3.24}$$

where **n** is the number of resource types used in the project.

The resource interruption index, on the other hand, is determined from the number of times each resource is interrupted (i.e. being demobilized and then remobilized). Both resource utilization and resource interruption indices are computed after resource allocation is completed.

3.4.4 Analysis of Expanded Test Results

The results of the expanded test are summarized in Tables 3.8 and 3.9. In Table 3.8, the number of successful solutions obtained by each model, classified

Table 3.8: Performance of Heuristic Models Classified by Individual Objectives*.

Scheduling Objectives	Number of Projects Produced		
	LIM	MINSLK	MINLFT
Minimum project duration	42	17	18
Minimum project duration exclusively by this method	28	2	5
Maximum resource utilization	33	22	22
Maximum resource utilization exclusively by this method	23	4	6
Minimum resource interruption	33	21	22
Minimum resource interruption exclusively by this method	20	6	7

LIM = Least Impact method; MINSLK = Minimum Slack; and MINLFT = Minimum Late Finish

* that the solutions obtained satisfy other criteria as a by product

according to an individual scheduling objective are presented. Both, the total number of best solutions and the exclusively best solutions are considered in this test. The exclusive solution is the best solution that is generated only by a particular model. Therefore, the number of exclusive solutions can be used to indicate the exceptional efficiency of each model.

Consider first the objective related to project durations. Of all the networks tested (i.e. 51 cases), LIM produces 82% of the total solutions, and 50% of those are exclusive solutions. The MINSLK model generates a total of 33% of all solutions, while only 4% of those are exclusive. Similarly, the MINLFT model generates about 35% solutions, 10% of which are exclusive. In the case of resource utilization, LIM produces 65% of all solutions, and 45% of those solutions are exclusive. The MINSLK and MINLFT models, on the other hand, generate 43 % of all solutions equally. The exclusive solutions of the MINLFT model is 12% which is slightly better than that of the MINSLK (i.e. 8%). With respect to the resource interruption objective, LIM provides 65% of all solutions while 39% of those solutions are exclusive. Meanwhile, the MINSLK model has 41% of all solutions, only 12% are exclusive. The MINLFT produces about 43% of the total solutions, and 14% of those are exclusive.

The above findings are graphically shown in Fig. 3.7. In general it can be stated that LIM is superior to the MINLSK and MINLFT with respect to all the three

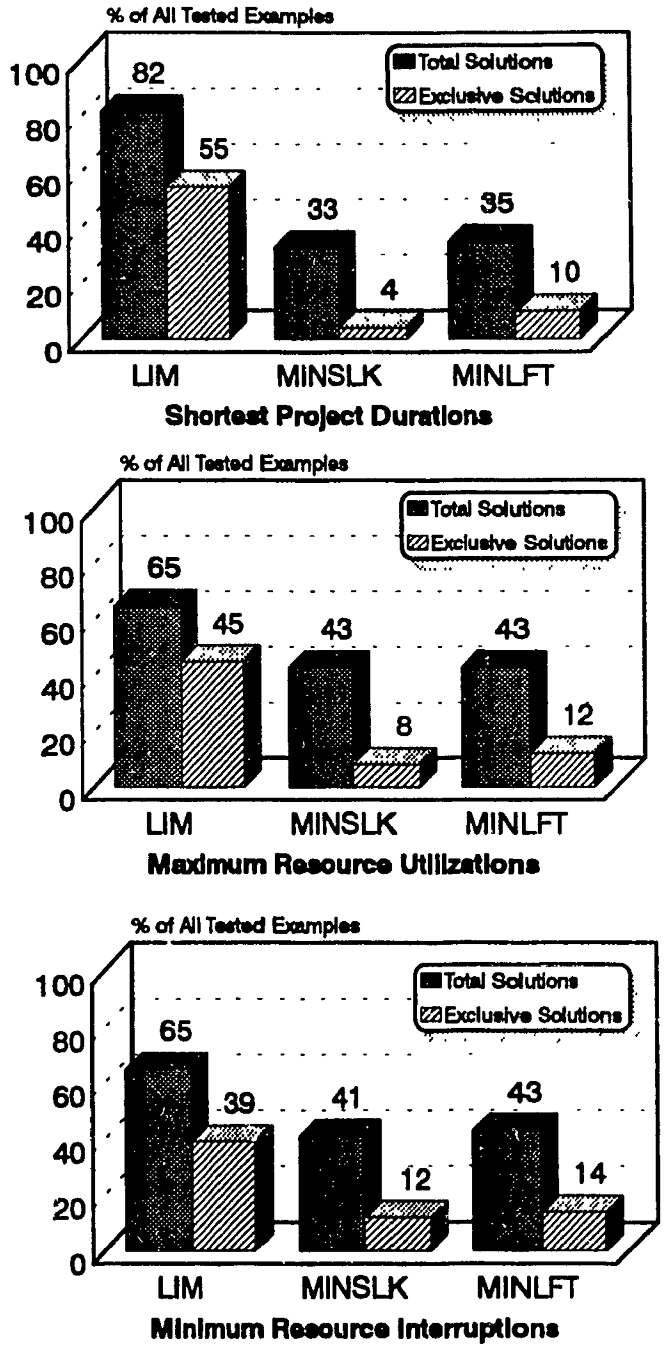


Fig. 3.7: Performance of Heuristic Models Classified by Objective.

objectives being considered. LIM is exceptionally better than the other two models with regard to the project duration produced and significantly better with respect to resource utilization and interruption. Further, the percentage of exclusive solutions for each objective produced by LIM are exceptionally higher than those obtained with the other two models.

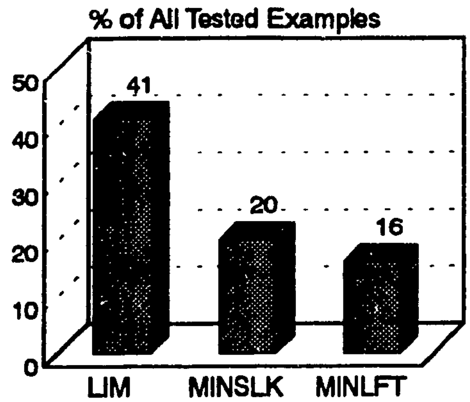
In addition to the performance classified by each objective, it is interesting also to investigate the performance of the three models based on multi-objective characteristics. The model is considered to be effective if it produces high number of solutions which satisfies more than one objective simultaneously. Table 3.9 summarizes the results of this test and Fig. 3.8 illustrates the trends. In interpreting the results shown in Fig. 3.8, the higher the number of objectives being simultaneously satisfied, the better the performance of the model.Consider first the solutions that satisfy the three objectives simultaneously (see Fig. 3.8a). It can be seen that LIM produces about 41% of the total 51 solutions. This number is about twice as many of those obtained using the other two models. A similar trend is also depicted in Fig. 3.8b.

With respect to the solutions that fail to satisfy more than one objective (i.e. the solution which satisfies either minimum project duration, or maximum resource utilization, or minimum resource interruption only), LIM is superior to the other two models (see Fig. 3.8c). LIM generates only 10% of the total solutions in this

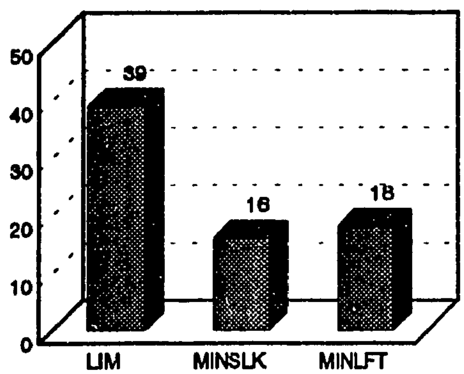
Table 3.9: Performance of Heuristic Models based on Single and Multiple Objectives.

Number of project durations which satisfy:	Number of Projects Produced		
	LIM	MINSLK	MINLFT
1) Minimum project duration only	3	2	3
2) Maximum resource utilization only	2	5	5
3) Minimum resource interruption only	-	10	12
4) Minimum project duration & Maximum resource utilization	8	4	5
5) Minimum project duration & Minimum resource interruption	10	1	2
6) Maximum resource utilization & minimum resource interruption	2	3	2
7) Minimum project duration & Maximum resource utilization & Minimum resource interruption	21	10	8

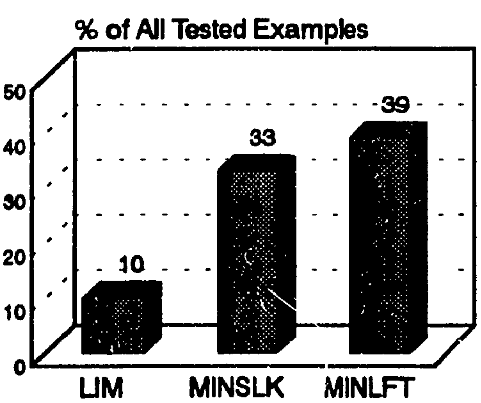
LIM = Least Impact method; MINSLK = Minimum Slack; and MINLFT = Minimum Late Finish



a) Solutions that satisfy three objectives simultaneously



b) Solutions that satisfy any two objectives only.



c) Solutions that fail to satisfy more than one objective.

Fig. 3.8: Multiple Objectives Performance of Heuristic Models.

category. This number is significantly lower than those produced by MINSLK and MINLFT (see Fig. 3.8c). These results reveal that the schedule obtained using LIM tends to have better quality than those generated by the MINSLK and MINLFT. The percentage of solutions that satisfies only one objective (see Fig. 3.8c) should not be confused with those shown in Fig. 3.7. They are not related.

3.5 Statistical Tests

A non-parametric model, Wilcoxon signed-rank sum test, was used to examine the significance of the difference between the three rules. The model is justified for the following reasons. First, there is no evidence that the delays, the average resource utilization, or the average resource interruption of the projects are normally distributed. Secondly, the way in which the test was conducted can be regarded as a matched pairs experiment (Keller et. al. 1988). That is, each of the 51 project networks was solved three times using the three rules. The three results of that project were then compared in a pairwise manner. The repetitions of the use of each network render the results of scheduling to be non-independent.

The three null hypothesises (H_0) can be specified that there are no significant differences in the performance of the two rules being compared with respect to: 1) project delays, 2) average resource utilization, and 3) average resource interruption. The alternative hypothesises can respectively be specified as: the proposed resource allocation model produces results significantly better than those of the MINSLK and MINLFT rules for all the three objectives measured.

Applying the Wilcoxon signed-rank test to the data shown in the Tables listed in Appendices I, II, III. The critical values in the Wilcoxon Signed Rank Sum Test used for rejecting the three hypothesises are taken from Keller et al. (1988). The analysis reveals that all the three null hypothesises are rejected. This indicates the significant difference between the performance of the proposed model and the two best traditional heuristic rules. With respect to the project duration, the data provides sufficient evidence at the 1.0% significance level that the proposed model outperforms the other two rules (see Appendix I. Regarding the resource utilization, it is found that the proposed model is better than the MINSLK and the MINLFT models at the significance level of 2.0% and 5.0% respectively (see Appendix II). Considering the resource interruption, the proposed model is also proved to be superior to the MINSLK and MINLFT models at the significance level of 10% and 5% respectively (see Appendix III).

3.6 Summary

From the results of the test, it can be seen that the least impact model outperforms the highly regarded heuristic rules including the MINSLK and the MINLFT rules in all three scheduling objectives (i.e. minimal delays, maximum resource utilization, and minimum resource interruption) considered. The performance of the least impact model is exceptionally superior to the other two models with regard to the project durations and resource utilization, and to a lesser degree with respect to resource interruptions. The least impact model also exhibits a characteristic which is of significance to the resource-constrained scheduling. That is, it has a higher tendency to generate solutions that can achieve all, or a combination of any of the two scheduling objectives simultaneously than the other two rules. Therefore, the least impact model can be viewed as a significant improvement to the heuristic-based resource allocation models. It provides a reasonable trade-off between the quality of solutions obtained using the optimization-based models and the practicality of the heuristic-based models.

CHAPTER 4

FUZZY NETWORK ANALYSIS

4.1 Introduction

Today, analytical tools such as the Critical Path Method (CPM) and Precedence Diagram Method (PDM) have been used extensively to analyze construction project networks (Tavakoli and Riachi 1990). Both methods assume scheduling problems to be deterministic, requiring definitive project data (i.e. activity durations) as input. In real-life operations, however, construction projects are normally executed under uncertain environments. With these uncertainties surrounding activities and resources data, it is unlikely that such deterministic methods can be used effectively. To circumvent this limitation, considerable research effort has been made to incorporate the underlying uncertainties into network scheduling. Previous solutions are based primarily on probabilistic theory as outlined in Chapter 2.

Probabilistic methods use marginal probability distributions to represent the characteristics of the scheduling variables being modeled. The use of probabilistic distributions implicitly assumes that past performance of the same activity has been observed, and that a marginal distribution has been constructed from these observations. This distribution is then used in the scheduling calculations performed on new projects. In practice, however, it is commonly known that no two construction projects are alike. The conditions for executing those projects at activity levels may also vary from one project to another. Frequently, scheduling experts have to exercise their own judgement based on their experiences, subjective knowledge and/or gut-feelings of the project at hand, and try to come up with reasonable estimations for the durations of project activities involved. These durations, if acquired in a nonstatistical manner, should be processed using a more direct and a more suitable method than conventional probabilistic methods.

This chapter presents a new nondeterministic network calculations method for resource-constrained scheduling. The proposed method employs fuzzy set theory (Zadeh 1965) for modeling uncertainties associated with the duration of project activities and resource availabilities. The method utilizes traditional fuzzy arithmetic operations and a number of newly developed techniques to propagate the underlying uncertainties through the project network. The following section describes the fundamental concepts that are necessary for the development of the proposed method. These concepts are summarized from the literature (Zadeh 1965, Dubois and Prade 1988, Zimmerman 1991).

4.2 Fuzzy Set Theory

Fuzzy sets theory was developed specifically to deal with uncertainties that are not statistical in nature (Zadeh 1965). A fuzzy set is a class of objects associated with their respective degrees of membership within the set. Fuzzy sets differ from the conventional crisp sets mainly in the degrees by which an object belongs to a set. In the crisp set theory, objects are either included or excluded from a set. In the fuzzy sets theory, on the other hand, objects are described in such a way to allow a gradual transition from being a member of a set to a nonmember. Each object contains a degree of membership ranging from zero to one, where zero signifies nonmembership, one indicates full membership, and the values between zero and one describe the degree of partial membership. As such, fuzzy sets can be viewed as a generalization of crisp set theory.

4.2.1 Definition of a Fuzzy Set

Let **X** be a collection of objects whose members are denoted by **x**.

1) A fuzzy set **A**, by definition, is characterized by the set of pairs:

$$A = \{(x, \mu_A(x)) \mid x \in X\} \tag{4.1}$$

where $\mu_A(x)$ is the grade of membership or membership function value of **x** in **A**. The grade of membership is determined by membership functions defined subjectively.

2) The *support* of fuzzy set **A**, S(A) is the set of objects **x** in **X**, where the degrees of membership are greater than zero.

$$S(A) = \{(x, \mu_A(x) > 0) \mid x \in X\} \tag{4.2}$$

3) A fuzzy set **A** is normal if there is at least one $x \in X$ such that $\mu_A(x) = 1.0$.

4) A fuzzy set **A** is convex if its membership distribution contains only one distinct peak:

$$\forall x_1 \in X,\ \forall x_2 \in X,\ \forall \lambda \in [0.1], \quad \mu_A(\lambda x_1 + (1-\lambda)x_2) \geq \min(\mu_A(x_1), \mu_A(x_2)) \tag{4.3}$$

5) An α-cut of fuzzy set **A** is crisp set whose elements belong to fuzzy set **A** at least to the degree of α. The α-cut of fuzzy set **A** is defined as:

$$A_\alpha = \{(x, \mu_A(x) \geq \alpha) \mid x \in X\} \tag{4.4}$$

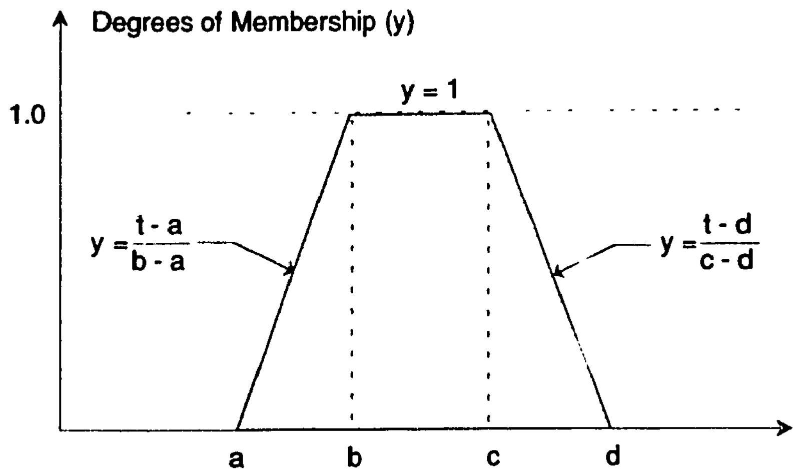
From the practical view point, the convex fuzzy set is more suitable for representing the fuzzy concepts such as "about" and "approximately" than the nonconvex fuzzy set. Therefore, all of the fuzzy sets described in this study are convex.

4.2.2 Fuzzy Numbers

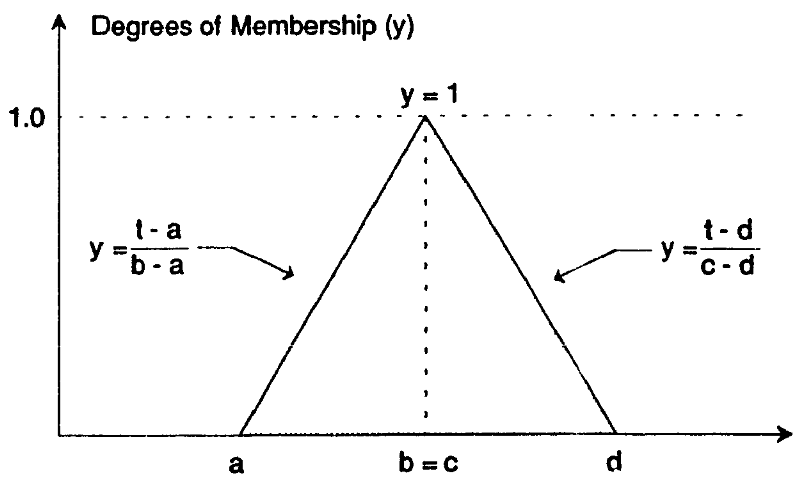
A fuzzy number is a continuous fuzzy set that contains two properties: 1) convexity (i.e. one distinct peak) and, 2) normality (i.e. at least one element in the set has a membership value equal to 1.0). Fuzzy numbers have been employed extensively in the development of mathematical models (Dubois and Prade 1988, Kaufmann and Gupta 1988). They are used to represent imprecise numerical quantities such as "approximately 10 days," "about 8 weeks," etc. Though fuzzy numbers can take various shapes, linear approximations such as triangular and trapezoidal fuzzy numbers are used frequently. A trapezoidal fuzzy number, **F**, can be represented by a quadruple (**a**,**b**,**c**,**d**) where **a** and **d** are the lower and the upper bounds, while **b** and **c** are the lower and the upper modal values respectively. A fuzzy triangular fuzzy number can be viewed as a special case of the trapezoidal number by which the lower and the upper modal values are identical (i.e. **b** = **c**). Fig. 4.1 shows the graphical representations of these two fuzzy numbers. The generic membership function of a trapezoidal fuzzy number is defined as:

$$\mu(t) = \begin{cases} \dfrac{t-a}{b-a} & a<t<b \\ 1 & b \le t \le c \\ \dfrac{t-d}{c-d} & c<t<d \\ 0 & \text{otherwise} \end{cases} \qquad (4.5)$$

In addition to the trapezoidal and triangular fuzzy distributions, other forms of representation such as single-valued variables, interval (i.e. a range) variables, and unbound variables can be found in the scheduling applications. These forms, also, can effectively be represented using the quadruple. For instance, the single-valued variable, can be represented by (**a**,**a**,**a**,**a**), the interval variable the quadruple can be characterized by (**a**,**a**,**b**,**b**), and the lower and upper fuzzy boundaries can be represented by the quadruples (**a**,**b**,∞,∞) and (-∞,-∞,**c**,**d**) respectively.



(a) Trapezoidal Fuzzy Numbers



(b) Triangular Fuzzy Numbers

Fig. 4.1: Triangular and Trapezoidal Fuzzy Numbers.

4.2.3 Fuzzy Set Operations

In fuzzy set theory, arithmetic operations (e.g. addition, subtraction, multiplication, etc.) and set theoretic operations (e.g. union, intersection, etc.) are performed using a technique called the *max-min convolution* (Kaufmann and Gupta 1985). The max-min convolution of two fuzzy numbers is represented as:

$$\mu_{A \odot B}(z) = \bigvee_{z = x \cdot y} (\mu_A(x) \wedge \mu_B(y)) \qquad (4.6)$$

where $\odot$ symbolizes fuzzy operations; and $\cdot$ can be addition (+); subtraction (-), multiplication (×), division (÷), intersection ($\wedge$) or union ($\vee$). Considering for example, fuzzy addition, the max-min convolution is:

$$\mu_{A \oplus B}(z) = \bigvee_{z = x + y} (\mu_A(x) \wedge \mu_B(y)) \qquad (4.7)$$

There are three steps required in the calculation of the degree of membership for each **z** value. The first step involves the identification of each and every pair of **x** and **y** that when applying the operation in question (e.g. addition) would result in an identical **z** value. The second step is the determination of the degree of membership associated with each pair identified in the first step. This degree of membership is the smallest degree of membership between those of **x** and **y** in that pair (i.e. $\mu_A(x) \wedge \mu_B(y)$). Thirdly, the degree of membership of each **z** assumes the largest value of the degree of membership of all the pairs identified earlier. The above procedure is then repeated for all possible **z** values.

It can be seen that the computational efforts required by the max-min convolution technique are quite cumbersome particularly when dealing the calculations of real-life problems. A major advantage of using the quadruple is that some of the exact operations which are based on the max-min convolution can be replaced by their respective direct operations (Chen and Hwang 1992, and Dubois and Prade 1988). For example, let **M** = (a_1,b_1,c_1,d_1), **N** = (a_2,b_2,c_2,d_2) be any two fuzzy numbers. The direct operations used to perform on these two numbers are as follows:

$$M \oplus N = (a_1 + a_2, b_1 + b_2, c_1 + c_2, d_1 + d_2) \quad (4.8)$$

$$M \ominus N = (a_1 - d_2, b_1 - c_2, c_1 - b_2, d_1 - a_2) \quad (4.9)$$

$$\tilde{\max}(M,N) = (\vee(a_1,a_2), \vee(b_1,b_2), \vee(c_1,c_2), \vee(d_1,d_2)) \quad (4.10)$$

$$\tilde{\min}(M,N) = (\wedge(a_1,a_2), \wedge(b_1,b_2), \wedge(c_1,c_2), \wedge(d_1,d_2)) \quad (4.11)$$

$$M \cap N = \{x, \mu_{M \cap N}(x)\}, \text{ where } \mu_{M \cap N}(x) = \mu_M(x) \wedge \mu_N(x) \quad (4.12)$$

$$\text{NOT } M\ (\neg M) = \{x, 1 - \mu_M(x)\} \quad (4.13)$$

$$M \subseteq N \text{ iff } \mu_M(t) \leq \mu_N(t),\ \forall t \in T \quad (4.14)$$

where $\oplus$ represents fuzzy addition; $\ominus$ symbolizes fuzzy subtraction; $\vee$ denotes supremum; $\wedge$ represents infremum; and $\tilde{\max}$ and $\tilde{\min}$ symbolize fuzzy maximum and fuzzy minimum respectively.

The fuzzy numbers and operations described in this subsection are used to model the imprecise temporal information in the scheduling of project networks. These temporal information are discussed in the following section.

4.3 Time representation

In the scheduling context, knowledge about time is usually expressed in terms of dates and durations. A date is used to specify scheduling events such as milestones, contractual dates, material delivery dates, etc., while the duration is an estimated time unit required to perform a project activity. Project uncertainties can result partially from various degrees of imprecision that are inherent in these time expressions. The imprecision of times can be expressed quantitatively using fuzzy set numbers. The following subsections discuss various profiles of time elements, normally encountered in construction.

4.3.1 Crisp Times and Intervals

Crisp times can be regarded as a single time or a set of time elements (i.e. an interval) where the degrees of membership of every element included in the set assume a unit value (i.e. the degree of membership = 1.0), and zero otherwise. Fig. 4.2 illustrates four possible crisp times and intervals with varying degrees of imprecision. The absolute form of time in which a single value, $\mathbf{t_1}$ is certainly known (e.g. the duration of activity **A** is 15 days) is depicted in Fig. 4.2a. This form of representation is used in most deterministic scheduling methods. The membership

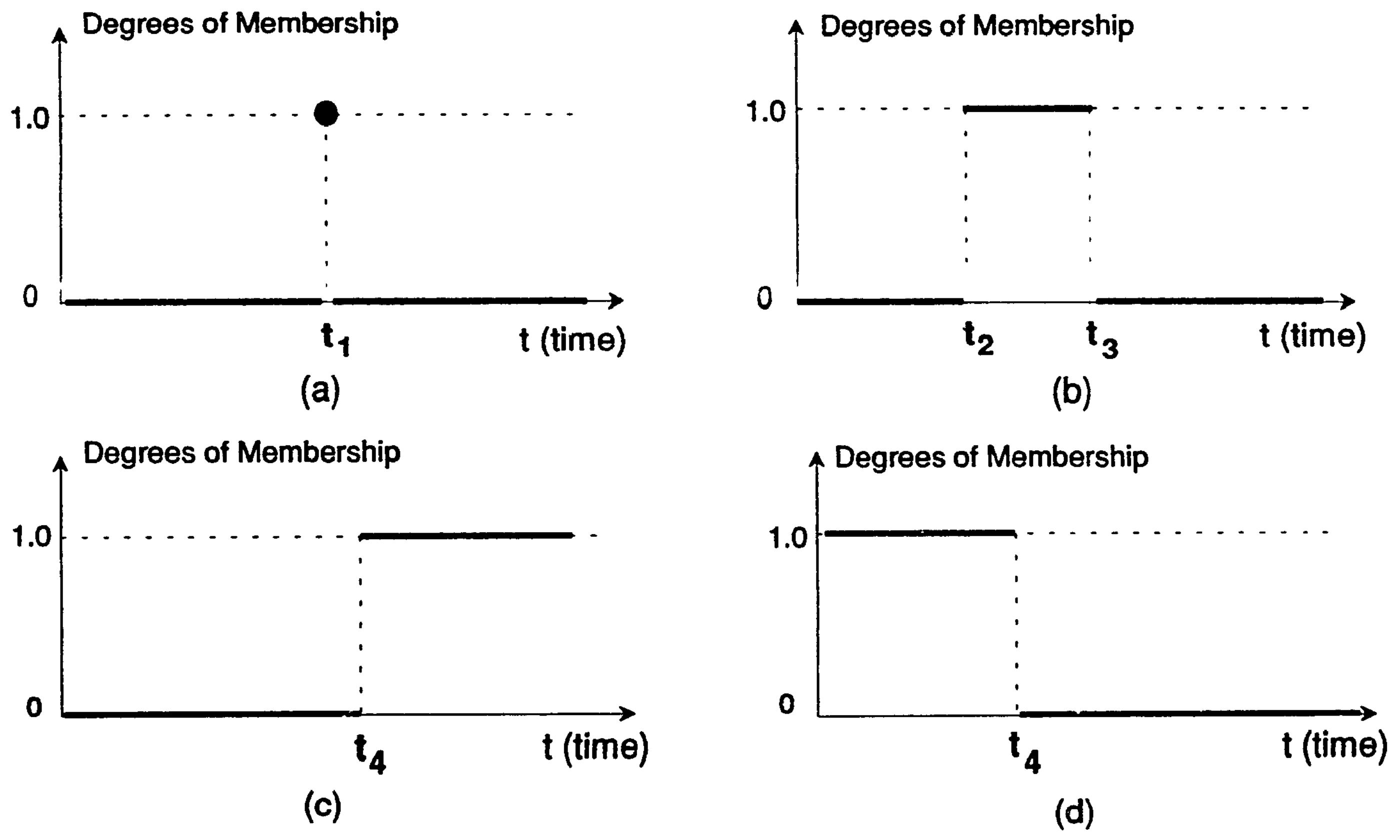


Fig. 4.2: Crisp and Interval Times.

function for this case is:

$$\mu(t) = \begin{cases} 1 & t = t_1 \\ 0 & \text{otherwise} \end{cases} \tag{4.15}$$

In the situation where a single-valued-type variable is not representative, one might, instead, want to specify a range or an interval of possible values. A closed interval, $[t_2,t_3]$, as shown in Fig. 4.2b can be used to represent this type of variable. The degree of imprecision for the interval variable clearly exceeds that of the previous case. Each time element in the interval $[t_2,t_3]$ has an equal possibility of occurrence. The membership function for a time interval is:

$$\mu(t) = \begin{cases} 1 & t_2 \le t \le t_3 \\ 0 & \text{otherwise} \end{cases} \tag{4.16}$$

In many circumstances, schedulers only know that a particular event will occur after or before a certain date. As such, it is not possible to use the closed time interval or the single time to represent such events. The interval that is bounded only from one side such as those shown in Figs. 4.2c and 4.2d are suitable for representing these situations. Fig. 4.2c demonstrates the membership function for the case where it is certain that the time in question certainly occurs after t_4 [i.e. $[t_4,\infty)$]. Fig. 4.2d, on the other hand, illustrates the time that happens certainly before t_4 (i.e.$(-\infty,t_4]$). The membership functions of these two cases are:

The after case:

$$\mu(t) \begin{cases} 1 & t_4 \geq t \\ 0 & \text{otherwise} \end{cases} \qquad (4.17)$$

The before case:

$$\mu(t) = \begin{cases} 1 & t \leq t_4 \\ 0 & \text{otherwise} \end{cases} \qquad (4.18)$$

4.3.2 Fuzzy Time Expression

It can be seen that the degree of membership function associated with each of the time elements for the representations discussed so far can either be 0 or 1.0. In real operations, schedulers may have other levels of confidence, in addition perhaps to the zero and one types, that can be associated with the occurrence of each of the possible time elements. This phenomenon can be regarded as vague or fuzzy. Mathematically, fuzziness occurs when the grade of membership of a variable assumes values between 0 and 1 (Zadeh 1965). That is, the transition from membership to non-membership is gradual rather than being sharply defined. This transition naturally represents the way schedulers approximate the values assigned to scheduling variables.

Fuzzy boundaries are imposed on each of the situations described previously, resulting in four different imprecisely-known times, as illustrated in Fig. 4.3. The

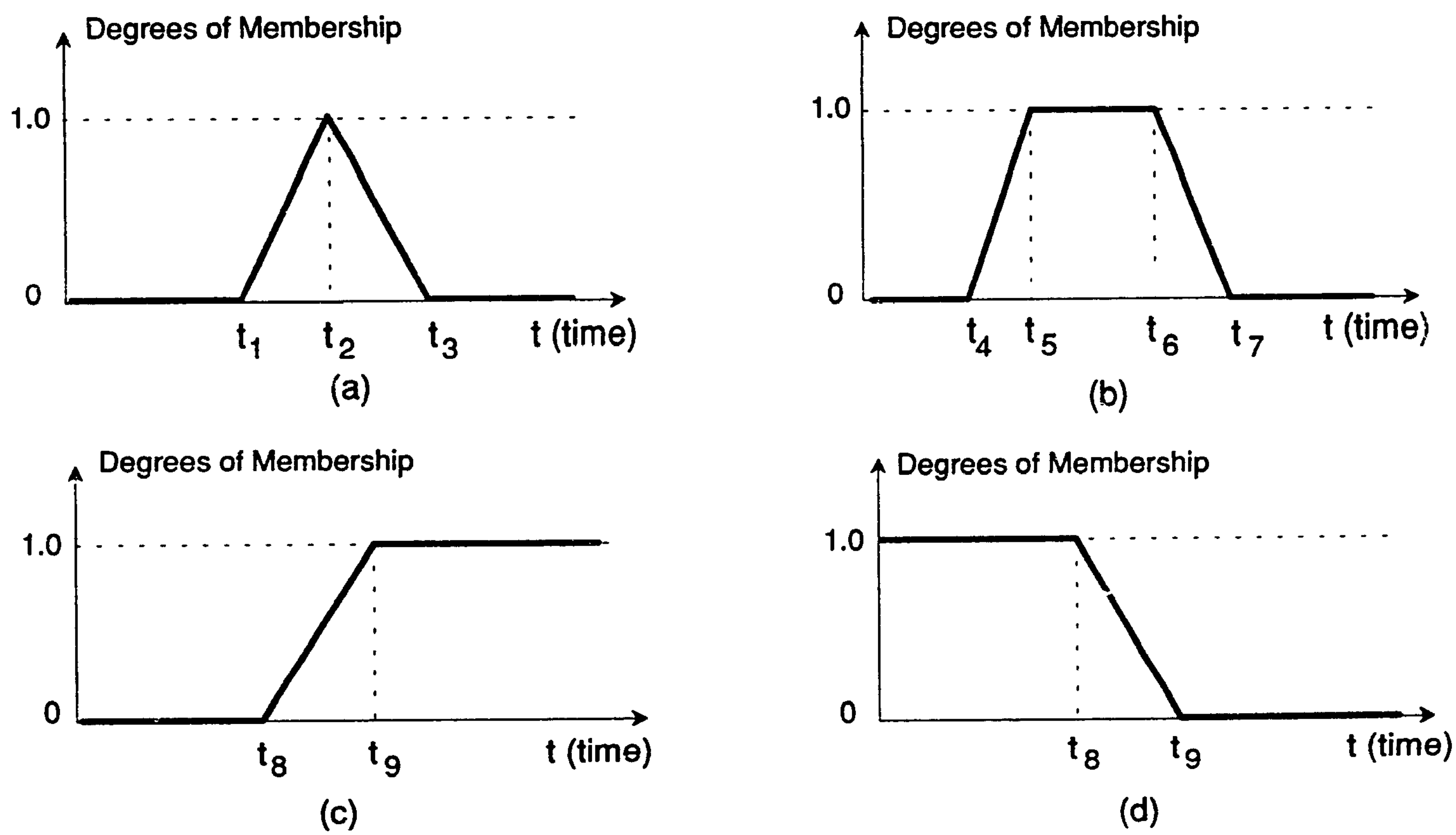


Fig. 4.3: Fuzzy Times.

triangular distribution shown in Fig. 4.3a represents the situation in which the time in question can assume a value anywhere between t_1 and t_3, with the most possible value being t_2. The membership function for this triangular distribution is:

$$\mu(t) = \begin{cases} \dfrac{t - t_1}{t_2 - t_1} & t_1 \le t < t_2 \\ 1 & t = t_2 \\ \dfrac{t - t_3}{t_2 - t_3} & t_2 < t \le t_3 \\ 0 & \text{otherwise} \end{cases} \tag{4.19}$$

Fig. 4.3b depicts the trapezoidal distribution which represents a more imprecise situation. The most possible values, this time, range between t_5 and t_6, and the degrees of possibility gradually reduce on both sides to the zero at t_4 and t_7, respectively. Any time value which is smaller than t_4 or greater than t_7, by definition, is not possible. The membership function for this trapezoidal distribution is:

$$\mu(t) = \begin{cases} \dfrac{t - t_4}{t_5 - t_4} & t_4 \le t < t_5 \\ 1 & t_5 \le t \le t_6 \\ \dfrac{t - t_7}{t_7 - t_6} & t_6 < t \le t_7 \\ 0 & \text{otherwise} \end{cases} \tag{4.20}$$

Fig. 4.3c depicts the time that occurs certainly after t_8, and, most plausibly after t_9. The membership function of this fuzzy bound is:

$$\mu(t) = \begin{cases} 0 & t \le t_8 \\ \dfrac{t - t_9}{t_9 - t_8} & t_8 \le t < t_9 \\ 1 & t \ge t_9 \end{cases} \qquad (4.21)$$

Fig. 4.3d, on the other hand, represents the time that most possibly occurs before t_8, but, certainly before t_9. The membership function os this case is:

$$\mu(t) = \begin{cases} 1 & t < t_8 \\ \dfrac{t - t_9}{t_8 - t_9} & t_8 \le t < t_9 \\ 0 & t \ge t_9 \end{cases} \qquad (4.22)$$

The distributions shown in Figs. 4.3c and 4.3d are also known as fuzzy bounds, where fuzzy lower bound represents the "after" case, and fuzzy upper bound symbolizes the "before" case.

4.4 Unconstrained Fuzzy Network Analysis

Previous work on network scheduling using fuzzy sets theory, provides methods for determining expected fuzzy early times of each event as well as project completion time (Chanas and Kamburowski 1981, McCahon 1987, Dubois and Prade 1988). These methods, however, do not support backward pass calculations in a direct manner similar to that used in the forward pass. This is mainly due to the fact that fuzzy subtraction is not the inverse of fuzzy addition (McCahon 1993). Also, the spread (i.e. degree of imprecision) of a fuzzy number resulting from a number of fuzzy arithmetic operations increases with the number of operations. The degree of imprecision associated with the calculated project completion time is the accumulation of the imprecision that exists in the activities involved.

This problem occurs in the backward pass when the durations of all activities are, once again, used to calculate their respective fuzzy late times. This process creates unrealistic accumulations of uncertainty. As a result, fuzzy late times of project activities derived from these calculations are not meaningful, and therefore can not be used as a basis for scheduling decisions. Dubois and Prade (1988), instead, suggest that the time to be used as a starting point in the backward pass calculations should be established independently from the forward pass. Following their approach, earlier activities would end-up being more uncertain than their successors. Still, the problem of unrealistic uncertainty in the network remains unresolved. To overcome this limitation, a new method called fuzzy network scheduling (FNET) is developed. The proposed method provides a complete fuzzy network analysis. FNET incorporates a number of interesting techniques which enable meaningful backward pass calculations, and practical interpretations of the results generated.

4.4.1 Fuzzy Forward Pass Calculations

The steps involved in the forward pass calculation in FNET are similar to that performed in PDM. The only difference is that the fuzzy operations introduced earlier are used instead of the conventional addition and maximum operations. The following equations are used to compute the fuzzy-early times.

$$FES_x = \underset{p \in P}{\tilde{max}}(FEF_p) \qquad (4.23)$$

$$FEF_x = FES_x \oplus FD_x \qquad (4.24)$$

$$T_{proj} = FEF_{ex} \qquad (4.25)$$

where $\mathbf{FES_x}$ is the fuzzy early start of activity **x**; **p** is the predecessor of **x**; **P** is the set of predecessors; **FEF** is the fuzzy early finish; **Fd** is the fuzzy duration; $\mathbf{T_{proj}}$ is the project completion time, and **e** is the last activity of the project.

It should be noted that the $\tilde{max}$ operation used in Eq. 4.23. is the one defined in Eq. 4.10. This operation performs pair-wise comparisons for each and every elements of the two quadruples, and accordingly selects the maximum value of each pair to represent their respective elements in the new quadruple. In other words, FNET accounts for both, the means and the spread of the two events, when making a comparison. This procedure is different from that of PERT where only the expected time values are compared. No consideration has been given to the standard deviation. Therefore, in the presence of multi-critical paths, FNET clearly produces a more realistic solution than that of PERT.

The forward pass calculations are completed when the early times for all activities are determined. Backward pass calculations can then begin by first assigning the project duration to a preliminary late finish of the last activity. This number is then used as a basis for the determination of fuzzy bounds, described in the following subsection.

4.4.2 Fuzzy Backward Path Calculations

As mentioned earlier, backward pass calculations using the traditional fuzzy subtraction (Eq.4.9) produces unrealistically large uncertainties associated with activity's fuzzy late times. This accumulates rapidly as the backward pass calculations continue to progress. Upon completion of the backward pass calculations, some of the earlier activities could end-up having negative fuzzy late times, which, practically, has no meaning. To circumvent such a drawback, a new technique for backward pass calculations is developed. This technique is based on an assumption that fuzzy late times (i.e. start and finish) are at least as uncertain as their respective fuzzy early times. In the PDM-based project network, uncertainties associated with activity durations directly influence the degree of uncertainties associated with the calculated start and finish times. The uncertainty associated with start times (i.e. early and late) is the result of the accumulation of uncertainties encountered in the durations of its predecessors. The uncertainty associated with the finish times, on the other hand, is that of the early time plus the uncertainty inherent in the duration of activity itself.

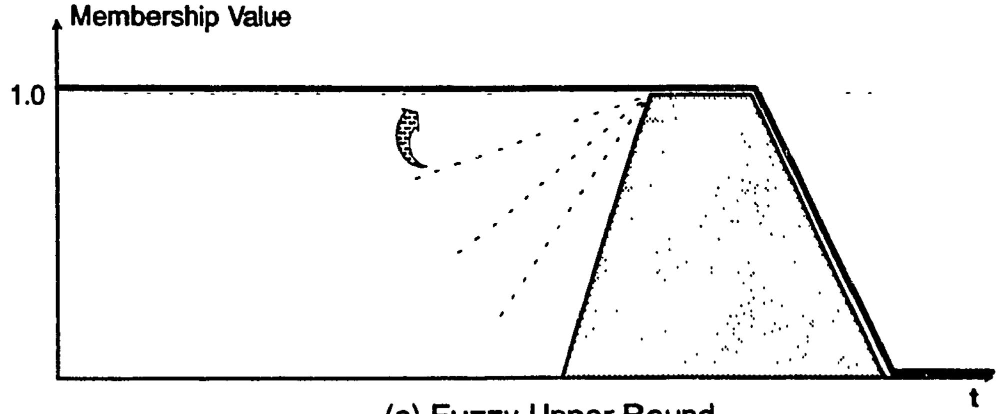
The above assumption can be justified by considering a linear project network as an example. The fact that the network has only one path, the early and late start times of each activity obtained from backward and forward pass calculations must be identical and equally uncertain. This is also true for the early and late finish times. In most cases, however, project networks contain several paths. In the backward pass calculations, higher uncertainties may be accumulated from a different path(s) than that used in the calculation of the early times. The above assumption, therefore, allows the late time to be more uncertain than its respective early times. This adds more flexibility and more accuracy for calculating fuzzy late start and finish times of project activities.

In general, realistic fuzzy results must adhere to the following conditions: 1) scheduling constraints generated from the calculations of forward and backward passes must be satisfied, 2) no negative quadruple for the activity late times is permitted, 3) the value of each element in the calculated quadruple does not exceed its successor (i.e. for a quadruple (**a,b,c,d**); $\mathbf{a} \leq \mathbf{b} \leq \mathbf{c} \leq \mathbf{d}$). In FNET, fuzzy bounds are employed to facilitate the calculations, satisfying these three conditions simultaneously. The use of fuzzy bounds is justified due to its flexibility with regard to the greater allowable time-window for calculating fuzzy late times. Further, the use of this time-window does not cause any violation to the constraints produced from both forward and backward pass calculations. For example, fuzzy late times of an activity indicate that this activity can not start or finish later than these times. Modifying fuzzy late times by extending their lower boundary (i.e. the left leg of that fuzzy number, see Fig. 4.4a) to an infinity clearly does not cause any violation to the original fuzzy late time. The extended fuzzy late time is called a fuzzy upper bound of the late time. Similarly fuzzy lower bounds can be used without violating the constraints generated from the forward pass calculations (see Fig. 4.4b).

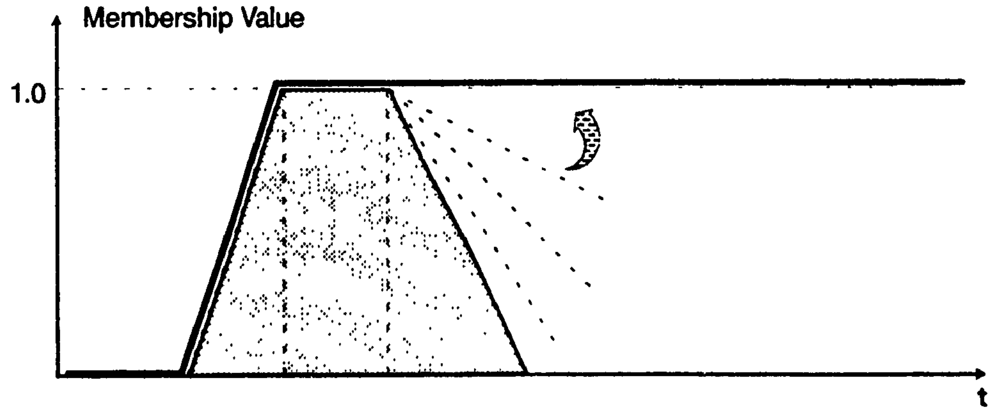
The conversion of a fuzzy number to its upper and lower bounds is simple and direct. Let **A** be a trapezoidal fuzzy number characterized by the quadruple (**a,b,c,d**). Its upper bound denoted by $\mathbf{A}^u$ can be obtained by subtracting an interval $[0,\infty)$, expressed in the form of the quadruple as $(\mathbf{0},\mathbf{0},\infty,\infty)$ from **A** as follows:

$$\mathbf{A}^{u} = (\mathbf{a},\mathbf{b},\mathbf{c},\mathbf{d}) \ominus (\mathbf{0},\mathbf{0},\infty,\infty) = (-\infty,-\infty,\mathbf{c},\mathbf{d}) \tag{4.26}$$

Fuzzy lower bound of **A** represented by $\mathbf{A}^l$ can be obtained by adding the quadruple $(\mathbf{0},\mathbf{0},\infty,\infty)$ to **A** yielding:



(a) Fuzzy Upper Bound



(b) Fuzzy Lower Bound

Fig. 4.4: Fuzzy bounds.

$$A^{l} = (a,b,c,d) \oplus (0,0,\infty,\infty) = (a,b,-\infty,-\infty) \tag{4.27}$$

Fuzzy upper and lower bounds are used to facilitate the calculations of fuzzy late times performed in the backward pass. The computations of fuzzy late times of each activity begin with the determination of its preliminary late finish (**PLF**). The definition of **PLF** is analogous to the late finish time in the PDM. The **PLF** can be calculated for any activity **x** as:

$$PLF_x = \widetilde{\min_{s \in S}}(FLS_s) \tag{4.28}$$

where FLS_s is the fuzzy late start time of its successor **s**; **S** is the set of the successors of activity **x**.

The **PLF** is then converted to the upper-bound for the late finish (FLF^u) using Eq. 4.26. Based on the assumption previously stated and the calculated FLF^u and **FEF**, it is possible to compute the fuzzy late finish time (**FLF**). Suppose that the **FEF** is characterized by (**a**,**b**,**c**,**d**), and the FLF^u is represented by (-∞,-∞,**e**,**f**). The 4-step procedure described below is used to calculate fuzzy late times for project activities.

1) Determine which of the two fuzzy quantities has a greater uncertainty (i.e. the larger right spread), by comparing the results of (**f** - **e**) and (**d** - **c**).

2) Calculate **Y**, the largest fuzzy number that satisfies the following condition.

$$\mathbf{FEF} \oplus \mathbf{Y} \subseteq \mathbf{FLF}^{u} \tag{4.29}$$

If **FEF** is more uncertain [i.e. (**d** - **c**) ≥ (**f** - **e**)], the uncertainty of **FLF** is set equal to that of the **FEF** (see Fig. 4.5a). In this case, all the quadruple of **Y** are equal to the difference between the upper bounds elements of the two fuzzy quantities as follows:

$$\mathbf{Y} = (\mathbf{f}-\mathbf{d},\ \mathbf{f}-\mathbf{d},\ \mathbf{f}-\mathbf{d},\ \mathbf{f}-\mathbf{d}) \tag{4.30}$$

On the other hand, If $\mathbf{FLF}^{u}$ is more uncertain, the uncertainty of **FLF** is the combination of those associated with the **FEF** and $\mathbf{FLF}^{u}$ (see Fig. 4.5b). That is, the left spread and the plateau takes their values from those of **FEF**, and the right spread is set equal to that of the $\mathbf{FLF}^{u}$. In this case, **Y** is calculated by:

$$\mathbf{Y} = (\mathbf{e}-\mathbf{c},\ \mathbf{e}-\mathbf{c},\ \mathbf{e}-\mathbf{c},\ \mathbf{f}-\mathbf{d}) \tag{4.31}$$

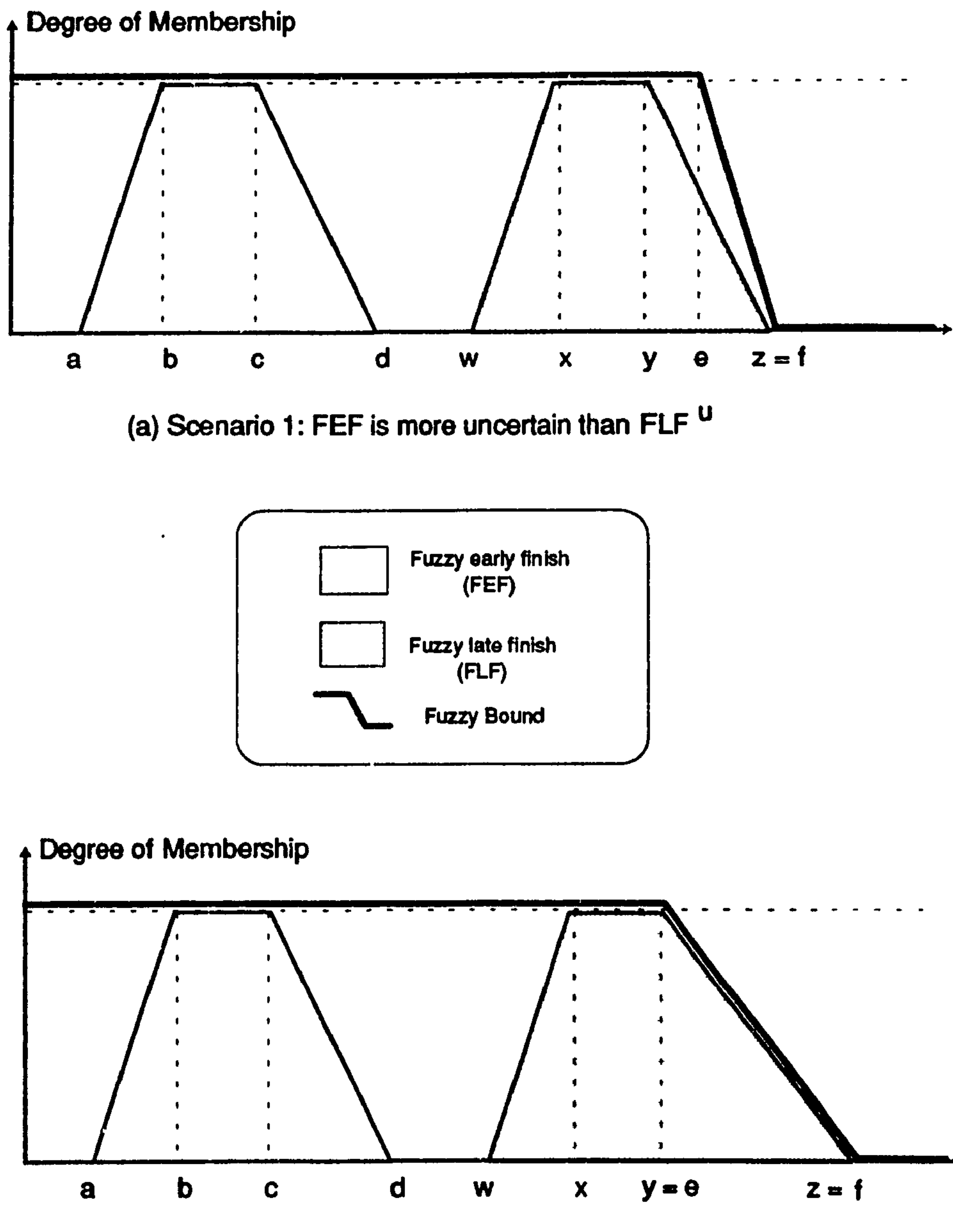
3) The **FLF** of project activities can then be calculated as follows:

$$\mathbf{FLF} = \mathbf{FEF} \oplus \mathbf{Y} \tag{4.32}$$

4) Fuzzy late start (**FLS**) can then be computed by substituting **FLF** and **FD** into the following equation.

$$\mathbf{FLS} \oplus \mathbf{FD} = \mathbf{FLF} \tag{4.33}$$

In addition, fuzzy bounds can also be used to calculate the fuzzy start (**FS**) and



(a) Scenario 1: FEF is more uncertain than FLF ^{u}

(b) Scenario 2: FEF is less uncertain than FLF ^{u}

Fig. 4.5: Graphical Presentation of Fuzzy Late Times.

fuzzy finish (FF) for each activity. Fuzzy start and finish times indicate, respectively, the ranges of possible time elements that each activity is expected to start and finish. Fuzzy start and finish times for activity x can be calculated as follows:

$$FS_x = FES_x^l \cap FLS_x^u \tag{4.34}$$

$$FF_x = FEF_x^l \cap FLF_x^u \tag{4.35}$$

4.5 Compatibility between Fuzzy Events

In addition to the common scheduling data such as: start and finish times, etc., project attributes such as: 1) expected project completion time, 2) degree of criticality for project activities, 3) possibility of meeting specified intermediate milestones, and, 4) possibility of violating project constraints, are also of interest to the project team. These four attributes can be used in conjunction with the scheduling data obtained to provide the project team with the necessary information for assessing the acceptability of the schedule. Two concepts namely: 1) the possibility measure (Zadeh 1978) and 2) the agreement index (Kaufmann and Gupta 1985) are used to perform such assessments. A brief description of these two concepts is given below.

4.5.1 Possibility Measure

Possibility measure (PM) is the concept introduced by Zadeh (1978) to evaluate the degree of belonging of a fuzzy number to another. In the scheduling context, these two numbers usually are the calculated and the expected events. While the calculated event is fuzzy, the expected event could either be fuzzy (e.g. a switch-gear will arrive on site "by the mid-July") or nonfuzzy (i.e. project completion date is 19 May, 1995). Suppose that **T** is a universe of discourse, **X** is the calculated fuzzy number representing the set of possible times of a project event which takes values in **T**. The possibility distribution function associated with **X** is defined to be numerically equal to the membership value of **X** [i.e. $\mu_x(t)$]. Considering, first, the case where the expected event **F** is fuzzy. The possibility measure of **F** is defined as (Zadeh 1978):

$$\text{Poss } \{X \text{ is } F\} = \bigvee_{t \in T} [\mu_F(t) \wedge \mu_x(t)] \tag{4.36}$$

As for the case where **N** is a nonfuzzy event, the possibility measure of **N** is defined as (Zadeh 1978):

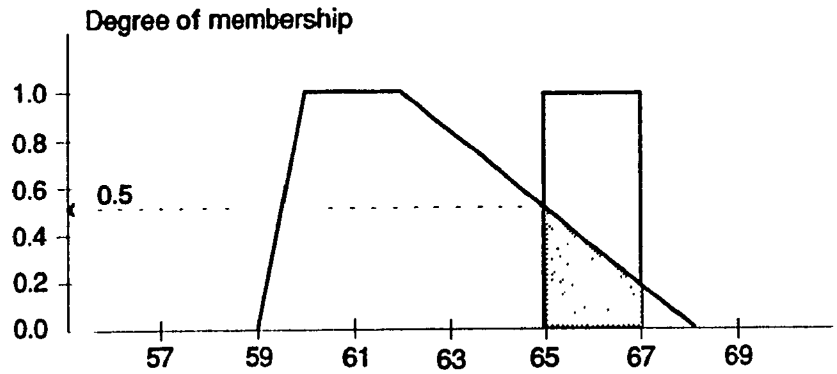
$$\text{Poss } \{X \in N\} = \bigvee_{t \in N} \mu_x(t) \tag{4.37}$$

In the case where the two events are described by the continuous piece-wise linear functions, PM is the height of the intersection area of two events.

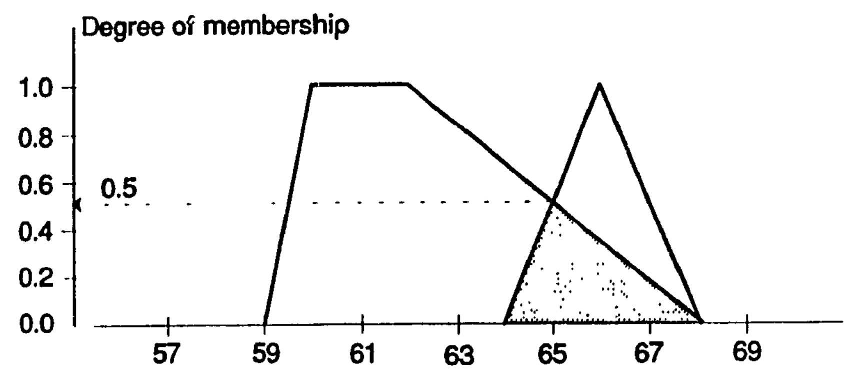
$$\text{Poss}\{X \in N\} = \text{Sup}_{u \in N} \mu_x(t) \tag{4.38}$$

For illustration purposes, let us assume that the duration of a project network was calculated to be (59,60,62,68). Suppose that one would like to know the possibility that the expected project duration would agree with the concepts (i.e. constraints or expectations) such as: *between 65 and 67 days, about 65 days,* and *exactly 64 days.* These fuzzy events are expressed by the three quadruples: (65,65,67,67), (64,66,66,68), and (64,64,64,64) and are demonstrated in Figs. 4.6 a and b respectively. In order to assess the compatibility between any two events, the intersection range and the elements which belong to that range must first be identified. For each identified element, its new degree of membership assumes the value from the smaller of the two original degrees of membership. The largest degree of membership among those elements signifies the possibility measure of the two events being considered. Graphically, the possibility measure of any two events that have continuous possibility distribution functions is the highest point of the intersection area of the two events (see also Fig. 4.6).

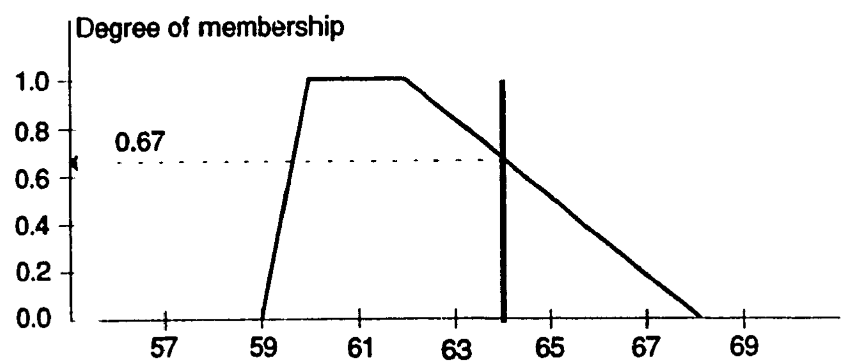
With regard to the first case, Fig. 4.6a, the elements that are included in the intersection range along with their associated degree of membership are: {65 | 0.50, 66 | 0.33, 67 | 0.17}. Applying Eq.4.37, the possibility that the project duration being *between 65 and 67 days* is 0.5. As for the second case, the assessment is made between two fuzzy events. The elements that fall into the intersection range and their degrees of membership include {65 | 0.50, 66 | 0.33, 67 | 0.17}. Therefore, the possibility of having the project duration *about 65 days* is 0.5 (see Fig. 4.6b) which is identical to that of the first case. With respect to the third case *exactly 64 days,* there exist only one element in the intersection range (i.e. t = 64). The smallest



(a) Between fuzzy and crisp events



(b) Between two fuzzy events



(c) Between fuzzy and exact events

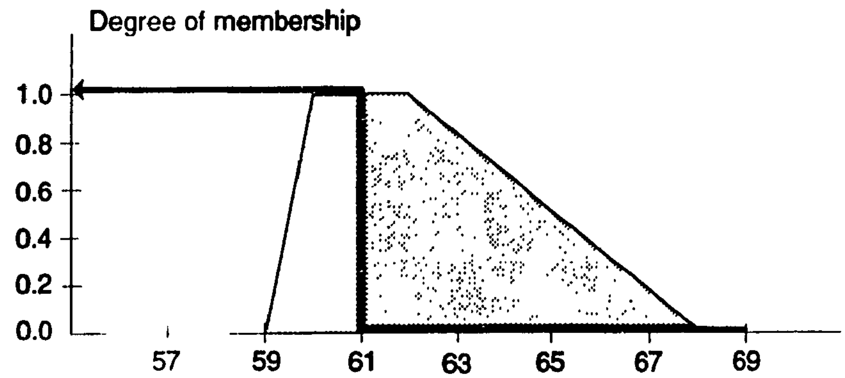
Fig. 4.6: Possibility Measure of Scheduling Events.

degree of membership of t = 64 is 0.67. According to the possibility measures, it can be stated that the third expectation, *exactly 64 days*, is the most possible one.

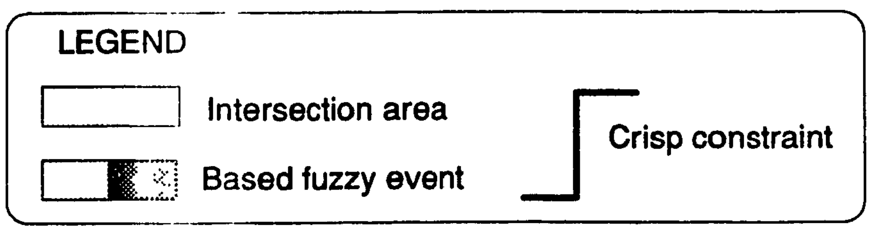
4.5.2 Agreement Index

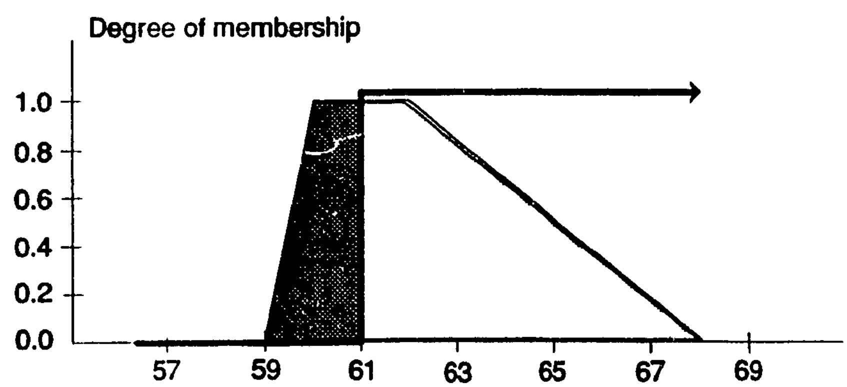
The possibility measure described in the previous section takes its value from the maximum membership function of the intersection area membership of the common elements) of the two events involved. No consideration, however, has been given to the size of that intersection area. In fact, it is possible that the value of possibility measure is high (i.e. close to 1) while the intersection area is small. Considering the two examples illustrated in Figs. 4.7 a and b, the **PM** of these two scenarios are identical. It can be seen, however, that the two events depicted in Fig. 4.7b are more in an agreement to each other than those shown in Fig. 4.7a. Clearly, the possibility measure, in this circumstance, does not provide an insightful assessment of the compatibility between the two fuzzy events. The agreement index introduced by Kaufmann and Gupta (1985) can be used as a compliment to the possibility measure. This index measures the ratio of the intersection area between two fuzzy events (i.e. the calculated and expected events) with respect to the area of the calculated event. More specifically, let **A** and **H** be any two events being considered. The agreement index of **A** with respect to **H**, **AI(A,H)** is defined as follows:

$$AI(A,H) = \frac{(\text{area } A \cap H)}{\text{area } A} \tag{4.39}$$



(a) Definitely before day 61





(b) Definitely after day 61

Fig. 4.7: Assessment of Constraint Satisfaction.

Analytically, the area of intersection can be determined from partial integration. Given the four numerical values for the quadruple of a trapezoidal fuzzy number, piece-wise linear functions for its boundaries can be derived (see Fig. 4.1a). The piece-wise linear functions of the two fuzzy events that form an intersection area are used for the determination of the boundaries and the corner points of that intersection area. A major short coming of the agreement index is that it cannot be used to assess the single-valued type event.

Considering once again the case illustrated in Fig. 4.7a, the intersection area is:

$$\int_{65}^{67} y\,dt = \int_{65}^{67} \left(-\frac{t}{6} + \frac{34}{3}\right) dt = \left[-\frac{t^2}{12} + \frac{34t}{3}\right]_{65}^{67} \tag{4.40}$$

The intersection area is 0.67 units. The area of the based event (i.e. the calculated project duration, (59,60,62,68)) is calculated to be 5.5 units. The agreement index is calculated using Eq.(4.39) to be 0.12. Similarly, the intersection area for the case shown in Fig. 4.7b, is:

$$\int_{64}^{68} y\,dt = \int_{64}^{65} \left(\frac{t}{2} - 32\right) dt + \int_{65}^{68} \left(\frac{-t}{6} + \frac{34}{3}\right) dt = \left[\frac{t^2}{4} - 32t\right]_{64}^{65} + \left[-\frac{t^2}{12} + \frac{34t}{3}\right]_{65}^{68} \tag{4.41}$$

The intersection area for the latter case is 1.0 unit, and the agreement index is

0.18. Despite the identical possibility measure, it can be said that the project duration agrees more with the concept labelled *about 65 days* than with the concept *between 65 and 67 days.*

The use of the agreement index is more pronounced for the assessment involving crisp events such as those demonstrated in Fig. 4.7. It can be seen that the possibility of meeting the two expected events: *finishing the project before day 61* and *finishing the project after day 61* are identically high (i.e. Possibility measures equal to 1.0). Further analysis using the agreement index reveals that the agreement index of the *before* case is 0.27, while that of the *after* case is 0.73. It can be stated that though, these two events appear to be highly possible, the latter one is more likely to occur.

The agreement index also has some limitations. For example, the index is not meaningful in the situation where any of the two events being examined is definitive (i.e the intersection area is zero). The agreement index also lacks an important characteristic called "the valuation" which can be given by the possibility measure (Kaufmann and Gupta 1985). It can be viewed that the possibility measure and the agreement index do compliment each other. As such, the two parameters should be used for assessing the compatibility of project events.

4.6 Criticality Measurement

When there are uncertainties associated with the durations of project activities, the determination of the degree of criticality for each project activity is not as obvious as in the case where the durations are definitive. Two techniques are used in this study to identify critical activities. The first technique involves the determination of the degrees of criticality using the possibility measure and the agreement index described in the previous section. The second technique is used to determine the approximate total float of each activity. The details of these two techniques are discussed in the following subsections.

4.6.1 Degree of Criticality

The following steps are used to determine the degree of criticality for each activity.

1) Calculate the duration for each and every path in the network as follows: where **k** is the activity that belongs to path **X**.

$$T_X = \sum_{k \in X} d_k \tag{4.42}$$

2) Determine the possibility measure of the duration of each path (**PM_X**) pertaining to fuzzy project duration using Eq.4.43.

$$PM_X = Poss\{T_X \text{ is } T_{proj}\} = Sup[\mu_{T_X}(t) \wedge \mu_{T_{proj}}(t)] \tag{4.43}$$

3) Calculate the agreement index for each path (**AI_X**) with the project duration:

$$AI_X(T_X, T_{proj}) = \frac{(AreaT_X \cap T_{proj})}{AreaT_X} \qquad (4.44)$$

4) Activities that belong to only one path assume their criticality from that path. As for the activities are common to several paths, their criticality indicators are set equal to take the largest possibility measure and the agreement index of the paths involved.

Generally, **PM** and **AI** for a given network path are proportionate to each other. It may appear that **PM** alone is quite adequate for identifying the critical path(s). The **PM**, by definition, provides only the possibility value for a single time element included in the fuzzy path duration being considered. This value does not give the overall perspective on how close the agreement between the entire fuzzy path duration and the fuzzy project duration. The **AI**, on the other hand, measures overall agreements between two fuzzy events. These two parameters therefore compliment each other.

4.6.2 Approximate Total Float

In addition to the degree of criticality, it is also possible to calculate the approximate activity's total float. In this study, the total float is computed from the difference between the geometric centroids of fuzzy early and late start times or between that of the late times. The geometric centroid (**C**) of a trapezoidal fuzzy number is calculated as follows:

$$C = \frac{(-a^2 - b^2 + c^2 + d^2 - ab + cd)}{3(-a - b + c + d)} \qquad (4.45)$$

where **a**, **b**, **c**, and **d** are the numerical values in the quadruple of a trapezoidal fuzzy numbers.

The total float of activity **x** is the difference between the centroid of fuzzy late times (i.e. start or finish) and that of their respective fuzzy early times, and can be computed as follows:

$$TF_x = C_x^{FLS} - C_x^{FES} \qquad (4.46)$$

or,

$$TF_x = C_x^{FLF} - C_x^{FEF} \qquad (4.47)$$

where C_x signifies the geometric centroid of each fuzzy time of activity **x**.

4.6.3 Composite Criticality Index

Another useful information which can be derived from the **PM** and **AI** values of each activity is the cumulative degree of criticality. The cumulative degree of criticality of an activity is the summation of each **PM** and **AI** values for all the paths to which it belongs. Generally, activities that are common to several paths in a network tend to have a higher cumulative degree of criticality than those common to smaller number of paths. The values of these parameters though has no absolute physical meaning, they are meaningful in relative context. These parameters indicate the degree of dependence and the potential for creating a bottle-neck that can affect the project schedule. As such, those activities with higher cumulative criticality values should draw the attentions of the project team. Any slippage on these activities can negatively affects the project completion time.

4.7 Fuzzy Resource-Constrained Scheduling

The FNET method described in the previous section accounts only for precedence constraints. This section extends the application of FNET to resource constrained scheduling. Fuzzy resource scheduling basically involves: 1) calculating resource demands, 2) assessing resource availabilities, 3) determining whether there is a resource conflict, and 4) allocating resources in such a way as to resolve arisen conflicts and satisfy stated project objectives (e.g. minimum project duration, maximum resource utilization, etc.). Traditional fuzzy operations have been proven inadequate to perform these four functions. The following subsections describe the methods needed for resource-constrained scheduling. These include the introduction for the concept of fuzzy resource quantities and its associated arithmetic operations.

4.7.1 Fuzzy Resource Quantity (FRQ)

In practice, the type and number of resources (i.e. standard crew compositions) employed to perform certain construction tasks are usually known. What might not be definitely known is the availability for each type of resource for a given period of time. This is because resources are usually shared among several projects. The availability of resources in the pool depends on the consumptions of these resources by these projects. If the durations of activities in the projects being considered are not certain, the demands for resources and consequently the availabilities in the pool are also uncertain. To model the demands and availabilities of resources in such an environment, a fuzzy resource quantity (FRQ) expressed in a couplet (quantity; degrees of availability or use) is introduced. The FRQ for resource **j** during $\mathbf{n}^{th}$ period ($\mathbf{R}_{j,n}$) is defined as:

$$R_{j,n} = [Q_k; F_k]_{j,n} \tag{4.48}$$

where $\mathbf{Q}_k$ is the quantity of resource **j** needed or released at time interval **n**; $\boldsymbol{F}_k$ are the degrees to which resource $\mathbf{Q}_k$ are either needed (from the project perspective) or available (from the resource pool perspective) at time interval **n**; **k** is an indicator used to differentiate resources based on either their quantities or degrees of availability or demand.

In general, $\boldsymbol{F}_k$ can be a trapezoidal fuzzy number expressed in the quadruple form **(a,b,c,d)**, or by a membership function of time (**t**) expressed by $\mathbf{f}_k(\mathbf{t})$ where **t** is the time element included in the $\mathbf{n}^{th}$ period. If $\boldsymbol{F}_k$ is represented by a quadruple (a,b,c,d) for example, the degree of need or availability will equal to 1.0 if the time being considered is between **b** and **c**. Such degrees would be less than 1.0 if **t** lies between **a** and **b** or between **c** and **d**, and would be 0.0 otherwise. Consider a statement such as 5 welders from project XYZ will be released to the resource pool approximately at the end of July. And that the expression "approximately at the end of July" is represented by a quadruple (Jul. 20, Jul. 25, Jul. 30, Aug. 4). According to (Eq. 4.48), the fuzzy quantity describing this resources of welders can be expressed as: $[5, (\text{Jul. } 20, \text{Jul. } 25, \text{Jul. } 30, \text{Aug. } 4)]_{\text{welders, end of July}}$.

Note that the membership function for the degree of demand or availability need not be a fuzzy number. In other words, the highest degree of membership can have a value less than 1.0. These degrees of membership can then be calculated from fuzzy operations described in the following subsection.

4.7.2 Arithmetic Operations on FRQs

Two basic operations that are used for the computations of FRQs are fuzzy addition and subtraction. Since FRQs are different from conventional fuzzy numbers, traditional fuzzy addition and subtraction are not applicable. As such there is a need to develop new fuzzy addition and subtraction techniques. Let $\mathbf{R}_{j1,n}$ and $\mathbf{R}_{j2,n}$ be any two fuzzy quantities for resource **j** during time interval **n**, represented by $[Q_1, F_1]_{j,n}$ and $[Q_2, F_2]_{j,n}$ respectively. The addition of $R_{j1,n}$ and $R_{j2,n}$ yields a new quantity $\mathbf{R}_{j3,n}$ defined as:

$$R_{j3,n} = [(Q_1 + Q_2),\ F_3]_{j,n} \quad \text{where } F_3 = F_1 \cap F_2 \tag{4.49}$$

While the addition of Q_1 and Q_2 in the above equation is apparent in describing the total quantity it is interesting to note that the membership function of $R_{j3,n}$, (F_3), results from applying the intersection operation (see also Eq. 4.12). The use of the intersection operation is logical since it compares and selects the smallest value from the two membership functions being considered (i.e. F_1 and F_2 in this case). Therefore, F_3 is always smaller or at most equal to F_1 and F_2. For this reason, after performing an addition, the two original FRQs (i.e. $R_{j1,n}$ and $R_{j2,n}$) must not be disregarded, to avoid any loss of useful information (i.e. the higher degrees of membership of the two original FRQs), that may compromise the quality of the schedule generated. This is described later towards the end of this subsection.

Fuzzy resource subtractions are more complicated than additions, since they require certain conditions to be met prior to the arithmetic operations. Let $R_{j1,n}$ be a fuzzy availability represented by $[Q_1, F_1]_{j,n}$ and $R_{j2,n}$ be an associated fuzzy demand represented by $[Q_2, F_2]_{j,n}$, the subtraction of $R_{j2,n}$ from $R_{j1,n}$ is feasible only when at any time interval **n**: 1) the quantity Q_1 is larger than Q_2 and 2) degree of availability F_1 exceeds that of demand F_2. In the conventional subtraction, the outcome indicates, quantitatively, the difference between two forerunners (i.e. Q_1 - Q_2 in this case). Due to the uncertainty in resource demands and availabilities, there is a possibility that the original quantity (i.e. Q_1) will still be available after the subtraction. As such, it is necessary to define two FRQs to represent the outcome. While the first FRQ describes the degree of availability for the quantity (Q_1 - Q_2), the second FRQ describes the degree of availability for the original amount, $\mathbf{Q}_1$. Although it is apparent that the new degrees of availability of $\mathbf{Q}_1$ after subtraction are smaller than those of the original resource pool, they could be of significance in certain circumstances, and therefore should not be neglected. These two FRQs are defined as:

$$R_{j4,n} = [(Q_1 - Q_2)\,;\,F_4] \qquad \text{where } F_4 = F_1 \cap F_2 \tag{4.50}$$

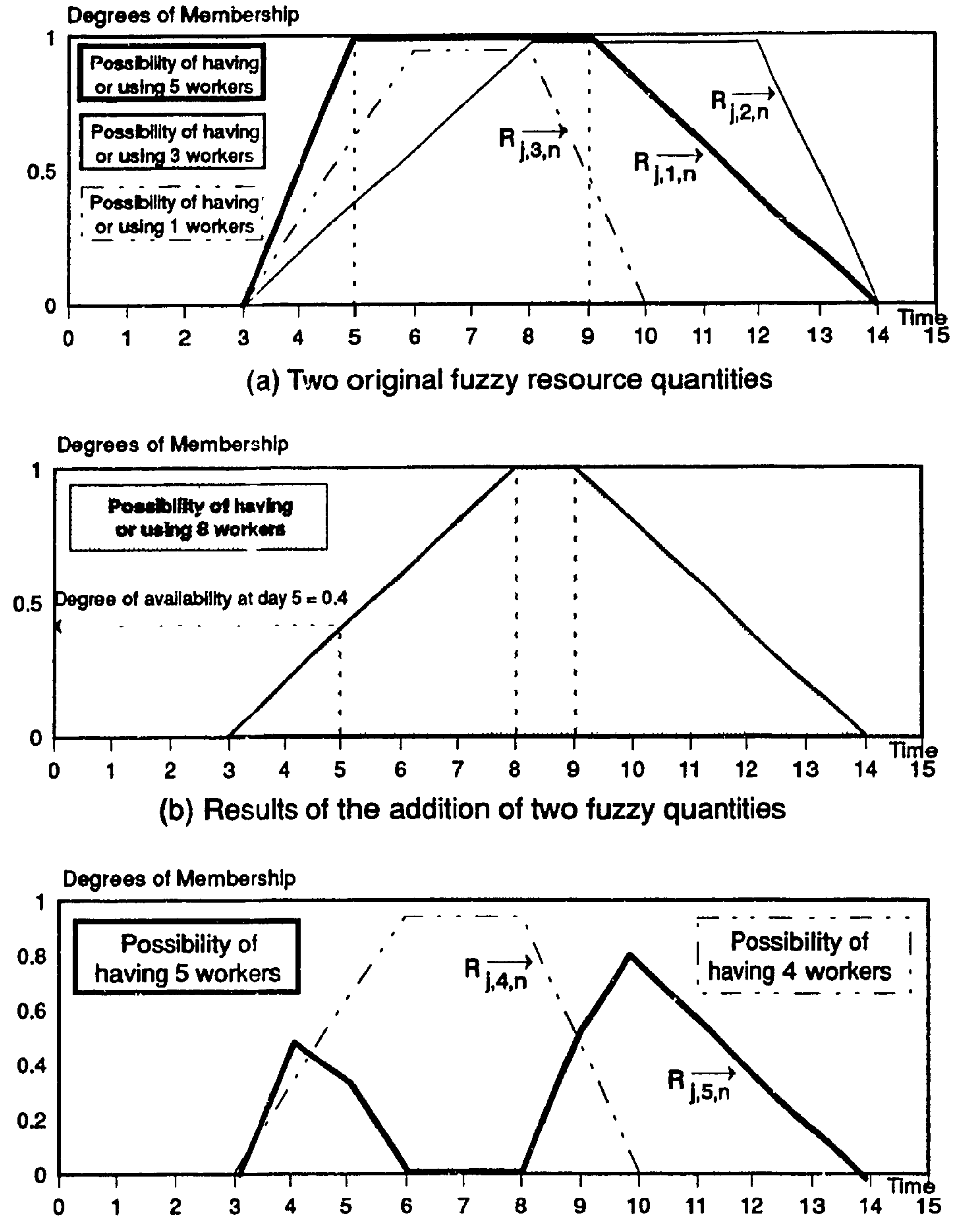
and,

$$R_{j5,n} = [Q_1\,;\,F_5]_{j,n} \qquad \textit{where: } F_5 = (F_1 \cap \neg F_2) \tag{4.51}$$

In fuzzy set theory, the intersection operator is used to determine the membership function that describes the simultaneous occurrence of any two fuzzy events. The most important task in performing the subtraction is therefore to identify the two events that occur concurrently. Considering Eq. 4.50, the quantity of $\mathbf{R}_{j4,n}$ takes its value from $\mathbf{Q}_1$ - $\mathbf{Q}_2$. This indicates that both $\mathbf{R}_{j1,n}$ and $\mathbf{R}_{j2,n}$ must take place simultaneously. Accordingly, the degrees of membership of the two FRQs (i.e. F_1 and F_2) can directly be used for calculating the degree of membership for $\mathbf{R}_{j4,n}$. In Eq. 4.51, on the other hand, the quantity of $\mathbf{R}_{j5,n}$ is set equal to that of $\mathbf{R}_{j1,n}$. This suggests that $\mathbf{R}_{j1,n}$ does not occur simultaneously with $\mathbf{R}_{j2,n}$ and may, on the contrary, be considered to occur with $\neg\mathbf{R}_{j2,n}$. As such, the degree of membership of $\neg F_2$ (i.e. 1 - F_2) and F_1 are used for computing F_5.

A numerical example is given to illustrate the use of these two fuzzy resource operations described above. Let $\mathbf{R}_{j1,n}$ = [5 workers, (3,5,9,14)], and $\mathbf{R}_{j2,n}$ = [3 workers, (3,8,12,14)], and $\mathbf{R}_{j3,n}$ = [1 workers, (3,5,7,10)] be three FRQs described graphically as shown in Fig. 4.8a. Find the results of: 1) the addition of $\mathbf{R}_{j1,n}$ and $\mathbf{R}_{j2,n}$, 2) the subtraction of $\mathbf{R}_{j2,n}$ from $\mathbf{R}_{j1,n}$, and 3) the subtraction of $\mathbf{R}_{j3,n}$ from $\mathbf{R}_{j1,n}$. Applying Eq. 4.49 for the addition of $\mathbf{R}_{j1,n}$ and $\mathbf{R}_{j2,n}$ yields a new quantity $\mathbf{R}_{j4,n}$ in which its quantity is the summation of $\mathbf{Q}_1$ and $\mathbf{Q}_2$ (i.e. 5 + 3 = 8 workers) and its degree of membership is calculated to be (3,8,9,14) (see Fig. 4.8b). Suppose that the project team would like to know the possibility of having 6 workers on day 5. Since the quantities of the two original FRQs are smaller than 6 workers (i.e. $\mathbf{Q}_1$ = 5 workers and $\mathbf{Q}_2$ = 3 workers), the degrees of availability of the newly calculated eight workers should be examined. This possibility is calculated to be 0.4 (see Fig. 4.8b). If, however, only 4 workers are required on that same day, the availability of 5 workers should alternatively be used. This is because it gives a higher degree of possibility (i.e. the possibility = 1.0, see Fig. 4.8a) than that of the 8 workers profile (i.e. the possibility = 0.4). From this example, it can be seen that if $\mathbf{R}_{j1,n}$ is disregarded, as in the case of crisp numbers, there will be a loss of useful information, and could possibly lead to a pessimistic result. This is the main reason for retaining the two original fuzzy quantities after performing fuzzy resource addition.

As for the subtraction, the two conditions described earlier must first be satisfied



(a) Two original fuzzy resource quantities

(b) Results of the addition of two fuzzy quantities

(c) Results of the subtraction between the two fuzzy quantities

Fig. 4.8: Operations On Fuzzy Resource Quantities.

prior to its operation. It can be seen that both $\mathbf{Q}_2$ (3 workers) and $\mathbf{Q}_3$ (1 workers) are smaller than $\mathbf{Q}_1$ (5 workers). As such, there is no violation with respect to the quantity. With regard to the degrees of membership, however, consider first the case of $\mathbf{R}_{j1,n}$ - $\mathbf{R}_{j2,n}$. It can be seen from Fig 4.8a that at **t** > 9 units, the degrees of demand of $\mathbf{R}_{j2,n}$ exceed their respective availabilities of $\mathbf{R}_{j1,n}$. Therefore, $\mathbf{R}_{j1,n}$ - $\mathbf{R}_{j2,n}$ is not practically feasible. Therefore It can be concluded that there is a resource conflict. In the case of $\mathbf{R}_{j1,n}$ - $\mathbf{R}_{j3,n}$, the degrees of the demand for $\mathbf{R}_{j3,n}$ are smaller than or equal to their respective availabilities of $\mathbf{R}_{j1,n}$ over the entire time interval **n**, rendering the subtraction feasible (see Fig 4.8a). As mentioned earlier, the subtraction of any two FRQs gives two new FRQs. According to Eqs. 4.50 and 4.51, the subtraction of $\mathbf{R}_{j3,n}$ from $\mathbf{R}_{j1,n}$ yields $\mathbf{R}_{j4,n}$ and $\mathbf{R}_{j5,n}$ in which their quantities are calculated to be 4 workers (i.e. $\mathbf{Q}_1$ - $\mathbf{Q}_3$,) and 5 workers (i.e. $\mathbf{Q}_1$) respectively. The degrees of availability for the resources 4 workers and 5 workers are shown in Fig. 4.8c. It can be seen from this figure that the degrees of availability for the 5 workers cannot be represented in a form of a quadruple.

4.7.3 Fuzzy Resource Allocation

This section describes a method developed for resolving resource conflicts, taking into consideration uncertainties inherent in the duration of project activities. The method utilizes the resource allocation algorithm presented in Chapter 3. The input of the proposed method including activity durations, early and late start times are fuzzy and can be obtained using FNET. Resource allocation basically involves: 1) identifying eligible activities, 2) checking whether there is a resource conflict(s), 3) if so, assigning priorities to the activities that are experiencing the conflict(s), and 4) allocating available resources to the conflicting activities according to the assigned priorities. Consider a typical resource conflict situation where four activities are to be scheduled. Fuzzy durations, fuzzy early start times, and the number of resources required by these activities are given in Table 4.1. Assuming that resources are obtained from various sources and that their availabilities are as depicted in Fig. 4.9a.

Table 4.1: Activity Data for Resource Allocation.

Activities	Fuzzy Durations	Fuzzy Late Start	Resources
A	(2,3,4,5)	(5,7,9,11)	8 workers
B	(2,4,4,6)	(4,6,8,10)	5 workers
C	(6,6,8,8)	(10,10,12,12)	4 workers
D	(3,5,7,9)	(3,5,7,9)	7 workers

Suppose that the current time is calculated to be (1,2,3,4) days. The first step is to investigate whether available resources are sufficient to schedule all of the four activities concurrently. An aggregation for the demand for resources of the four activities is performed using the addition operation described in the previous section (Eq. 4.49). The aggregation requires possible fuzzy time-periods needed to complete each activity. The fuzzy-period of an activity is taken as the times

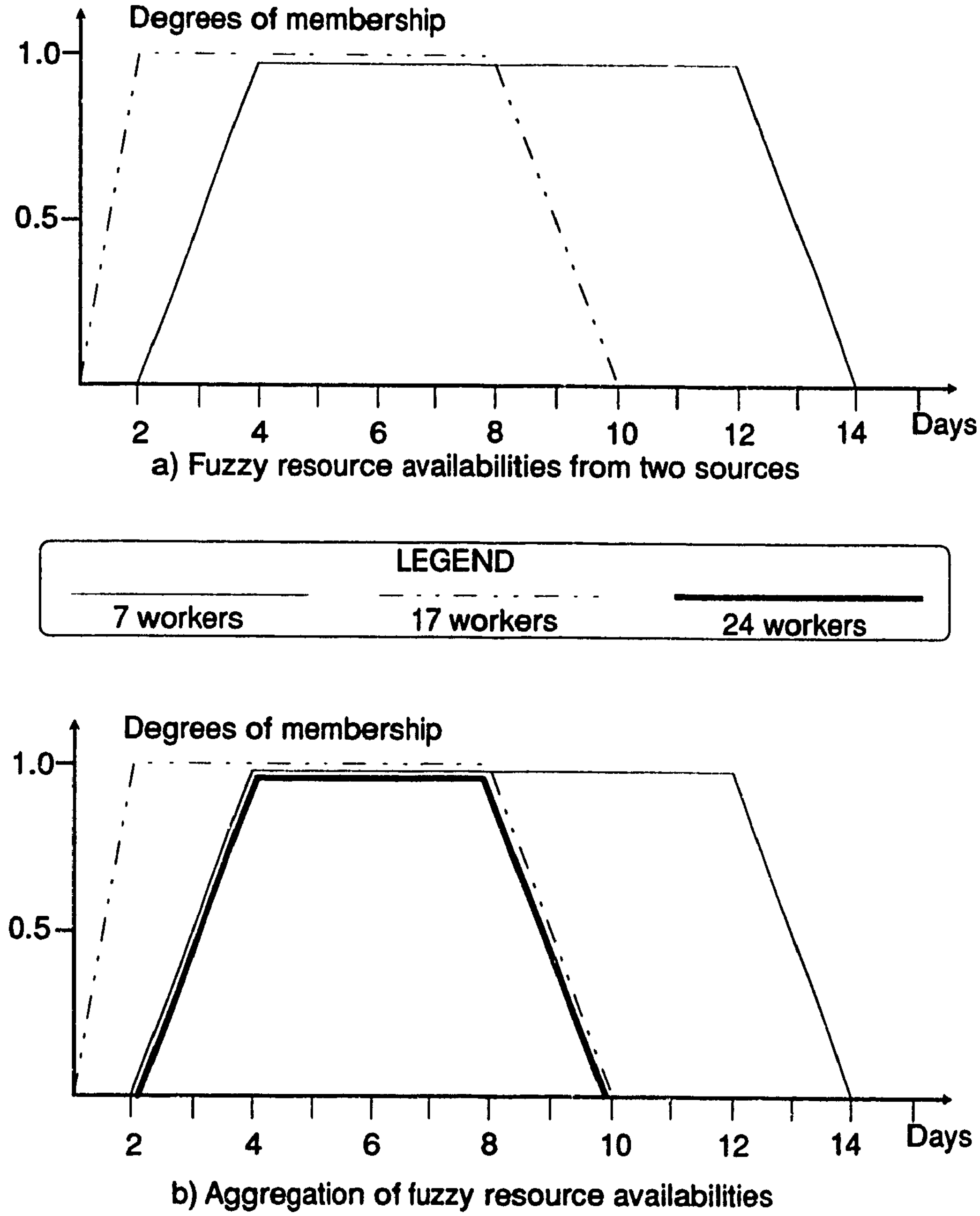


Fig. 4.9: Demonstration of Fuzzy Resource Availabilities.

between its fuzzy start **FS** (i.e. the current time in this case) and fuzzy finish times (**FF**). The calculated **FFs** for all four activities are given in Table 4.2. Once the **FS** and **FF** of each activity are obtained, they are used to calculate the associated $\mathbf{FS}^l$ and the $\mathbf{FF}^u$ using Eqs. 4.27 and 4.26 respectively. The fuzzy time interval required by each activity is the intersection between its $\mathbf{FS}^l$ and $\mathbf{FF}^u$. Accordingly, fuzzy resource demand of each activity are identical to its fuzzy time interval. The calculations of fuzzy resource demands for each of the four activities are summarized in Table 4.2. The results are illustrated in Fig. 4.10.

Table 4.2: Fuzzy Resource Demand.

Activities	Fuzzy Start	Fuzzy Finish	Fuzzy Start^L	Fuzzy Finish^U	Fuzzy Period
A	(1,2,3,4)	(3,5,7,9)	$(1,2,\infty,\infty)$	$(-\infty,-\infty,7,9)$	(1,2,7,9)
B	(1,2,3,4)	(3,6,7,10)	$(1,2,\infty,\infty)$	$(-\infty,-\infty,7,10)$	(1,2,7,10)
C	(1,2,3,4)	(7,8,11,12)	$(1,2,\infty,\infty)$	$(-\infty,-\infty,11,12)$	(1,2,11,12)
D	(1,2,3,4)	(7,10,12,14)	$(1,2,\infty,\infty)$	$(-\infty,-\infty,12,14)$	(1,2,12,14)

^{L} = Lower Bound; ^{U} = Upper Bound.

Resource aggregation is the summation of these fuzzy resource demands. Fig. 4.11 depicts examples of possible fuzzy-resource-demand aggregations for various activity subsets along with the fuzzy calculated availabilities. Consider, first, the set consisting of eligible activities A,B,C, and D, as depicted in Fig. 4.11a. There are

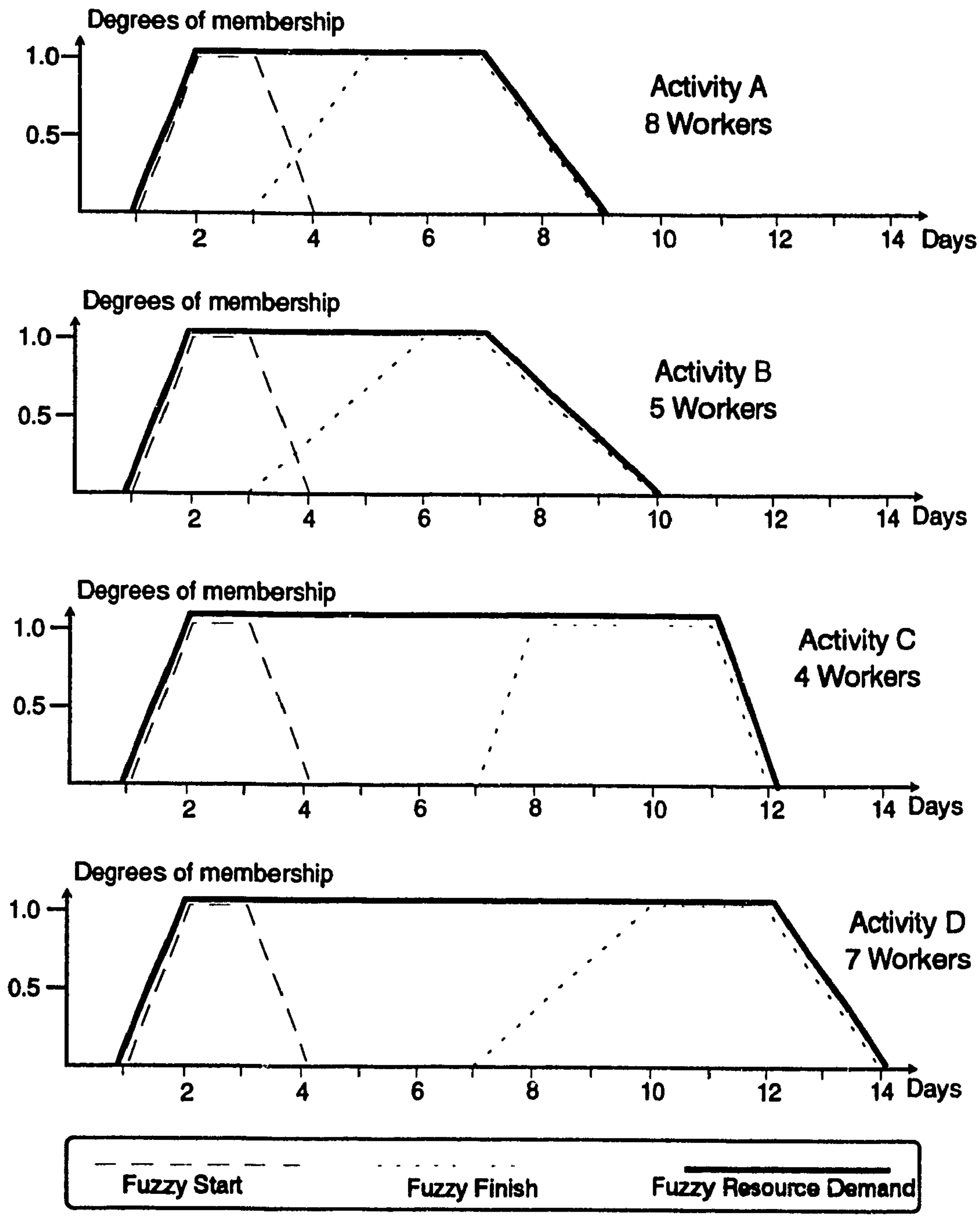


Fig. 4.10: Demonstration of Fuzzy Resource Demands.

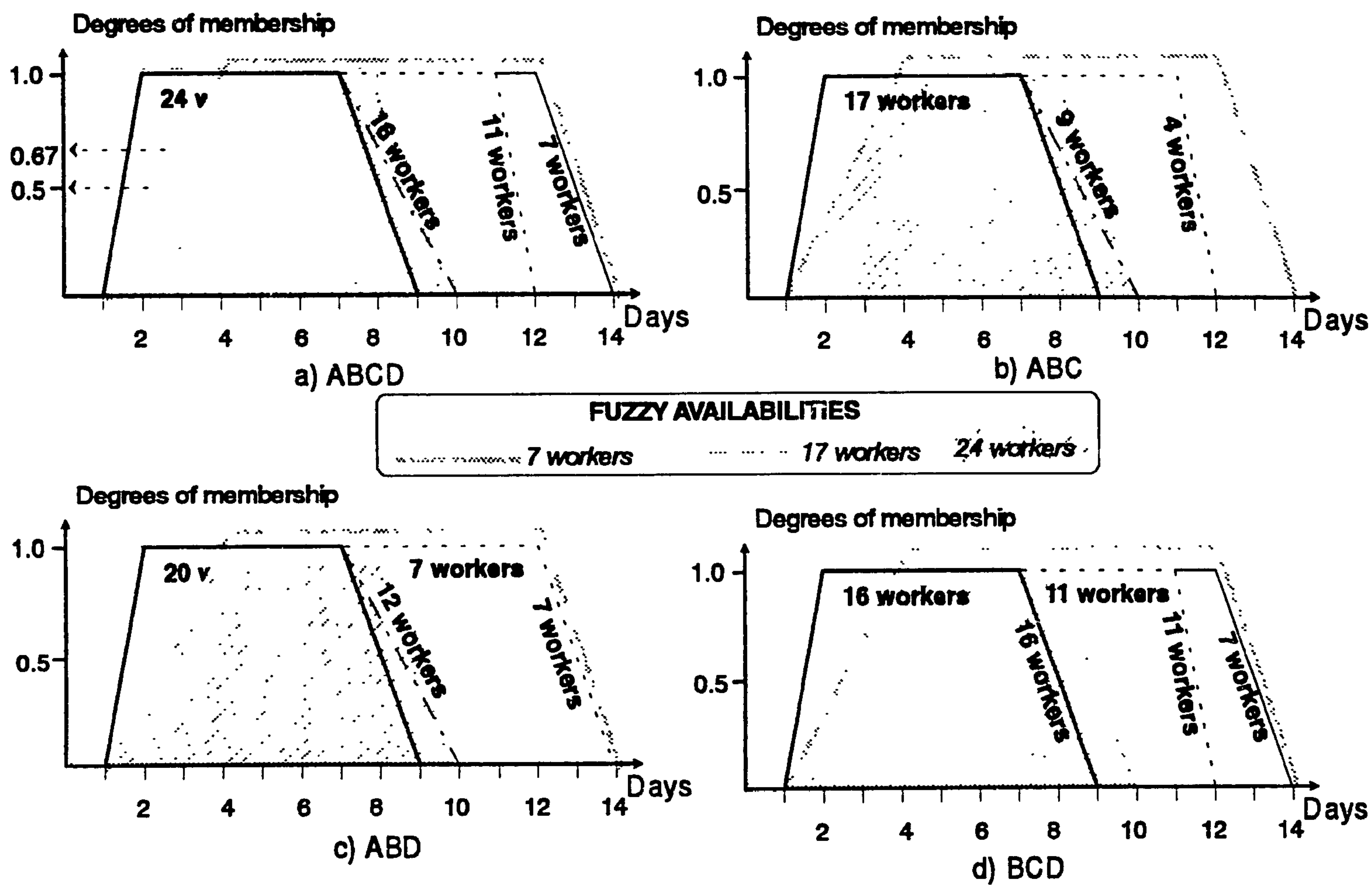


Fig. 4.11: Fuzzy Resource Demands of Activity Sets.

four possible different resource profiles yielding 24, 16, 11, and 7 workers respectively. For example, the FRQ for the 24 workers is calculated using Eq. 4.49 as:

$$R_{24\ workers} = [(8+5+4+7), \{(1,2,7,9)\cap(1,2,7,10)\cap(1,2,11,12)\cap(1,2,12,14)\}]$$
$$= [24,(1,2,7,9)]$$

The FRQs for other number of workers can be calculated in a similar manner. At a certain day, say day 8, the overall resource aggregation profile can be interpreted as: *the possibilities that the set consisting of activities A,B,C,D requires 24, 16, 11, and 7 workers can be calculated as 0.5, 0.67, 1.0, and 1.0 respectively,* (see Fig. 4.11a).

The feasibility of this set is then examined. The set is considered infeasible if at any time the degree of resource demand exceeds that of the degree of availability. Fig. 4.11a illustrates fuzzy demands and availabilities for the set A,B,C,D. Fuzzy availabilities consist of three FRQs: 7, 17, and 24 workers, illustrated using lighter color. From this figure it can be seen that at the times between days 1 and 4 the degrees of demand for 24 workers are higher than their respective availabilities. Also between days 8 and 12, this set requires 11 workers, but the profile of availability for 17 workers ends at day 10. Only 7 workers are available from the beginning to the end of day 14. As such it can be concluded that resources are not sufficient to schedule activities A,B,C, and D simultaneously. Therefore set-decomposition is required.

Following the activity set decomposition procedure described in Chapter 3, activities A,B,C, and D are used to generate three smaller subsets: ABC, ABD, and BCD. Fuzzy resource demands for these three subsets are depicted in Figs. 4.11 b, c, and d respectively. These three subsets are, once again, examined for their feasibility using the same procedure described earlier. From Figs. 4.11 b and d, it can be seen that only subsets ABC and BCD are feasible. With regard to subset ABD, it can be observed from Fig. 4.11c that during days 1 to 4 the degrees for the demand of 20 workers are higher than the degrees for the availability of 24 workers. As such subset ABD is not feasible, rendering the need for further decomposition. Assuming that no new feasible subset is found, as a result, two feasible subsets, ABC and BCD, are considered for further evaluations.

The next step is to calculate fuzzy next time frame (**FNT**) for all feasible subsets. The **FNT** coincides with the smallest **FF** of the activities included the subset being considered. The determination of the smallest **FF** is performed by comparing the magnitude of the geometric centroid calculated using Eq. 4.45. Fuzzy numbers with larger geometric centroid values are considered to be greater than those having smaller ones. If there is a tie, the upper bound value of the quadruple shall be used to break the tie. The calculated geometric centroid for the **FNT** of each subset are given in Table 4.3. The remaining total float (**RTF**) generated by each subset is taken as the difference between the geometric centroids of **FNT** and **FLS** of the postponed activity. If there are more than one activities to be postponed, the smallest **FLS** shall be used in calculating the **RTF**. The calculated **RTF** for each subset is given in Table 4.3. Resources are allocated to the subset which has the smallest **RTF** (i.e. subset ABC in this case). Resource availabilities in the pool are updated. The above cycle is repeated until all activities have been scheduled.

Table 4.3: Results of Fuzzy Resource Allocation.

Activity Subsets	FNT	C^{FNT}	FLS	C^{FLS}	$C^{FLS} - C^{FNT}$
ABC	(7,8,11,12)	9.5	(3,5,7,9)	6.0	-3.5
BCD	(7,10,12,14	10.7	(5,7,9,11)	8.0	-2.7

FNT = Fuzzy Next Time Frame; C^X = Centroid of Fuzzy number X;

FLS = Fuzzy Late Start

CHAPTER 5

COMPARISON BETWEEN FUZZY AND PROBABILISTIC SCHEDULING

5.1 Introduction

This chapter presents a comprehensive comparison between the fuzzy network scheduling method (FNET) described in Chapter 4 and the probabilistic network methods including PERT and Monte Carlo simulation through the use of numerical examples. A number of scenarios are generated from those examples to illustrate the attractive features of FNET over the highly regarded probabilistic methods. The comparison is conducted considering three aspects: 1) suitability of the theory, 2) method assumptions, and 3) scheduling information provided to the user, upon completion of the analysis. Each of these aspects are discussed in turn in the following sections.

5.2 Suitability of The Theory

Probability theory was originally developed in an effort to provide a basis for dealing with uncertainty due to randomness (Bellman and Zadeh 1970, Milton and Arnold 1986). The theory is conceptually based on experiments which are repeated, theoretically under identical circumstances, and without mutual dependence. To model the characteristic of a variable (i.e. activity durations in the case of scheduling) various steps would have to be carried out to compile historical data before they can be used on the calculations of new projects (AbouRizk and

Halpin 1992, and AbouRizk et al. 1994). These steps include: 1) collecting historical data, 2) constructing a frequency histogram of the gathered data, 3) selecting a probability density function to represent the data set, 4) estimating statistical parameters associated with the selected probability function, and 5) testing for goodness of fit. Theoretically, the above procedure is required for all activities in the network. If the simulation is to be used, correlation coefficients among project activities are additionally required. Apparently, this process requires considerable documented field observations.

In practice, however, most of the construction companies do not systematically record the durations of project activities for statistical analysis. Estimating activity durations are usually based on experience as well as gut-feeling of experts. The statements used by the experts to describe their estimations usually contain linguistic expressions such as "approximately," "more or less," "about," etc. These types of expressions exhibit some sort of imprecision that naturally leads to a range of possible values, rather than a single-valued variable. In addition, the relative degree of belief associated with each of these values could be different. Such statements as *"the duration for excavating this foundation could be anywhere between 5 to 12 days"* which indicates a range of possible durations, or more specifically, *"the duration for excavating this foundation is between 7 to 9 days but due to some factors, the progress may sometimes be as slow as 12 days or as fast as 5 days"* which signifies the most plausible range within the total range are frequently encountered in the statements used by experts. These aforementioned durations can practically be estimated as being within a certain range without even any knowledge of a probability distribution within that interval.

Uncertainties inherent in the duration of project activities can occasionally stem from vague definitions of the terms used by different parties involved in the construction. For example, the term "completion" could have different meanings to different parties. To a main contractor for instance, this term could mean completely demobilizing all of the resources used by subcontractors. To the subcontractor, on the other hand, "completion" could mean substantially completed to the extent that successive work can start. When asked to estimate the duration of the activity, the subcontractor could provide what might be the optimistic value to the main contractor. Such an undesirable situation, although can be improved through effective communications, they cannot be absolutely prevented from happening. Realistic schedules should be flexible enough to accommodate this type of uncertainty.

The types of uncertainties described earlier can be classified as nonstatistical. They are more of the vague-type (fuzzy) rather than the random one. Scheduling variables that exhibit this type of uncertainties can be represented in a more natural and simple way using the trapezoidal fuzzy number (as in the case of FNET) rather than the marginal probability distribution (as in the case of PERT and

Monte Carlo simulation). The use of trapezoidal fuzzy numbers is even more appropriate particularly for the project that employs new construction techniques and/or new materials where the relevant historical data is not available. Fuzzy set theory itself is developed as a tool to: 1) deal with uncertainty that is nonstatisitcal in nature, and 2) to model human perceptions about real-life problems (Zadeh 1965). In the scheduling context, if the input data is fuzzy, the output should also be fuzzy. Therefore the solution that is based on fuzzy set theory is intuitively more appealing than those based on probability theory.

5.3 Theoretical Assumptions

In general, PERT is based on three basic assumptions (Elmaghraby 1977, Hendrickson and Au 1989, and Badiru 1993): 1) activities are independent, 2) the critical path is substantially longer than other paths, and 3) the critical path contains a "sufficiently large" number of activities. The first assumption also applies to FNET, but it is not necessary for Monte Carlo Simulation only when the analysis accounts for the correlations among project activities. The data required for this type of analysis are, however, difficult to obtain and maintain, and is rarely available in practice.

The second assumption has consistently been criticized as a major drawback for PERT. In the situation where project networks contain several near-critical paths, project durations calculated using PERT are optimistic (Hendrickson and Au 1989, Badiru 1993). The optimistic results can also be observed when calculating the probability o, occurrence for an expected event. This is due to the fact that the expected project duration and its variance are computed from their respective values of the activities that lie exclusively on the critical path. The determination of the critical path is performed in the same manner as that in the traditional CPM calculations. Consequently, the project completion time is the sum of the durations of the critical activities identified. No consideration, however, is given to the fact that noncritical paths merging into the critical path could be delayed (i.e. merge bias) and consequently could extend the project completion time.

This drawback has been treated in the proposed FNET method. In the forward pass calculation of CPM and PERT, the expected early start time of a joint node is determined from the largest early finish of all the activities leading to that node. Unlike CPM and PERT, FNET employs the **Mãx** operation (see Eq.4.10) to determine the earliest possible start time of the joint activities. The **Mãx** operation performs paired-wise comparisons for each and every element in the quadruples involved (i.e. the **FEF** of the preceding activities), and accordingly selects the maximum value of each pair to represent their respective elements for the new quadruples (i.e. the **FES** of the joint activity). This procedure has alleviated such a drawback associated with PERT. In current practice, this limitation can be overcome using Monte Carlo simulation technique. This technique, however, requires considerably more computation effort.

The third assumption enables the use of the central limit theorem. The central limit theorem imposes a specific characteristic on the calculated project duration. It suggests that the project duration resulting from the use of this theorem is normally distributed, disregarding the distribution assumed for activity durations (Badiru 1993). The proposed FNET method, on the other hand, produces an output in the form of simple linear approximations. The shape of the distribution for fuzzy project duration produced depends on those of the input data. For instance, if all inputs of activity durations are triangular, the resulting project duration will be the triangular. This form of output is believed to be easier for the project team to adopt for practical purposes. The main reason is that it is more direct and more natural, for example, to represent linguistic interpretations such as "about", "between", etc. in the form of the trapezoidal or triangular fuzzy numbers rather than in the form of a probabilistic density function.

5.4 Scheduling Outputs

The main purposes of this section are two fold: 1) to demonstrate the calculations performed using FNET and to interpret the results generated, 2) to compare the results of FNET to those using probabilistic methods including PERT and Monte Carlo simulation. The subsequent subsection describes, first, the characteristics of the example networks used for the comparison. Next, detailed outputs generated by FNET and the step-by-step demonstration of the calculations performed using a selected example are presented. Then, the comparison of the results obtained using the three methods is discussed. Finally, the summary of this chapter is described.

5.4.1 Example Networks

Two example networks are worked out to demonstrate the capabilities and the performance of the proposed FNET method. The first example (i.e. Example Network No. 3) is a PDM-type network consisting of 17 activities. The network, including logical relationships as well as the durations of each activity, is shown in Fig. 5.1. Two sets of activity durations are assumed for this network. The first set (project A) is designed such that the network exhibits multiple critical paths. The second set (project B), on the other hand, renders the network to have a dominant longest path. In regard to the flexibility in modeling uncertainties associated with activity durations, FNET is capable of handling the coexistence of various degrees of fuzziness (i.e. crisp, interval, triangular, and trapezoidal) that could occur in most construction projects. To enable the comparison with the probabilistic methods, however, all activity durations are assumed to have a triangular distribution. The durations of activities in both projects, expressed in the form of quadruples, probabilistic means and variances are listed in Table 5.1.

The second example (i.e. Example Network No. 4) is adopted from the literature (Hendrickson and Au 1989). The network and its detailed description are shown in Fig. 5.2. The estimated optimistic, most likely and pessimistic durations for the

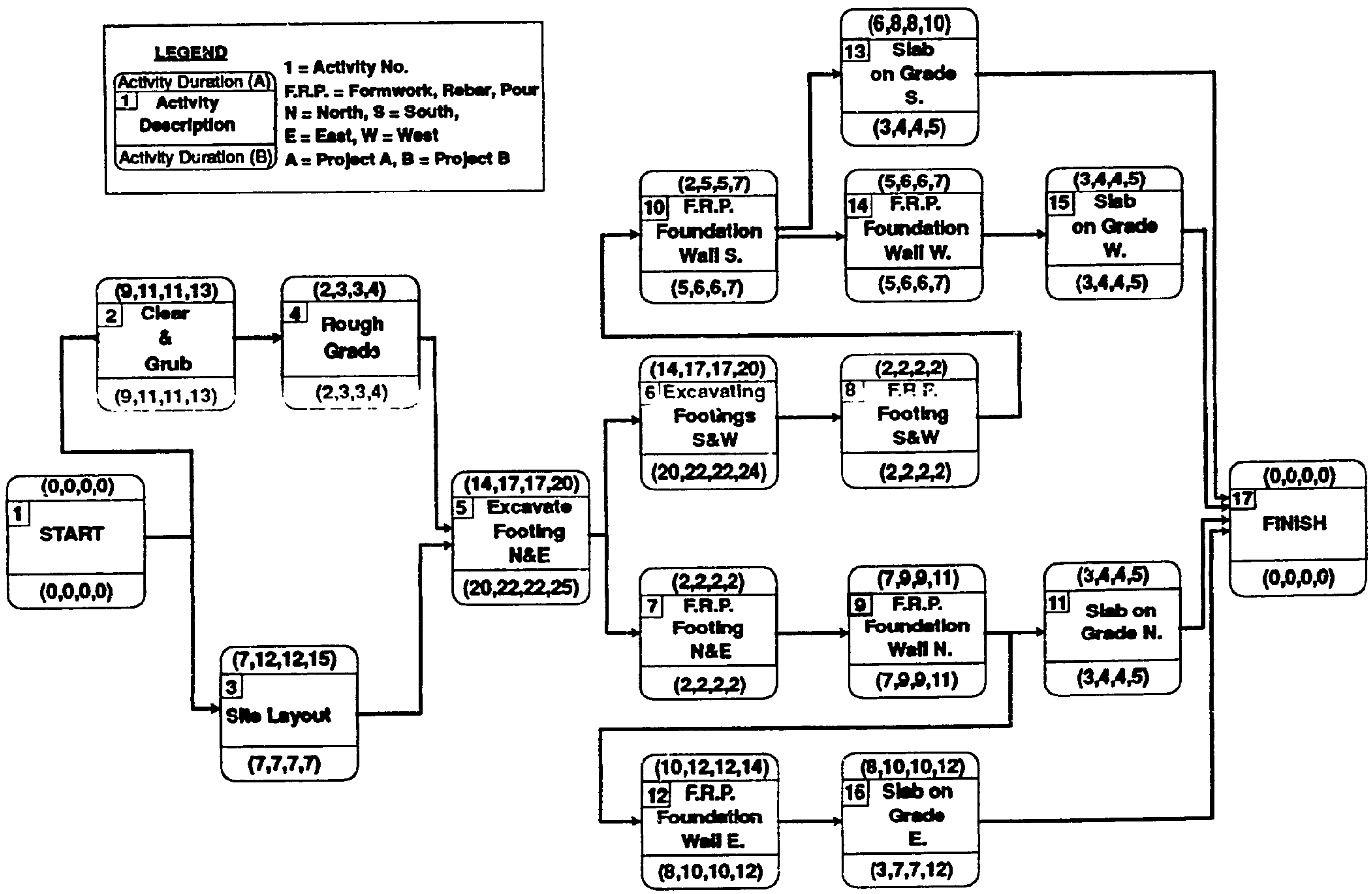


Fig. 5.1: Example Network No. 3.

Table 5.1: Input data for Example Network No. 3.

Activity	Project A			Project B		
	TFN	Mean	Var.	TFN	Mean	Var.
1	(0,0,0,0)	0.0	0.00	(0,0,0,0)	0.0	0.00
2	(9,11,11,13)	11.0	0.67	(9,11,11,13)	11.0	0.67
3	(7,12,12,15)	11.3	2.72	(7,7,7,7)	7.0	0.00
4	(2,3,3,4)	3.0	0.17	(2,3,3,4)	3.0	0.17
5	(14,17,17,20)	17.0	1.50	(20,22,22,25)	22.3	1.06
6	(14,17,17,20)	17.0	1.50	(20,22,22,24)	22.0	0.67
7	(2,2,2,2)	2.0	0.00	(2,2,2,2)	2.0	0.00
8	(2,2,2,2)	2.0	0.00	(2,2,2,2)	2.0	0.00
9	(7,9,9,11)	9.0	0.67	(7,9,9,11)	9.0	0.67
10	(3,5,5,6)	4.7	0.39	(5,6,6,7)	6.0	0.17
11	(3,4,4,5)	4.0	0.17	(3,4,4,5)	4.0	0.17
12	(9,12,12,16)	12.3	2.05	(8,10,10,12)	10.0	0.67
13	(6,8,8,10)	8.0	0.67	(3,4,4,5)	4.0	0.17
14	(5,6,6,7)	6.0	0.17	(5,6,6,7)	6.0	0.17
15	(3,4,4,5)	4.0	0.17	(3,4,4,5)	4.0	0.17
16	(8,10,10,12)	10.0	0.67	(3,7,7,12)	7.3	3.39
17	(0,0,0,0)	0.0	0.00	(0,0,0,0)	0.00	0.00

TFN = Triangular Fuzzy Numbers; Var. = Variance

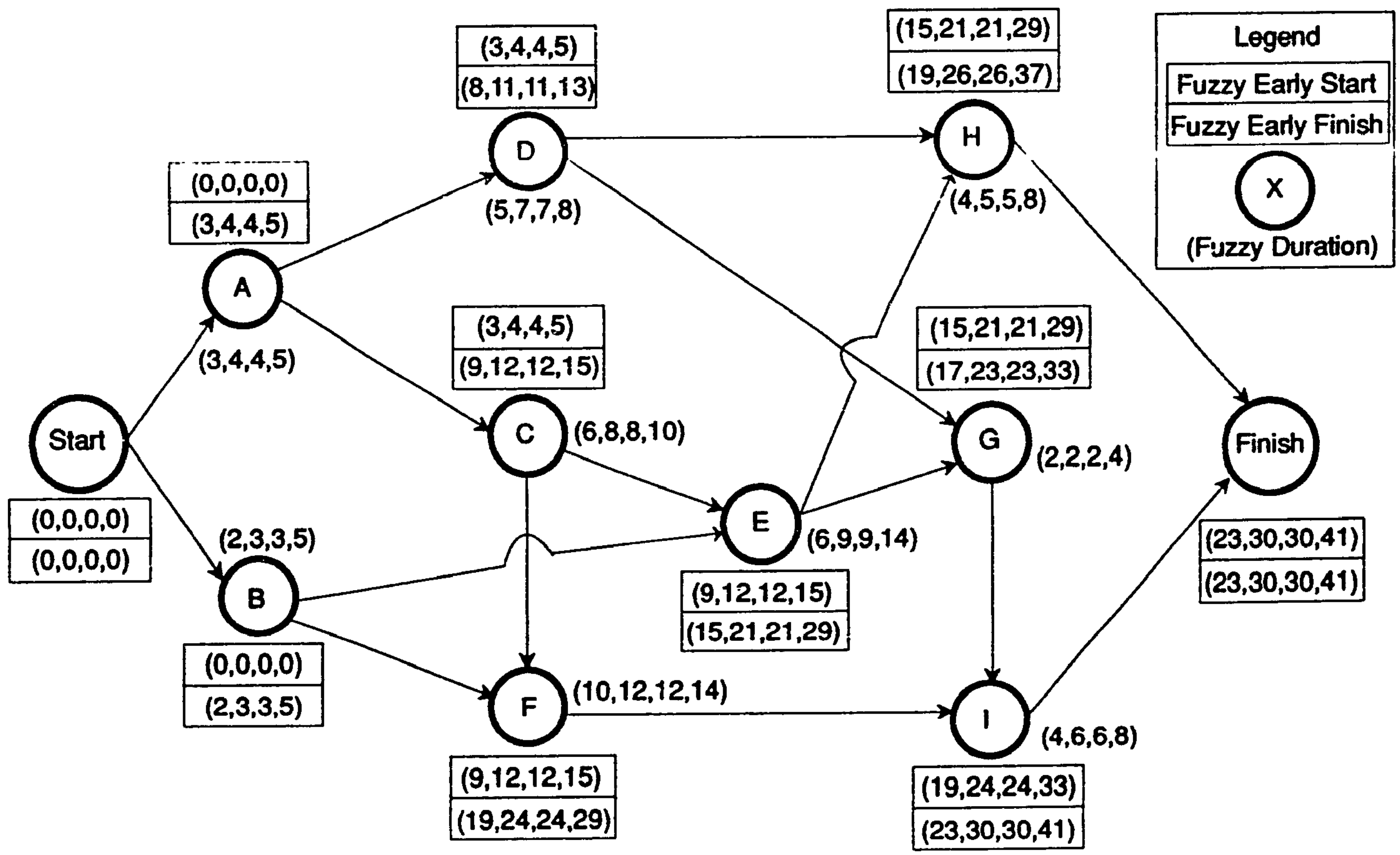


Fig. 5.2: Example Network No. 4. (adopted from Hendrickson and Au 1989).

nine activities are shown in Table 5.2. Two different probability distributions: the triangular and beta are assumed. In the case of the beta distribution, two different limits (Hendrickson and Au 1989): 1) the ninety-fifth percentile values, and 2) the absolute for the optimistic and pessimistic activity durations are examined. The means and variances of activity durations are also listed in Table 5.2. To apply FNET, a triangular membership function is assumed in consistency with the data used in PERT and Monte Carlo simulation. Activity durations are represented by fuzzy numbers expressed in the quadruple form (see Table 5.2).

5.4.2 Demonstration of FNET Calculations

The main objective of this subsection is to discuss, in detail, the step-by-step calculations performed using FNET, as well as to demonstrate how fuzzy results are interpreted. The results of the Example Network No. 3 are used for illustration purposes. Table 5.3 gives the detailed outputs for all activities of project **A**. In order to illustrate how these results are obtained, the calculations performed on a selected activity (i.e. activity 11, "Slab on Grade N.") are shown. In the forward pass calculations, using (Eq. 4.23), $\mathbf{FES}_{11}$ takes the value from the **FEF** of its predecessor (activity 9 in this case) to be (34,42,42,50). The $\mathbf{FEF}_{11}$ can then be calculated by adding its duration (3,4,4,5) to (34,42,42,50) using (Eq. 4.24) to become (37,46,46,55) days. The fuzzy project duration resulting from the forward pass calculation is taken as the **FEF** of activity 17 to be (52,66,66,78) days (Eq. 4.25). This duration can be interpreted as: *approximately 66 days, with a definite*

Table 5.2: Input data for Example Network No. 4.

Activity	Activity Durations			Beta Distribution			Triangular		Triangular Fuzzy Number
	Opt. (a)	Most. (m)	Pess. (b)	Mean	Var.[1]	Var.[2]	Mean[3]	Var.[4]	
A	3	4	5	4.0	0.40	0.11	4.0	0.17	(3,4,4,5)
B	2	3	5	3.2	0.90	0.25	3.3	0.39	(2,3,3,5)
C	6	8	10	8.0	1.60	0.44	8.0	0.67	(6,8,8,10)
D	5	7	8	6.8	0.90	0.25	6.7	0.39	(5,7,7,9)
E	6	9	14	9.3	6.40	1.78	9.7	2.72	(6,9,9,14)
F	10	12	14	12.0	1.60	0.44	12.0	0.67	(10,12,12,14)
G	2	2	8	2.3	0.40	1.00	2.7	0.22	(2,2,2,4)
H	4	5	8	5.3	1.60	0.44	5.7	0.72	(4,5,5,8)
I	4	6	8	6.0	1.60	0.44	6.0	0.67	(4,6,6,8)

[1]Variance $= \frac{1}{10}(b-a)^2$ [2]Variance $= \frac{1}{36}(b-a)^2$

[3]Mean $= \frac{1}{3}(a+b+m)$ [4]Variance $= \frac{1}{18}(a^2+b^2+m^2-ab-am-mb)$

Table 5.3: FNET Results of Example Network No. 3 (Project A).

Activity No.	Fuzzy Duration	Fuzzy Early Start	Fuzzy Early Finish	Fuzzy Late finish[u]	Fuzzy Late Start	Fuzzy Late Finish
1	(0,0,0,0)	(0,0,0,0)	(0,0,0,0)	$(-\infty,-\infty,0,0)$	(0,0,0,0)	(0,0,0,0)
2	(9,11,11,13)	(0,0,0,0)	(9,11,11,13)	$(-\infty,-\infty,11,13)$	(0,0,0,0)	(9,11,11,13)
3	(7,12,12,15)	(0,0,0,0)	(7,12,12,15)	$(-\infty,-\infty,14,17)$	(2,2,2,2)	(9,14,14,17)
4	(2,3,3,4)	(9,11,11,13)	(11,14,14,17)	$(-\infty,-\infty,14,17)$	(9,11,11,13)	(11,14,14,17)
5	(14,17,17,20)	(11,14,14,17)	(25,31,31,37)	$(-\infty,-\infty,31,37)$	(11,14,14,17)	(25,31,31,37)
6	(14,17,17,20)	(25,31,31,37)	(39,48,48,57)	$(-\infty,-\infty,48,57)$	(25,31,31,37)	(39,48,48,57)
7	(2,2,2,2)	(25,31,31,37)	(27,33,33,39)	$(-\infty,-\infty,33,41)$	(25,31,31,39)	(27,33,33,41)
8	(2,2,2,2)	(39,48,48,57)	(41,50,50,59)	$(-\infty,-\infty,50,59)$	(39,48,48,57)	(41,50,50,59)
9	(7,9,9,11)	(27,33,33,39)	(34,42,42,50)	$(-\infty,-\infty,42,52)$	(27,33,33,41)	(34,42,42,52)
10	(2,5,5,7)	(41,50,50,59)	(43,55,55,66)	$(-\infty,-\infty,55,66)$	(41,50,50,59)	(43,55,55,66)
11	(3,4,4,5)	(34,42,42,50)	(37,46,46,55)	$(-\infty,-\infty,66,78)$	(54,62,62,73)	(57,66,66,78)
12.	(10,12,12,14)	(34,42,42,50)	(44,56,56,64)	$(-\infty,-\infty,56,64)$	(34,42,42,52)	(44,56,56,66)
13	(6,8,8,10)	(43,55,55,66)	(49,63,63,76)	$(-\infty,-\infty,66,78)$	(45,57,57,68)	(51,65,65,78)
14	(5,6,6,7)	(43,55,55,66)	(48,61,61,73)	$(-\infty,-\infty,61,73)$	(43,55,55,66)	(48,61,61,73)
15	(3,4,4,5)	(48,61,61,73)	(51,65,65,78)	$(-\infty,-\infty,66,78)$	(48,61,61,73)	(51,65,65,78)
16	(8,10,10,12)	(44,56,56,64)	(52,66,66,76)	$(-\infty,-\infty,66,78)$	(44,56,56,66)	(52,66,66,78)
17	(0,0,0,0)	(52,66,66,78)	(52,66,66,78)	$(-\infty,-\infty,66,78)$	(52,66,66,78)	(52,66,66,78)

[u]Fuzzy Upper bound

minimum time of 52 days and a definite maximum completion time of 78 days. Other scheduling results can be interpreted in a similar manner.

Upon completion of the forward pass, backward pass calculations can start. The computations for the late times of each activity begins with the determination of its preliminary late finish (**PLF**) using (Eq. 4.28). As for the case of activity 11, $\mathbf{PLF}_{11}$ takes the smallest **FLS** of its successor(s) (i.e activity 17), which is (52,66,66,78). $\mathbf{FLF}_{11}{}^{u}$ can then be computed using $\mathbf{PLF}_{11}$ and Eq. 4.25 to be (-∞,-∞,66,78) days. The 4-step procedure described in Chapter 4 (see section 4.4.2) is employed to calculate the $\mathbf{FLF}_{11}$. The first step involves the determination of a more uncertain event between the $\mathbf{FLF}^{U}{}_{11}$ and $\mathbf{FEF}_{11}$. It can be seen that $\mathbf{FLF}^{U}{}_{11}$ is more uncertain than $\mathbf{FEF}_{11}$ [i.e. (78- 66) > (55 - 46). Following step 2, the fuzzy number **Y** can be calculated using (Eq. 4.31) to be (20,20,20,23). Steps 3 and 4 are performed to calculate the $\mathbf{FLF}_{11}$ and $\mathbf{FLS}_{11}$ using Eqs. 4.32 and 4.33 to be (57,66,66,78), and (54,62,62,73) days respectively.

Table 5.4 lists the calculated fuzzy start and finish times of all activities. The fuzzy start (**FS**) and fuzzy finish (**FF**) times are particularly useful for the calculations of fuzzy time intervals. These intervals are used in the resource allocation process. The **FS** of an activity is the intersection of its $\mathbf{FES}^{L}$ and $\mathbf{FLS}^{U}$. Similarly, the **FF** is the intersection of $\mathbf{FEF}^{L}$ and $\mathbf{FLF}^{U}$. Fuzzy start and finish times of activity 11 are calculated using Eqs. 4.34 and 4.35 to be (34,42,62,73) and (37,46,66,78) days

Table 5.4: Fuzzy Start and Finish Times of Activities in Example Network No. 3 (Project A).

Activity No.	Fuzzy Early Start[L]	Fuzzy Late Start[U]	Fuzzy Start	Fuzzy Early Finish[L]	Fuzzy Late Finish[U]	Fuzzy Finish
1	(0,0,∞,∞)	(-∞,-∞,0,0)	(0,0,0,0)	(0,0,∞,∞)	(-∞,-∞,0,0)	(0,0,0,0)
2	(0,0,∞,∞)	(-∞,-∞,0,0)	(0,0,0,0)	(9,11,∞,∞)	(-∞,-∞,11,13)	(9,11,11,13)
3	(0,0,∞,∞)	(-∞,-∞,2,2)	(0,0,2,2)	(7,12,∞,∞)	(-∞,-∞,14,17)	(7,12,14,17)
4	(9,11,∞,∞)	(-∞,-∞,11,13)	(9,11,11,13)	(11,14,∞,∞)	(-∞,-∞,14,17)	(11,14,14,17)
5	(11,14,∞,∞)	(-∞,-∞,14,17)	(11,14,14,17)	(25,31,∞,∞)	(-∞,-∞,31,37)	(25,31,31,37)
6	(25,31,∞,∞)	(-∞,-∞,31,37)	(25,31,31,37)	(39,48,∞,∞)	(-∞,-∞,48,57)	(39,48,48,57)
7	(25,31,∞,∞)	(-∞,-∞,31,39)	(25,31,31,39)	(27,33,∞,∞)	(-∞,-∞,33,41)	(27,33,33,41)
8	(39,48,∞,∞)	(-∞,-∞,48,57)	(39,48,48,57)	(41,50,∞,∞)	(-∞,-∞,50,59)	(41,50,50,59)
9	(27,33,∞,∞)	(-∞,-∞,33,41)	(27,33,33,41)	(34,42,∞,∞)	(-∞,-∞,42,52)	(34,42,42,52)
10	(41,50,∞,∞)	(-∞,-∞,50,59)	(41,50,50,59)	(43,55,∞,∞)	(-∞,-∞,55,66)	(43,55,55,66)
11	(34,42,∞,∞)	(-∞,-∞,62,73)	(34,42,62,73)	(37,46,∞,∞)	(-∞,-∞,66,78)	(37,46,66,78)
12.	(34,42,∞,∞)	(-∞,-∞,42,52)	(34,42,42,52)	(44,56,∞,∞)	(-∞,-∞,56,66)	(44,56,56,66)
13	(43,55,∞,∞)	(-∞,-∞,57,68)	(43,55,57,68)	(49,63,∞,∞)	(-∞,-∞,65,78)	(49,63,65,78)
14	(43,55,∞,∞)	(-∞,-∞,55,66)	(43,55,55,66)	(48,61,∞,∞)	(-∞,-∞,61,73)	(48,61,61,73)
15	(48,61,∞,∞)	(-∞,-∞,61,73)	(48,61,61,73)	(51,65,∞,∞)	(-∞,-∞,65,78)	(51,65,65,78)
16	(44,56,∞,∞)	(-∞,-∞,56,66)	(44,56,56,66)	(52,66,∞,∞)	(-∞,-∞,66,78)	(52,66,66,78)
17	(52,66,∞,∞)	(-∞,-∞,66,78)	(52,66,66,78)	(52,66,∞,∞)	(-∞,-∞,66,78)	(52,66,66,78)

[U] Fuzzy Upper Bound   [L] Fuzzy Lower Bound respectively. Accordingly the fuzzy interval of activity 11 is computed as $FS^{L}_{11} \cap FF^{U}_{11}$, which is (34,42,66,78). This fuzzy interval indicates a possible time window in which activity 11 can be executed.

Another useful information which can be obtained using FNET is shown in Table 5.5. This Table presents examples of the calculated fuzzy early start and finish times for all activities considered at different α-cut levels (see the definition of the α-cut in section 4.3.1 of Chapter 4). The α-cut levels could range anywhere between 0 and 1.0, in which the lower α-cut level includes a wider range of time values as compared to that of the higher level. in other words, the lower the α-cut level, the more pessimistic the schedule will be. One of the several advantages of using trapezoidal fuzzy numbers is that various α-cut levels of the calculated scheduling variables can be computed with relative ease.

Next, the criticality of project activities are calculated. Table 5.6 shows the fuzzy duration, the possibility measure, and the agreement index for each network paths in projects **A** and **B**. Consider for example the network paths of project **A**. It can be seen from Table 5.6 that none of them has both **PM** and **AI** values equal to 1.0. However, there are several paths that contain high **PM** and **AI** values (i.e. the value close to 1.0). This indicates that project **A** exhibits several nearly-critical paths. According to the calculated **PM** and **AI** values, path 1 is considered to be the most critical (i.e. **PM** = 0.96, **AI** = 0.93). Meanwhile, path 3 also has a very

Table 5.5: FNET Results of Example Network No. 3 (Project A) at Different α-cut Levels.

Activity No.	$\alpha = 0.0$		$\alpha = 0.5$		$\alpha = 0.7$		$\alpha = 1.0$	
	FES	FEF	FES	FEF	FES	FEF	FES	FEF
1	[0 ,0]	[0 ,0]	[0 ,0]	[0 ,0]	[0 ,0]	[0 ,0]	[0,0]	[0,0]
2	[0 ,0]	[9 ,13]	[0 ,0]	[10 ,12]	[0 ,0]	[10.4 ,12.4]	[0,0]	[11,11]
3	[0 ,0]	[7 ,15]	[0 ,0]	[9.5 ,13.5]	[0 ,0]	[10.5 ,14.1]	[0,0]	[12,12]
4	[9 ,13]	[11 ,17]	[10 ,12]	[12.5 ,15.5]	[10.4 ,12.4]	[13.1 ,16.1]	[11,11]	[14,14]
5	[11 ,17]	[25 ,37]	[12.5 ,15.5]	[28.0 ,34.0]	[13.1 ,16.1]	[29.2 ,35.2]	[14,14]	[31,31]
6	[25 ,37]	[39 ,57]	[28.0 ,34.0]	[43.5 ,52.5]	[29.5 ,35.2]	[45.3 ,54.3]	[31,31]	[48,48]
7	[25 ,37]	[27 ,39]	[28.0 ,34.0]	[30.0 ,36.0]	[29.2 ,35.2]	[31.2 ,37.2]	[31,31]	[33,33]
8	[39 ,57]	[41 ,59]	[43.5 ,52.5]	[45.5 ,54.5]	[45.3 ,54.3]	[47.3 ,56.3]	[48,48]	[50,50]
9	[27 ,39]	[34 ,50]	[30.0 ,36.0]	[38.0 ,46.0]	[31.2 ,37.2]	[39.6 ,47.6]	[33,33]	[42,42]
10	[41 ,59]	[43 ,66]	[45.5 ,54.5]	[49.0 ,60.5]	[47.3 ,56.3]	[51.4 ,62.7]	[50,50]	[55,55]
11	[34 ,50]	[37 ,55]	[38.0 ,46.0]	[41.5 ,50.5]	[39.6 ,47.6]	[43.3 ,52.3]	[42,42]	[46,46]
12.	[34 ,50]	[44 ,64]	[38.0 ,46.0]	[50.0 ,60.0]	[39.6 ,47.6]	[52.4 ,61.6]	[42,42]	[56,56]
13	[43 ,66]	[49 ,76]	[49.0 ,60.5]	[56.0 ,69.5]	[51.4 ,62.7]	[58.8 ,72.1]	[55,55]	[63,63]
14	[43 ,66]	[48 ,73]	[49.0 ,60.5]	[54.5 ,67.0]	[51.4 ,62.7]	[57.1 ,69.4]	[55,55]	[61,61]
15	[48 ,73]	[51 ,78]	[54.5 ,67.0]	[58.0 ,71.5]	[57.1 ,69.4]	[60.8 ,74.1]	[61,61]	[65,65]
16	[44 ,64]	[52 ,76]	[50.0 ,60.0]	[59.0 ,71.0]	[52.4 ,61.6]	[61.8 ,73.0]	[56,56]	[66,66]
17	[52 ,78]	[52 ,78]	[59.0 ,72.0]	[59.0 ,72.0]	[61.8 ,74.4]	[61.8 ,74.4]	[66,66]	[66,66]

Table 5.6: Criticality of Network Paths for Example Network No. 3.

Path No.	Sequence of Activities	Fuzzy durations	Possibility Measure	Agreement Index
1[a]	***1 - 2 - 4 - 5 - 6 - 8 - 10 - 14 - 15 - 17***	(51,65,65,78)	0.96	0.93
2[a]	1 - 2 - 4 - 5 - 7 - 9 - ***11*** - 17	(37,46,46,55)	0.14	0.02
3[a]	1 - 2 - 4 - 5 - ***7*** - ***9*** - ***12*** - ***16*** - 17	(52,64,64,76)	0.92	0.92
4[a]	1 - 2 - 4 - 5 - 6 - 8 - 10 - ***13*** - 17	(49,63,63,76)	0.89	0.79
5[a]	1 - 3 - 5 - 7 - 9 - 12 - 16 - 17	(48,62,62,74)	0.85	0.72
6[a]	1 - 3 - 5 - 7 - 9 - 11 - 17	(33,44,44,53)	0.12	0.01
7[a]	1 - ***3*** - 5 - 6 - 8 - 10 - 14 - 15 - 17	(47,63,63,76)	0.89	0.74
8[a]	1 - 3 - 5 - 6 - 8 - 10 - 13 - 17	(45,61,61,74)	0.82	0.62
1[b]	***1 - 2 - 4 - 5 - 6 - 8 - 10 - 14 - 15 - 17***	(66,76,76,87)	1.00	1.00
2[b]	1 - 2 - 4 - 5 - 7 - 9 - ***11*** - 17	(43,51,51,60)	0.00	0.00
3[b]	1 - 2 - 4 - 5 - ***7*** - ***9*** - ***12*** - ***16*** - 17	(52,64,64,76)	0.45	0.19
4[b]	1 - 2 - 4 - 5 - 6 - 8 - 10 - ***13*** - 17	(61,70,70,80)	0.70	0.52
5[b]	1 - 3 - 5 - 7 - 9 - 12 - 16 - 17	(48,57,57,66)	0.00	0.00
6[b]	1 - 3 - 5 - 7 - 9 - 11 - 17	(39,44,44,50)	0.00	0.00
7[b]	1 - ***3*** - 5 - 6 - 8 - 10 - 14 - 15 - 17	(62,69,69,77)	0.61	0.45
8[b]	1 - 3 - 5 - 6 - 8 - 10 - 13 - 17	(57,63,63,70)	0.23	0.07

Note: The activity shown in a ***bold, italic*** character assumes its criticality from the path in that row.

[a] and [b] indicate the project.

high degree of criticality (i.e. **PM** = 0.92, **AI** = 0.92). Paths 4,5,7, and 8 although exhibit less criticality than the former two paths, they can also become critical. Project teams should therefore pay close attentions to the activities on paths 1 and 3, and to a lesser degree to those on paths 4,5,7, and 8. In the case of project **B**, it can be seen from Table 5.6 that path 1 is clearly the only critical path (i.e. PM and AI = 1.0). Paths 4 and 7, however, can be considered as moderately critical, since about 50% of their path durations agree with the calculated fuzzy project duration (i.e. the AI values are approximately 0.5).

For the criticality at the activity level, let us continue to use activity 11 of project **A** as an illustrative example. This activity is present on paths 2 and 6. Since path 2 is more critical than path 6, activity 11 takes its **PM** and **AI** from those of path 2 (i.e. **PM** = 0.14 and **AI** = 0.02, see Table 5.4). The approximate total float of this activity can be obtained by first calculating the geometric centroids of $\mathbf{FES}_{11}$, $\mathbf{FLS}_{11}$, $\mathbf{FEF}_{11}$, and $\mathbf{FLF}_{11}$ using Eq. 4.45. These calculated values are then substituted into Eqs. 4.46 or 4.47, yielding 42, 63, 46, and 67 days for $\mathbf{FES}_{11}$, $\mathbf{FLS}_{11}$, $\mathbf{FEF}_{11}$, and $\mathbf{FLF}_{11}$, and 21 days [i.e. (63-42) or (67-46)] for the total float.

Other useful information which can be derived from the **PM** and **AI** values of each activity is the cumulative degree of criticality shown in Table 5.7. The cumulative degree of criticality is a composite indicator that combines two characteristics: the criticality and the degree for being a bottle-neck activity. Activities that belongs to

Table 5.7: Cumulative Criticality of Activities in Example Network No. 3 (Project A).

Activity	Project A				Project B			
	PM	ΣPM	AI	ΣAI	PM	ΣPM	AI	ΣAI
1	0.96	5.59	0.93	4.01	1.0	2.99	1.0	2.23
2	0.96	2.91	0.93	2.66	1.0	2.15	1.0	1.71
3	0.89	2.68	0.74	2.09	0.61	0.84	0.45	0.52
4	0.96	2.91	0.93	2.66	1.0	2.15	1.0	1.71
5	0.96	5.59	0.93	4.01	1.0	2.99	1.0	2.23
6	0.96	3.56	0.93	3.08	1.0	2.54	1.0	2.04
7	0.92	2.03	0.92	1.67	0.45	0.45	0.19	0.19
8	0.96	3.56	0.93	3.08	1.0	2.54	1.0	2.04
9	0.92	2.03	0.92	1.67	0.45	0.45	0.19	0.19
10	0.96	3.56	0.93	3.08	1.0	2.54	1.0	2.04
11	0.14	0.26	0.02	0.03	0.0	0.00	0.0	0.00
12	0.92	1.77	0.92	1.64	0.45	0.45	0.19	0.19
13	0.89	1.71	0.79	1.41	0.70	0.93	0.52	0.59
14	0.96	1.85	0.93	1.67	1.0	1.61	1.0	1.45
15	0.96	1.85	0.93	1.67	1.0	1.61	1.0	1.45
16	0.92	1.77	0.92	1.64	0.45	0.45	0.19	0.19
17	0.96	5.59	0.93	4.01	1.0	2.99	1.0	2.23

the same path (e.g. path 1 in this example) and have their respective **PM** and **AI** values identical, may have different cumulative **PM** and **AI**. For example, activities 4 and 5 lie in path 1, and have identical **PM** and **AI** values of 0.96 and 0.93 respectively (see Table 5.7). The respective cumulative **PM** and **AI** of the two activities are, however, different. The cumulative **PM** and **AI** for activity 4 are 2.91 and 2.66, while those of activity 5 are 5.59 and 4.01 respectively.

5.4.3 Comparison between FNET and Probabilistic Results

Considering first, Example Network no. 3, Monte Carlo simulation is performed on the two projects (i.e. **A** and **B**). The results on 5000 runs expressed in terms of mean and standard deviation for each activity are summarized in Tables 5.8 and 5.9. In the case of project **A**, the average project duration obtained with Monte Carlo simulation and PERT are 65.28 days with a standard deviation of 1.98 days. The fuzzy project duration obtained using FNET is (52,66,66,78) days. The percent deviation of the fuzzy mode from probabilistic mean is +1.1%.

With respect to other scheduling variables, disregarding those having fuzzy modes equal to zero, the percent deviations of fuzzy modes from their respective means range from -30.0% to +7.3% with an average of -1.60% for the underestimated values (i.e. the fuzzy mode is smaller than the probabilistic mean), and +1.68% for the overestimated values. Such a high deviation of -30% occurs only to the late start time of activity 3 where its value is relatively small (i.e. 2 days). As a result,

Table 5.8: Monte Carlo Simulation vs FNET Results for Example Network No. 3 (Project A).

Activity	Early start		Early Finish		Late Start		Late Finish		Total Float			
	MC[1] Mean	FNET Mode	MC[1] Mean	FNET Mode	MC[1] Mean	FNET Modes	MC[1] Mean	FNET Modes	MC[1] Mean	FNET TF	FNET PM	FNET AI
1	0.00	0	0.00	0	0.04	0	0.04	0	0.04	0.00	0.96	0.93
2	0.00	0	11.03	11	0.04	0	11.07	11	0.04	0.00	0.96	0.93
3	0.00	0	11.18	12	2.86	2	14.04	14	2.86	2.00	0.89	0.74
4	11.03	11	14.00	14	11.07	11	14.04	14	0.04	0.00	0.96	0.93
5	14.04	14	31.08	31	14.04	14	31.08	31	0.00	0.00	0.96	0.93
6	31.08	31	48.05	48	31.64	31	48.63	48	0.57	0.00	0.96	0.93
7	31.08	31	33.07	33	32.33	31	34.33	33	1.26	0.67	0.92	0.92
8	48.05	48	50.05	50	48.63	48	50.63	50	0.57	0.00	0.96	0.93
9	33.08	33	42.09	42	34.33	33	43.34	42	1.26	0.67	0.92	0.92
10	50.05	50	54.68	55	50.62	50	55.25	55	0.57	0.00	0.96	0.93
11	42.09	42	46.06	46	61.30	62	65.28	66	19.21	21.00	0.14	0.02
12	42.09	42	54.04	56	43.34	42	55.29	56	1.26	0.67	0.92	0.92
13	54.68	55	62.67	63	57.28	57	65.28	65	2.60	2.00	0.89	0.79
14	54.68	55	60.70	61	55.26	55	61.28	61	0.58	0.00	0.96	0.93
15	60.70	61	64.70	65	61.28	61	65.28	65	0.58	0.00	0.96	0.93
16	54.04	56	64.02	66	55.29	56	65.28	66	1.26	0.67	0.92	0.92
17	64.70	66	64.70	66	65.28	66	65.28	66	0.58	0.00	0.96	0.93

[1] Monte Carlo Simulation

Project Duration obtained using Monte Carlo simulation:

Mean = 65.28 days; Standard deviation = 1.98

Project duration obtained using FNET = (52,66,66,78) days.

Table 5.9: Monte Carlo Simulation vs FNET Results for Example Network No. 3 (Project B).

Activity	Early start		Early Finish		Late Start		Late Finish		Total Float			
	MC[1] Mean	FNET	MC[1] Mean	FNET	MC[1] Mean	FNET	MC[1] Mean	FNET	MC[1] Mean	FNET TF	FNET PM	FNET AI
1	0.00	(0,0,0,0)	0.00	(0,0,0,0)	0.00	(0,0,0,0)	0.00	(0,0,0,0)	0.00	0.0	1.0	1.0
2	0.00	(0,0,0,0)	11.02	(9,11,11,13)	0.00	(0,0,0,0)	11.02	(9,11,11,13)	0.00	0.0	1.0	1.0
3	0.00	(0,0,0,0)	7.00	(7,7,7,7)	7.04	(7,7,7,7)	14.04	(14,14,14,17)	7.04	8.0	0.89	0.74
4	11.02	(9,11,11,13)	14.04	(11,14,14,17)	11.02	(9,11,11,13)	14.04	(11,14,14,17)	0.00	0.0	1.0	1.0
5	14.04	(11,14,14,17)	36.37	(31,36,36,42)	14.04	(11,14,14,17)	36.37	(31,36,36,42)	0.00	0.0	1.0	1.0
6	36.37	(31,36,36,42)	58.36	(51,58,58,66)	36.37	(31,36,36,42)	58.36	(51,58,58,66)	0.00	0.0	1.0	1.0
7	36.37	(31,36,36,42)	38.37	(33,38,38,44)	48.77	(42,47,47,53)	50.77	(44,49,49,55)	12.40	11.0	0.45	0.19
8	58.36	(51,58,58,66)	60.36	(53,60,60,68)	58.36	(51,58,58,66)	60.36	(53,60,60,68)	0.00	0.0	1.0	1.0
9	38.37	(33,38,38,44)	47.40	(40,47,47,55)	50.77	(44,49,49,55)	59.81	(51,58,58,66)	12.40	11.0	0.45	0.19
10	60.36	(53,60,60,68)	66.37	(58,66,66,75)	60.36	(53,60,60,68)	66.37	(58,66,66,75)	0.00	0.0	1.0	1.0
11	47.40	(40,47,47,55)	51.40	(43,51,51,60)	72.37	(65,72,72,82)	76.37	(68,76,76,87)	24.97	25.0	0.0	0.0
12	47.40	(40,47,47,55)	57.40	(48,57,57,67)	59.81	(51,58,58,66)	69.81	(59,68,68,78)	12.40	11.0	0.45	0.19
13	66.37	(58,66,66,75)	70.36	(61,70,70,80)	72.38	(64,72,72,82)	76.37	(67,76,76,87)	8.01	6.3	0.7	0.52
14	66.37	(58,66,66,75)	72.38	(63,72,72,82)	66.37	(58,66,66,75)	72.38	(63,72,72,82)	0.00	0.0	1.0	1.0
15	72.38	(63,72,72,82)	76.37	(66,76,76,87)	72.38	(63,72,72,82)	76.37	(66,76,76,87)	0.00	0.0	1.0	1.0
16	57.40	(48,57,57,67)	63.97	(52,64,64,76)	69.81	(59,68,68,78)	76.37	(63,75,75,87)	12.40	11.0	0.45	0.19
17	76.37	(66,76,76,87)	76.37	(66,76,76,87)	76.37	(66,76,76,87)	76.37	(66,76,76,87)	0.00	0.0	1.0	1.0

[1] Monte Carlo Simulation

Project Duration obtained using Monte Carlo simulation:
Mean = 76.37 days
Standard deviation = 1.76

Project duration obtained using FNET:
(66,76,76,87) days

a small deviation of 0.86 can cause a high percentage of deviation.

With regard to the criticality measurement, Monte Carlo simulation generally overestimates FNET. It overestimates the total floats in 15 out of 17 activities. Disregarding the zero total float values, the overestimation ranges from 23% to 47% with an average of 40%. Monte Carlo simulation underestimates the total float in 1 out of 17 activities with a percent underestimation calculated to be 19%. It may appear that the percent deviations associated with the total floats are higher than those encountered with other scheduling variables. This could be attributed to the relatively small magnitude of the total float itself.

As for the case of project **B** (see Table 5.9), the average project duration is calculated to be 76.37 days with a standard deviation of 1.76 days. The fuzzy project duration is (66,76,76,87) days. The percent deviations of the fuzzy mode from the mean duration is -0.48%. With regard to the differences between the means and fuzzy modes of all activities, the percent deviations of fuzzy modes from their respective means range from -3.6% to 0.05% with an average of -0.90% for the underestimation values and +0.05% for the overestimation. With respect to the criticality of project activities, the total floats calculated by the two methods are identical in 10 out of 17 activities. As for the remaining 7 activities, FNET underestimates Monte Carlo simulation in 4 activities with an average of -11.3%, and overestimates Monte Carlo simulation by an average of 6.0%. The results of the two projects generated by FNET and Monte Carlo simulation reveal reasonably close agreements. It should be noted however that all of the observations made earlier are applicable to the two examples being considered, they should not be viewed as a general trend.

In addition to the usual start and finish times, it is interesting to compare the probability to complete the project within a certain time. Such a probability can be calculated in Monte Carlo simulation using the cumulative probability (**CP**) and in FNET using the agreement index (**AI**) (Eq. 4.39). A number of expected project completion times for both projects are examined using the two methods. The results depicted in Fig. 5.3 indicate a wider range for the project duration associated with FNET and a smaller range associated with Monte Carlo simulation. This can be related to the manner in which the uncertainty inherent in the input data is modeled. In FNET, all of the three estimated durations (i.e. the optimistic, the pessimistic, and the most possible) of the critical activities are actually included in the spread of the calculated fuzzy project durations.

Theoretically, the standard deviation calculated for a duration is smaller than the difference between the optimistic and pessimistic values. As such, the spread of fuzzy project durations tend to be larger than the standard deviation obtained from PERT and Monte Carlo simulation. It should be noted however that although the **CP** and **AI** are comparable, they are two different scales used by two different

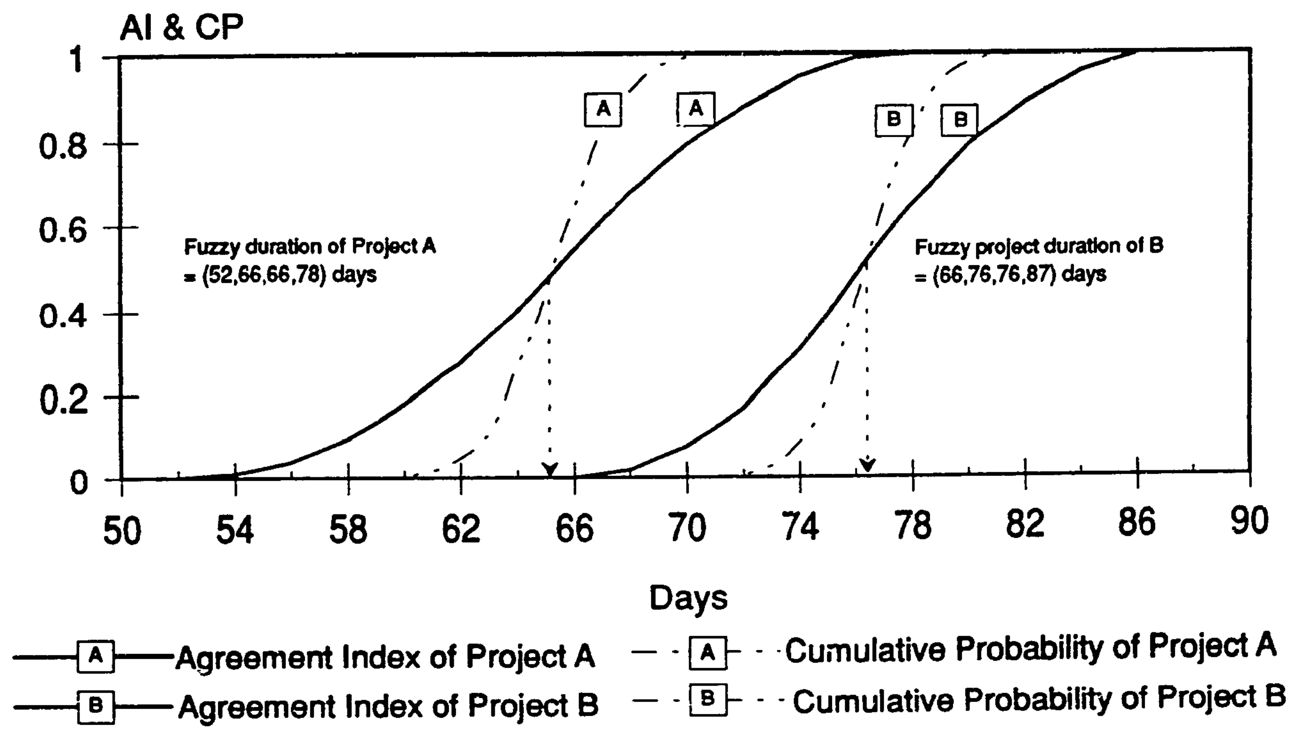


Fig. 5.3: Cumulative Probability vs Agreement Index for Example Network No. 3.

methods to describe the uncertainty associated with a given project completion time. The results of these two projects, however, indicate a similar pattern. That is the CP values tend to be smaller than the AI values when expected project durations are smaller than the calculated means and otherwise beyond the means.

The results of Example Network No. 4 generated by the three methods are summarized in Table 5.10. In general, probabilistic methods produce scheduling outputs in terms of means and variances of a probability density function. The distribution of project durations, following the central limit theorem, is generally assumed to be normal. The shape of fuzzy project durations calculated using FNET, on the contrary, is not predetermined. It depends on those of the input data. Table 5.10 also reveals the difference between the results obtained using PERT when altering the definitions of the optimistic and pessimistic values. For this example, the estimated durations for the critical activities are symmetrical. Therefore, the average project durations obtained with PERT are identical, disregarding the distributions assumed. This, however, is not true for the calculated standard deviations. Among the three PERT solutions, the standard deviation obtained using the 5 and 95 percentile estimates for the beta distribution is found to be the nearest to its respective value given by Monte Carlo simulation. This finding agrees with those stated in the literature (Moder and Rodgers 1968; Perry and Greig 1975). From this example, it can be seen that the shape of probability distributions as well as the definitions of three estimated activity durations can

Table 5.10: Comparison of Characteristics Among the Three Methods.

Methods	Input Distribution	Output Distribution	Project Duration (d)	Standard Deviation	P(d≤35)	P(d=35)	Is P(d=35) > P(d=28) ?	Is P(G = 27) > P(F = 27)?
PERT[a]	Beta	Normal	30	2.30	0.985	NA	NA	NA
PERT[b]	Beta	Normal	30	1.19	1.000	NA	NA	NA
PERT	Triangular	Normal	30	1.47	0.999	NA	NA	NA
MCS[c]	Beta	Normal	30.9	2.50	0.945	NA	NA	NA
MCS[d]	Beta	Normal	36.5	4.90	0.397	NA	NA	NA
FNET	Triangular	Triangular	(23,30,30,41)	NA	0.818	0.55	Yes[e]	Yes[f]

[a] The 95th percentile activity durations
[b] The absolute limit activity durations
[c] Monte Carlo simulation (Independent)
[d] Monte Carlo simulation (Correlation)
[e] Poss(d=35) = 0.55, Poss(d=28) = 0.71
[f] Poss(G=27) = 0.6, Poss(F=27) = 0.4

have a significant impact on the results calculated by PERT.

Fig. 5.4 depicts the cumulative probability values for different expected events calculated by the four probabilistic methods and the agreement indices computed by FNET. It can be seen that for this example, the results produced by FNET are reasonably close to those generated by Monte Carlo simulation. Similar to the pattern found in the previous example, the AI values calculated by FNET tend to overestimate their respective CP values of Monte Carlo simulation for the events that occur before the average project duration, underestimate otherwise. Most of CP values calculated using PERT, disregarding the type of the distribution and the definitions of the time estimates, generally overestimate their respective values obtained with Monte Carlo simulation.

In addition to the traditional scheduling information described earlier, the FNET method also provide other useful information which, by definition, can not be obtained from probabilistic methods. The possibility of such an event as the project duration exactly equal to 35 days can easily be calculated using the concept of possibility measure incorporated in FNET. The possibility of having the project duration exactly equal to 35 days was calculated using Eq. 4.37 to be 0.55 (see Table 5.10). This possibility, however, does not have a direct analogy in the PERT or Monte Carlo Simulation. This is due to the fact that the probability of a strict equality [i.e. $P(T = 35)$] is always zero when a continuous probability distribution

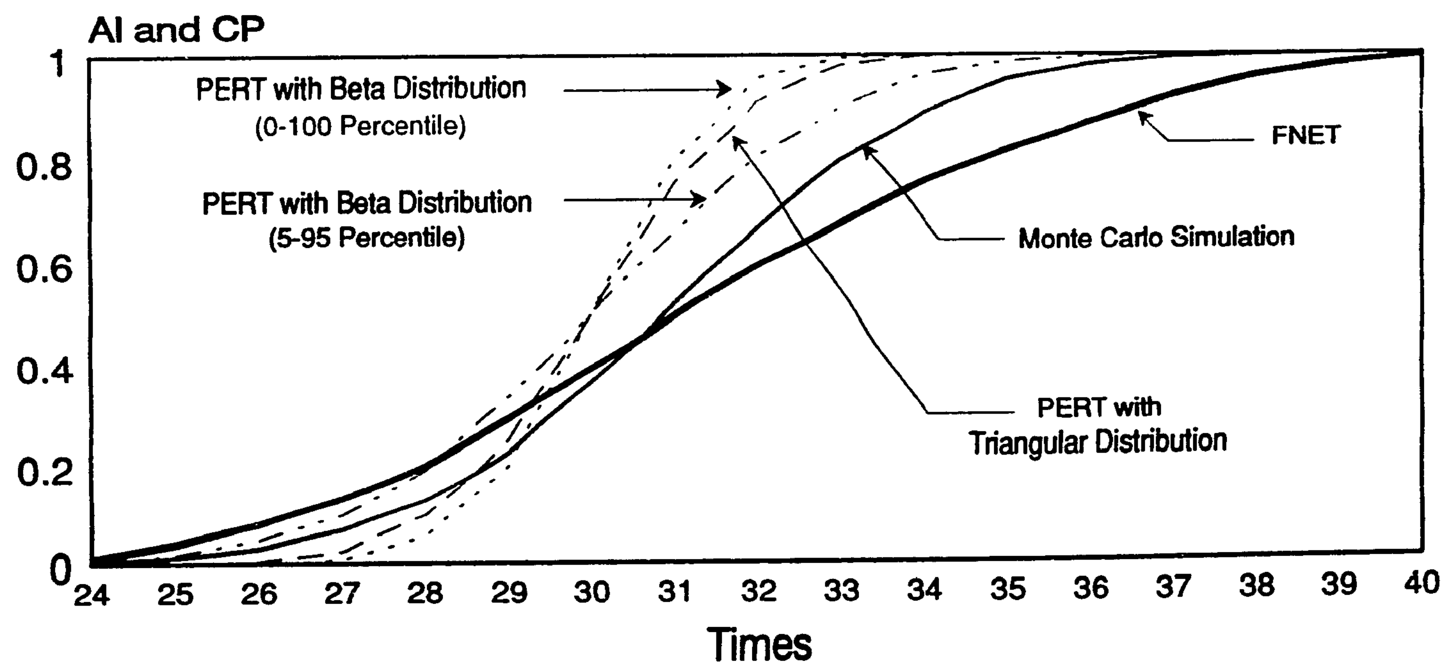


Fig. 5.4: Cumulative Probability vs Agreement Index for Example Network No. 4.

is used. Accordingly, probabilistic methods can not provide an answer to such a question as "*which of the following project durations: exactly 35 days or exactly 28 days is more likely to occur?*". The results obtained using the FNET method reveal that the project duration is more plausibly to be 28 days rather than 35 days.

Through the use of the possibility measure, it is also possible to determine which activity has more possibility to be completed on a certain date. For example, the project team might be interested in knowing which of the two activities (i.e. activities F and G for this network example) has a more possibility to be finished on the day 27. Based on the fuzzy early schedule illustrated in Fig. 5.2, the early finish of activities F and G are (19,24,24,29) and (17,23,23,33) days respectively. Using Eq. 4.37, the Poss(F=27) and Poss(G=27) are calculated to be 0.4, and 0.6 respectively. Accordingly, activity G has more chance to be completed on the day 27 than activity F.

5.5 Summary

For those situations where project teams cannot specify activity durations as exact or deterministic numbers, the durations can properly be represented as fuzzy numbers. If the input data are fuzzy then the scheduling outputs should also be fuzzy, rendering the solution that is based on fuzzy set theory suitable. This Chapter demonstrates the capability of the fuzzy network scheduling method FNET proposed in Chapter 4, and compares its performance to the highly regarded

PERT and Monte Carlo simulation methods. The comparison is conducted considering three aspects: 1) suitability of the theory, 2) method assumptions, and 3) scheduling information provided to the user, upon completion of the analysis.

From the theoretical point of view, fuzzy sets theory is suitable for modeling uncertainty associated with activity durations in construction network scheduling. In comparison with PERT and Monte Carlo simulation, the calculations performed using FNET are simple and more transparent, and the results are practical. The fact is that FNET provides a direct solution, and it requires less computational times than those of Monte Carlo simulation. As far as the comparison is concerned, the results obtained with FNET are reasonably close to those of Monte Carlo simulation. Some useful information which can be generated using FNET can not be obtained from the probabilistic methods.

FNET can alleviate a major shortcoming associated with PERT (i.e. focus on a single critical path) through the use of fuzzy maximum operation. As for the data acquisition FNET does not require historical data which, in practice, is difficult to compile and maintain. The linear approximation such as the trapezoidal fuzzy numbers expressed in the quadruples form used to represent activity durations are simple, effective, and practical.

CHAPTER 6

RESOURCE-CONSTRAINED SCHEDULING SYSTEM

6.1 Introduction

Currently, many computerized algorithmic scheduling systems capable of resource-allocation are commercially available (Johnson 1992). The majority of those systems perform resource allocation automatically, using a set of heuristic rules in conjunction with necessary assumptions to establish priorities and resolve resource conflicts. Those systems do not permit the user to engage in the decision making process. As a result, user's experience and project specific knowledge are not utilized in generating project schedules. The quality of these schedules therefore relies heavily on the performance of rigid heuristic models, and the scheduler's capability in defining input and interpreting output.

Without any access during the resource allocation process, the user would have to comprehend the required data and specify project constraints at the outset in order to produce a desirable schedule. In addition, the user will have to make sure that there is no conflict among the constraints being imposed. Otherwise, no valid schedule can be obtained. This requires huge efforts from the user. Of more importance, perhaps, is that the resulting schedules often bear little resemblance to the actual situation in view of the fact that project specific knowledge is not utilized.

This chapter presents a prototype decision support system developed to assist the scheduler in allocating resources to project activities that are executed in an uncertain environment. The following section outlines some of the desirable features for the system.

6.2 System Requirements

A desirable decision support scheduling system should possess three main features: 1) effective representations, 2) powerful reasoning strategies, and 3) user-friendly interface. With regard to the effectiveness of the system, issues related to numerical representations and temporal propagation have been of major concern in the system development. Object-oriented programming (OOP) has proved to exhibit a number of attractive features in modeling and representing engineering knowledge (De La Garza and Ibbs 1987, Logcher 1987, Froese and Obayashi 1990, Alagar and Periyasamy 1992, Hakim and Garrett 1992, Moselhi and Lorterapong 1993c). Therefore it has been selected as a tool in the development of the present system to represent domain objects such as activities, resources, and construction crews, etc.

In addition to the effective representation, a scheduling system should also have several inference strategies, This stems primarily from the characteristic of the scheduling which basically involves: 1) identifications of scheduling entities, 2) calculations of their attribute values, and 3) sending the messages from an entity to another. A programming environment which supports various inference strategies is therefore desirable for the development of the system. An effective scheduling system should also have user-friendly interfaces to support information exchanges between the user and its modules. The suitable programming environment should provide rich tools for developing effective means to enhance the communication between the user and the system.

6.3 System Outline

RESource-CONstrained Scheduler (RESCONS) is a prototype decision support system developed for resources allocation considering uncertainty. The main focus for the development of the proposed system is on the challenge of interchanging information between the user and the system to facilitate schedule preparations. Fig. 6.1 shows the architecture of the proposed system. It consists of five main modules (i.e. *Controller, User Interface, Scheduler, Resource Allocator*, and *Report Generator*). The system has access to two databases containing project specific and resources data respectively. The project specific database contains project activities and resource data items directly input by the user and calculated by the system. The resource database contains the information related to the availability of the resources in the organization. The user may be engaged in a dialog with the system, providing input data via the keyboard and mouse, specifying actions and choices from menus, and receiving output via a screen and hard copies.

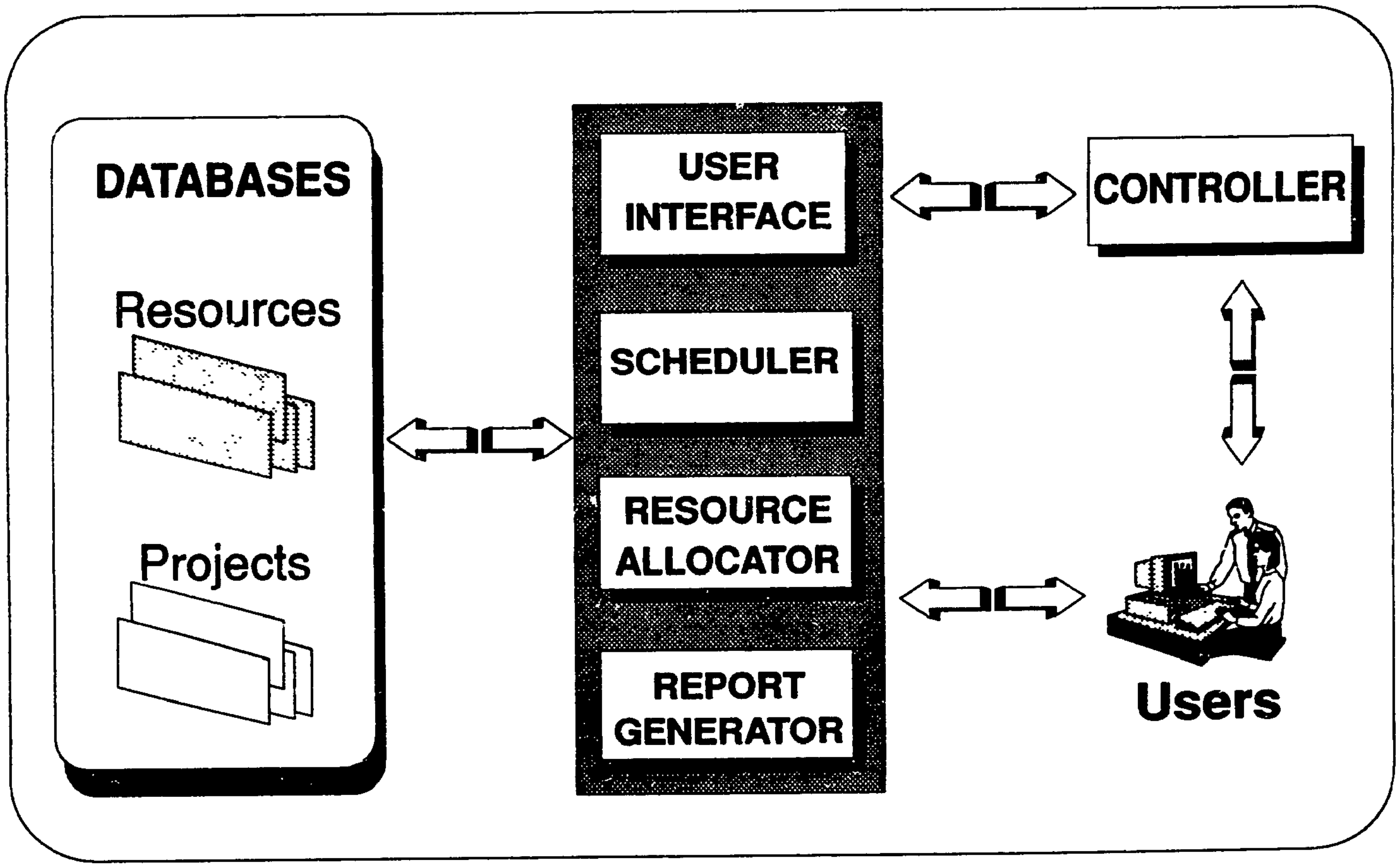


Fig. 6.1: RESCONS Architecture.

RESCONS has been implemented in an object-oriented programming environment called **LEVEL5 OBJECT™**, marketed by Information Builders, Inc. **LEVEL5 OBJECT** was chosen due its simplicity, facilitating rapid prototyping. The inference strategies of the prototype system include backward and forward chaining. In **RESCONS**, scheduling entities such as activities, resources, construction crews, and constraints are represented as objects. Each object contains attributes and a set of rules or methods to operate on its attributes. Objects communicate to each other via messages.

RESCONS adopted the fuzzy network scheduling method presented in Chapter 4. Temporal-related attributes such as activity durations, scheduling variables (e.g. start and finish times), resource availabilities and demands, are represented using fuzzy numbers. A class called fuzzy number is developed as a generic class for activity and resource classes. The fuzzy number class consists of a number of numerical attributes including lower and upper bounds, and lower and upper modal values together with a set of methods (e.g. fuzzy addition, fuzzy maximum, etc.) that are used to operate on these attributes. The activity and resource classes inherit all the features of the fuzzy number class. Inheritance is one of the many powerful features offered by OOP that helps the programmer organize a solution to make it easy to maintain and extend (Eckel 1989).

6.4 RESCONS Prototype

RESCONS generates project schedules through the interactive communications between the users and the system modules. The prototype supports both deterministic and nondeterministic scheduling. Fig. 6.2 shows a flow diagram of **RESCONS**. The system has four main functions: data preparations, unconstrained scheduling, resource-constrained scheduling, and report generations. These functions are respectively performed by four main modules: *User Interface, Scheduler, Resource Allocator,* and *Report Generator.* These four modules are supported by *Controller. Controller* first provides the user with 5 options: 1) create a new project, 2) edit existing projects, 3) perform scheduling, 4) allocate resources, and 5) produce reports. According to the selection made by the user, *Controller* activates the appropriate module for further actions. Detailed description including the functions of each four modules are presented in the following subsections.

6.4.1 User Interface Module

User Interface is activated by *Controller* when the user wants to create a new project or edit the data of the existing one. The main function of this module is to prepare all the data required by **RESCONS** for scheduling and resource allocation. This data is grouped into five categories: project data, activity data, resource availability data, and crew data. If the user chooses to edit an existing project, the title of that project is requested. The module then retrieves all the data from the

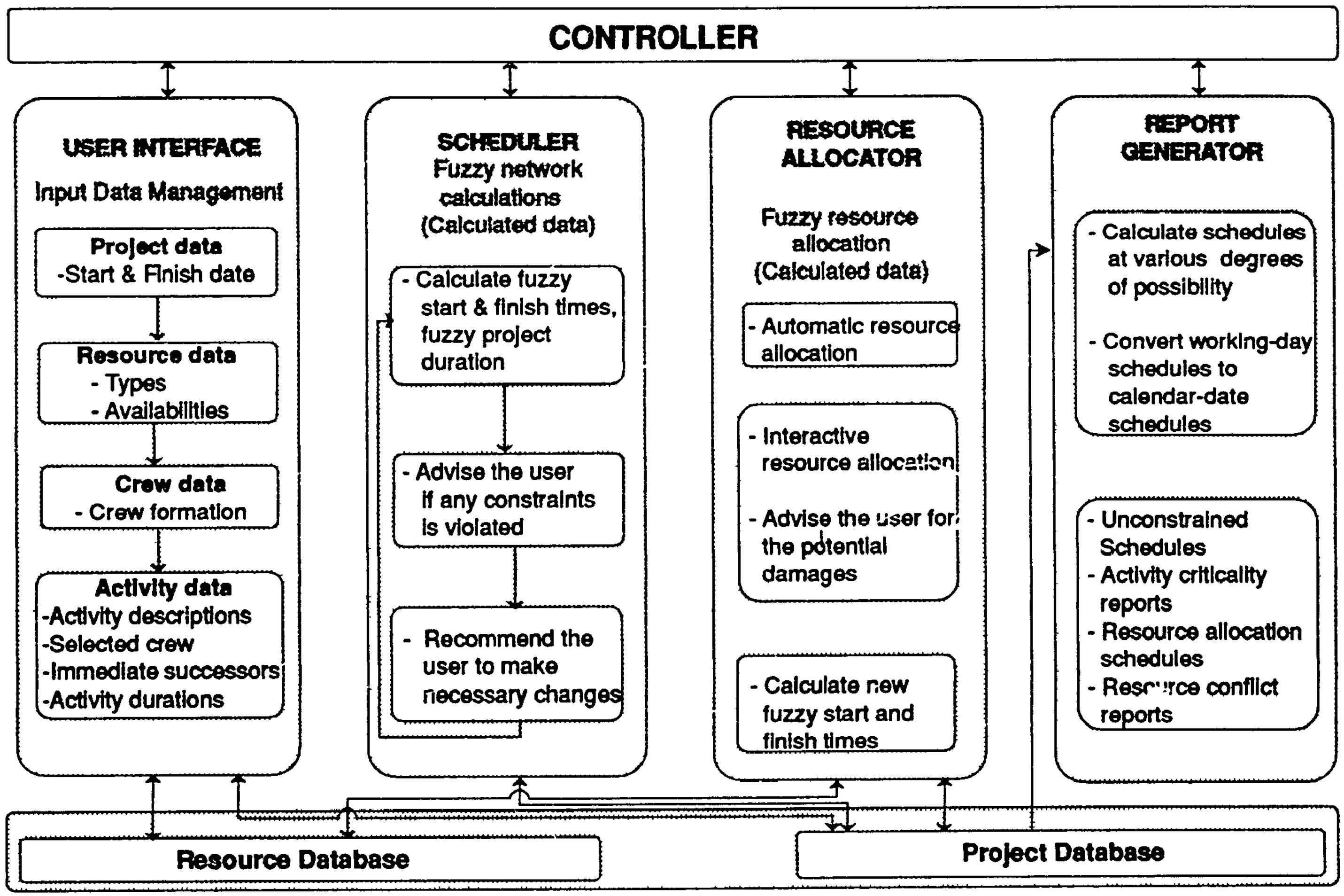


Fig. 6.2: Flow Diagram of RESCONS.

databases. On the other hand, if a new project is to be created, the module prompts the user for the data in each group. The screens of this module are user-friendly. Information boxes are provided, explaining the necessary steps required to furnish the data in each screen.

Fig. 6.3 presents an example of the screen designed to enter a new type of resource. For each resource type, the user provides the description, quantity, and its associated fuzzy availability period. Examples of different membership functions are also given for representing the uncertain availability (See Fig. 6.4). Each membership function requires different input data (e.g. a crisp number requires only one time value while a trapezoidal fuzzy number needs four time values). The user simply clicks on the desired membership function which is created using the "Hyperregion" technique supported by LEVEL5 OBJETCS. Each hyperregion will lead to the appropriate input screen pattern, preventing possible input-errors at that stage.

Once all of the resources are input, the user can form construction crews using these resources. Fig. 6.5 shows an example of the screen used for creating a new crew called "Pile Driver Gang". All of the resources previously input are shown in the list box. The user can select up to ten types of resources for each crew. The selection is done by double-clicking on the designated resource listed in that box. The user then enters the needed quantity associated with this type of resource.

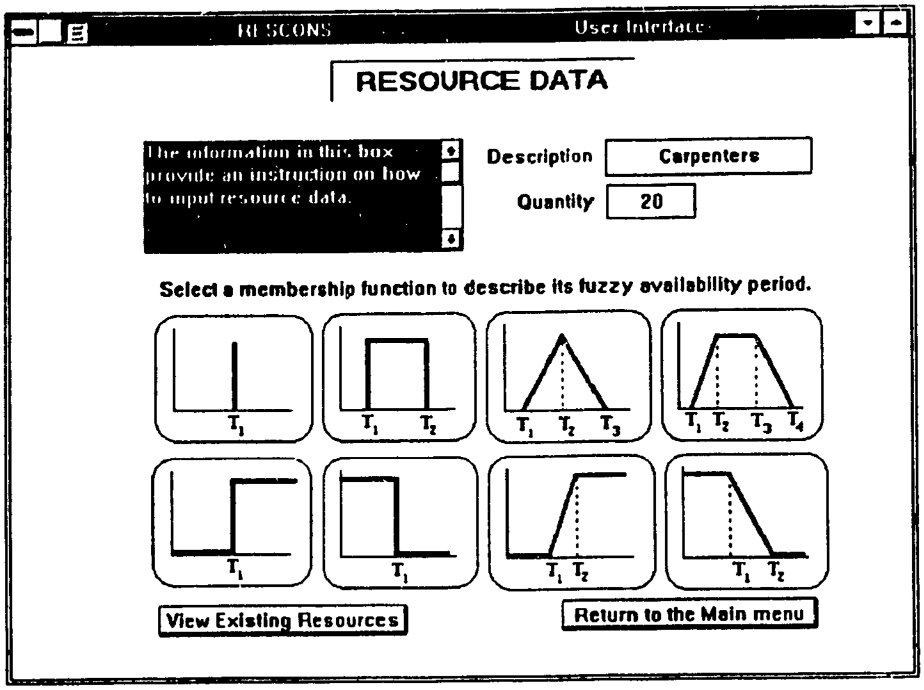


Fig. 6.3: Resource Data Screen.

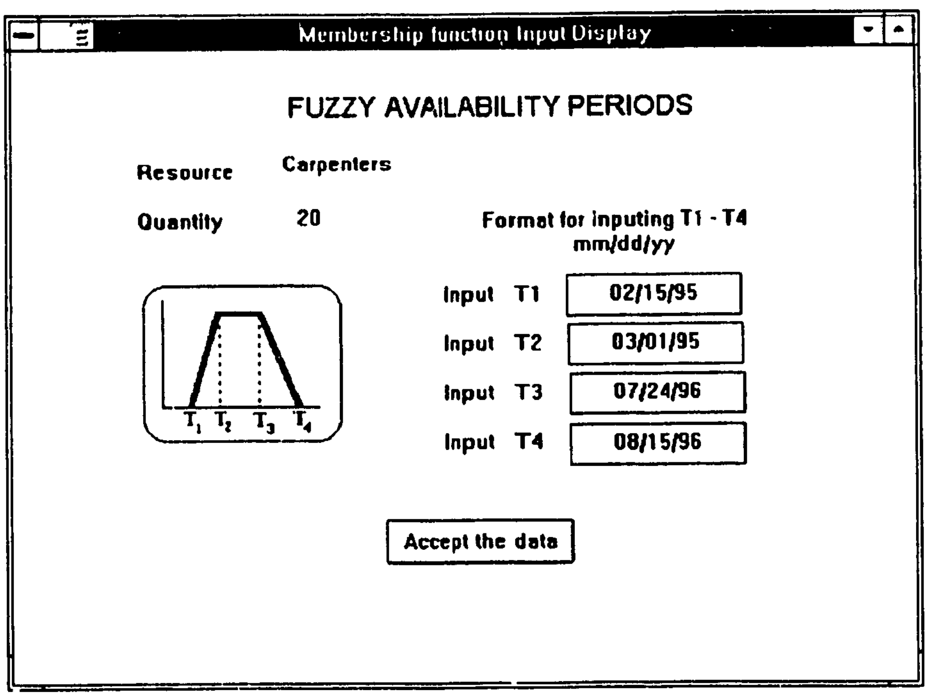


Fig. 6.4: Fuzzy Resource Availability Screen.

Upon confirmation, this resource is included in the crew composition.

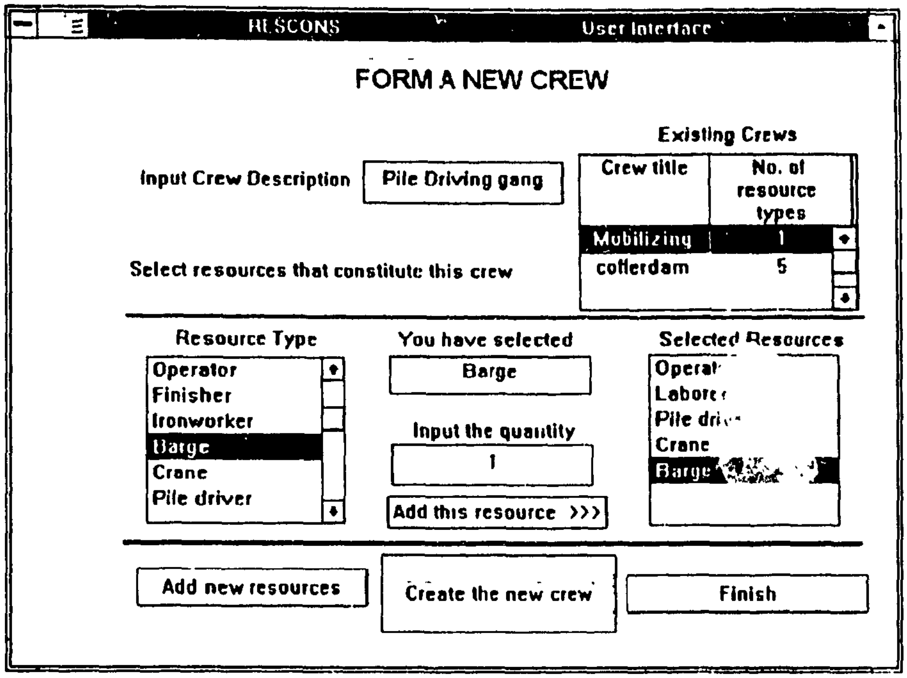


Fig. 6.5: Crew Data Screen.

Once construction crews have been formed, the user can then define project activities. Fig. 6.6 illustrates the input data screen pertaining to each activity in the project. At this stage, only the description of the activity and the crew used to perform it are required as input. The screen includes the list of available existing crews as well as activities already entered. If the needed crew is not available in the list, the user can then define a new crew at this stage. Confirmations of the data being entered are required to prevent possible errors. This feature of the design is used throughout the development of **RESCONS**.

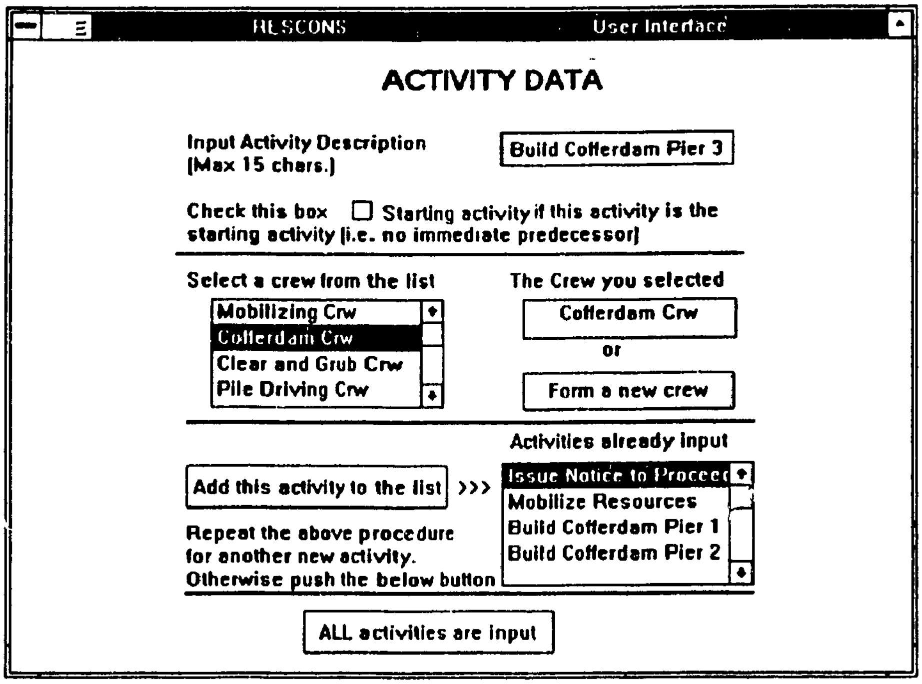


Fig. 6.6: Activity Data Screen.

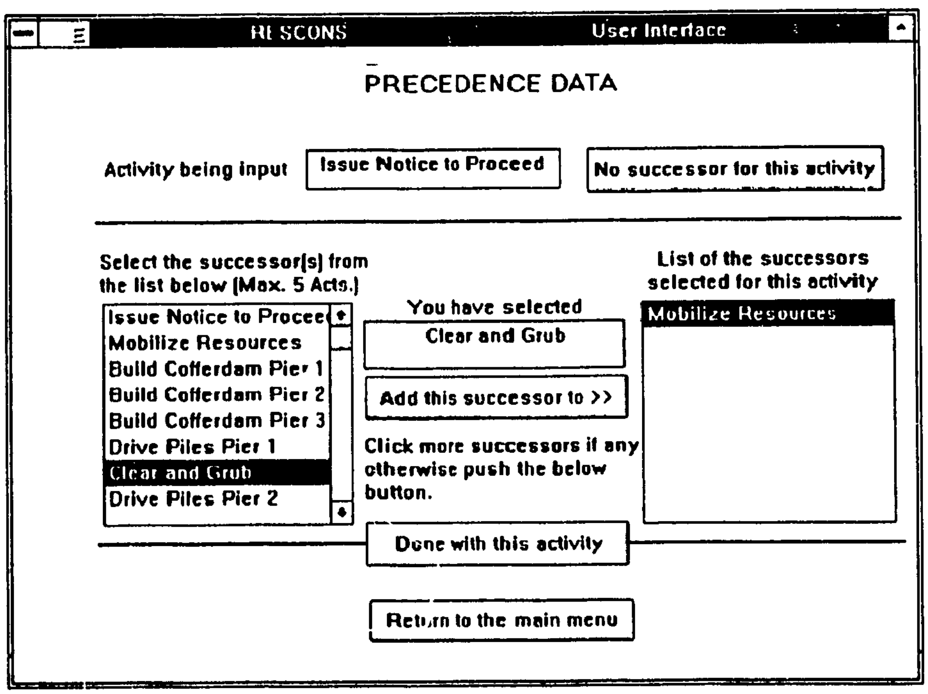


Fig. 6.7: Precedence Constraints Screen.

The next step is to define precedence constraints among the project activities. Fig. 6.7 shows how precedence constraints are input. *User Interface* searches for the activities in which their successors have not been entered, and sequentially bring them up to the user. To specify a successor to an activity, the user simply clicks on the activity shown in the list, one by one, until all of the successors are selected.

The last function of *User Interface* module is to facilitate the input of activity durations. The input of activity durations is carried out in a step-wise manner, similar to that of the forward pass calculations. The module sequentially presents an activity to the user beginning from the start to the end activities. Some useful information pertaining to the estimation of activity durations are given to the user. Fig. 6.8 shows the screen designed for entering activity durations. In this screen, the module displays the activity description (i.e. "Clear and grub") to the user. Along with the description are the crew assigned to perform this activity, a range of approximate start dates and its respective days of the week, the list of activities that could possibly start or already in-progress during this period. All of the provided information is useful in estimating the duration of project activities, taking into consideration the possibility of site congestion, the impact of weather, and the interruptions due to weekends. It should be noted that the program allows the entry of approximate data using fuzzy number representations. Examples of various degrees of fuzzy durations and their respective input format (see Fig. 6.9)

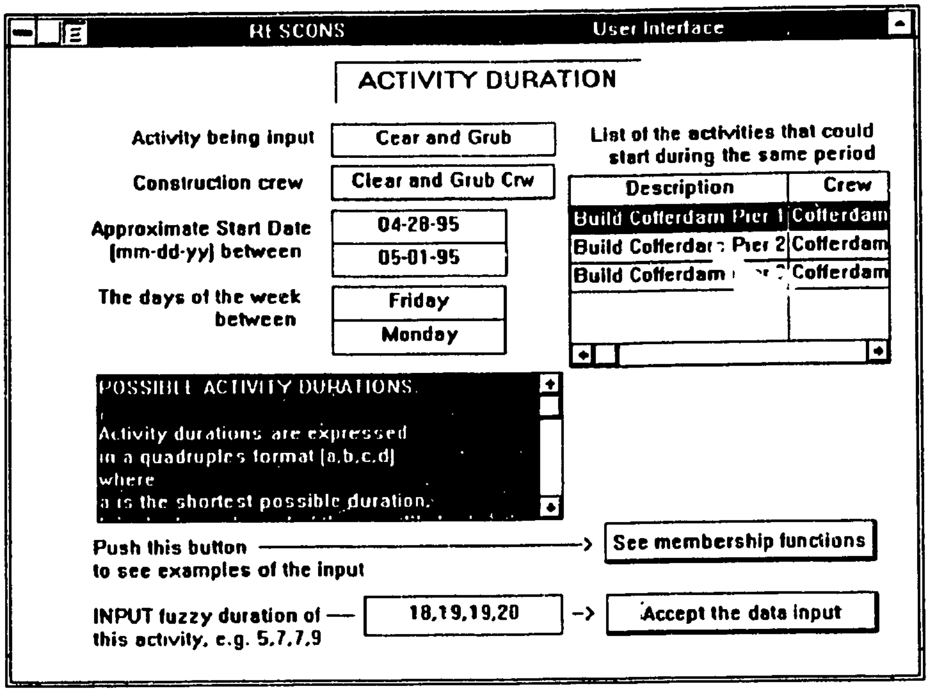


Fig. 6.8: Activity Durations Screen.

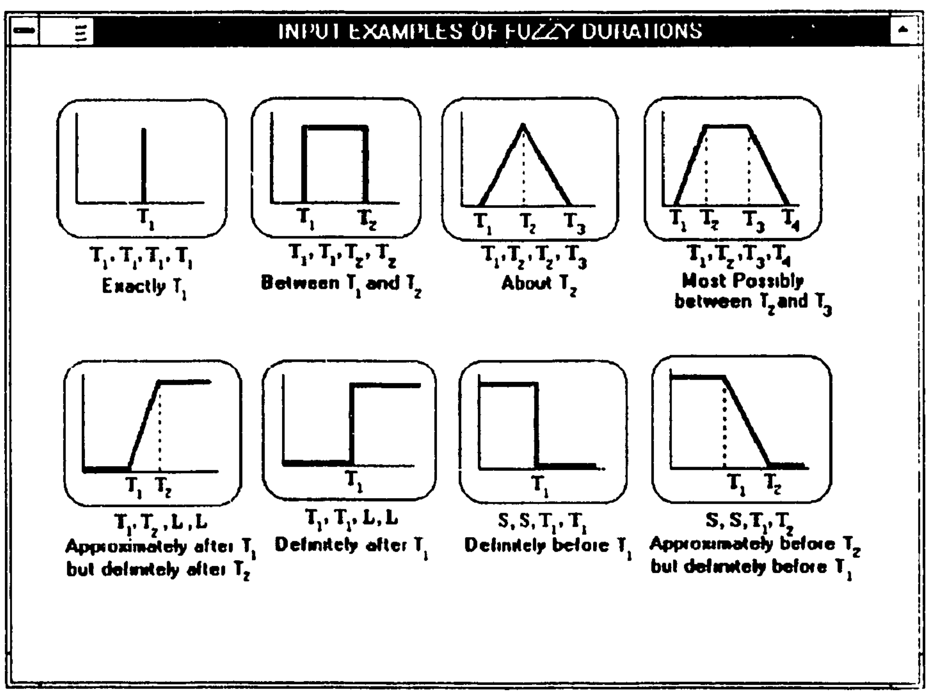


Fig. 6.9: Fuzzy Durations Screen.

can easily be accessed by pushing the "See examples" button provided in that screen.

6.4.2 Scheduler Module

The main function of the *Scheduler* module is to perform fuzzy forward and backward pass calculations. The module calculates the project duration, start and finish times as well as the criticality of each activity using the FNET method described in Chapter 4. When activated by *Controller*, *Scheduler* retrieves the required data of the project being considered from the project database. At this stage, the user can define temporal constraints such as intermediate milestones. Each temporal constraint can be attached to four scheduling parameters: unconstrained start, unconstrained finish, or, constrained start and constrained finish times associated with an activity. Temporal constraints can be crisp or fuzzy numbers. A fuzzy temporal constraint consists of a range of time values together with various degrees of satisfaction associated with each time. The same input format used when entering fuzzy resource availability period are also applicable for specifying temporal constraints.

Scheduler assists the user in generating schedules that satisfy the imposed precedence and temporal constraints. The module performs scheduling in two steps. The first step involves the unconstrained fuzzy scheduling using fuzzy activity durations as originally input by the user. The results of this step is a preliminary fuzzy schedule including fuzzy early and late, start and finish times as well as the criticality (**PM** and **AI** values, see Chapter 4) of project activities. In addition, the module also calculates and displays the degrees of satisfaction associated with each specified temporal constraint. If the user is satisfied with the schedule, the module will save this schedule in the database and return to *Controller*. On the other hand, if the user is not satisfied with the schedule (e.g. certain temporal constraints are not met), he/she would have to modify the duration of some activities and activate *Scheduler*.

Scheduler proceeds with the second step by presenting the user with a list of constraints and their respective degrees of satisfaction. To modify those violated, the user simplv clicks-on the unsatisfied constraint. Accordingly, *Scheduler* provides a list of activities that could contribute to that constraint. *Scheduler* automatically ranks the activities involved according to their cumulative possibility measure obtained from the fuzzy forward pass calculations. The user then modify the duration of the activity in question as desired. Once the modifications are completed, *Scheduler* repeats the first-step network calculations. This 2-step scheduling process is repeated until satisfactory schedule is obtained.

6.4.3 Resource Allocator Module

The main function of *Resource Allocator* is to assist the user in allocating resources to project activities, taking into consideration any resource limitations.

The module operates on the schedules generated by *Scheduler*. *Resource Allocator* supports both automatic and interactive resource allocation operations (Moselhi and Lorterapong 1993d). In the automatic mode, the module allocates resources utilizing the fuzzy resource-constrained scheduling technique described in chapter 4. Upon completion, the user receives a resource-conflict-free schedule. In the interactive mode, on the other hand, *Resource allocator* acts as an assistant to the user. At any scheduling period, the user is presented with two sets of activities: 1) the set of activities that are in-progress, and 2) the set of activities that are ready to start (i.e. eligible activities). The user is allowed to interrupt the activities that are in progress. In so doing, *Resource Allocator* releases the resources consumed by these activities back to the resource pool. The duration required to perform the unexecuted portion of that activity as well as its new fuzzy late start time are calculated. The resource allocation process can then progress.

Given a set of eligible activities, the user can either consult *Resource Allocator* for recommending a set of activities to be scheduled, or manually select the desirable activities from that set. If consulted, *Resource Allocator* will find the best feasible set of activities and present it to the user. If the user agrees with the suggestion made by the system, resources will be allocated and the resource pool is then updated accordingly. In the interactive mode, the user can select his/her activities from the given eligible activity set. The module then examines the feasibility of the set being selected. If the set is feasible, the system calculates the approximate project extension or the remaining total float as a result of scheduling the activities in this set. The user can accept this alternative or experiment with other sets of activities, if desired. It should be noted that if the user is not satisfied with any of the alternatives generated, the user can, at this point, split the activities that are in progress and re-examines new alternatives.

Resource Allocator is capable of resolving resource conflicts in a multiple project environment. In this case, the module retrieves all the data of the projects involved. It then prompts the user for weight factors to be assigned to the individual projects being considered. These weights will be used for calculating the negative impact of the activity being postponed during the resource allocation process (see Chapter 3, section 3.3.3 for more details). It should be noted that the unconstrained scheduling parameters such as the late start or finish times as well as the criticality associated with each activity are project dependent. This way, the user is kept informed about the status of each individual project as the allocation process continue to progress. Examples of the screens designed to facilitate resource allocation in this environment will be presented later in section 6.5 when numerical examples are analyzed.

6.4.4 Report Generator Module

This module presents schedules generated both before and after resource allocation. Schedules produced from *Scheduler* and *Resource Allocator* are saved in the form of the quadruples of working days [e.g. (50,56,59,65)]. One of the main functions of this module is to convert working-day schedules to calendar-day schedules, accounting for weekends and holidays input by the user. Before the conversion can take place, the module prompts the user for an α-cut value (see section 5.4.2 of chapter 5 for more details). The α-cut value is used for calcuiating the range of working days schedule at the appropriate α-cut level. This range is then converted to the calendar-day schedule.

Upon completion of the resource allocation process, the schedule data and analysis results before and after resource allocation are tabulated in various forms of reports. The reports have been designed to provide the user with useful information pertaining to: 1)the possible intervals associated with each scheduled time, 2) the degree of criticality, 3) constraint satisfactions, and 4) the severity of resource conflicts, if any. The activity scheduled times includes fuzzy early start & finish as well as fuzzy late start and finish times. The activity criticality report contains the possibility measure, the agreement index, and their respective cumulative values for each activity. The constraint satisfactions report shows the possibility measure and the agreement index associated with each constraint. In the resource conflict report, the user is given information about the historical record of the resources used in the project. Associated with each type of resource are the frequencies of their conflicts occurrence, the maximum shortage experienced in the resource pool in terms of quantities and the percentage to their respective availabilities.

6.5 Example Applications

Two numerical examples adapted from the literature are worked-out to demonstrate the use and illustrate the capabilities of **RESCONS**. The first example is adopted from Trauner (1993). It is used to illustrate the application of **RESCONS** for scheduling of a construction project. The second involves the scheduling of two engineering, procurement and construction (EPC) type projects (Oberlender 1993). This later example is designed to demonstrate the applicability of **RESCONS** for multiple projects scheduling which is commonly found in most architecture/engineering (A/E) firms. Detailed description of these two projects together with thei analysis results are discussed in the following subsections.

6.5.1 Bridge Construction Project

This project involves the construction of a four-span highway bridge in which its elevation is shown in Fig. 6.10 (Trauner 1993). A PDM project network consisting of 29 activities along with its data are shown in Fig. 6.11. Resource demands and availabilities are summarized in Table 6.1. The contractual start date of this project is assumed to be on March 13, 1995.

RESCONS is used to schedule this project. Project data are entered and saved in the project database. Examples of data input screens included in *User Interface*

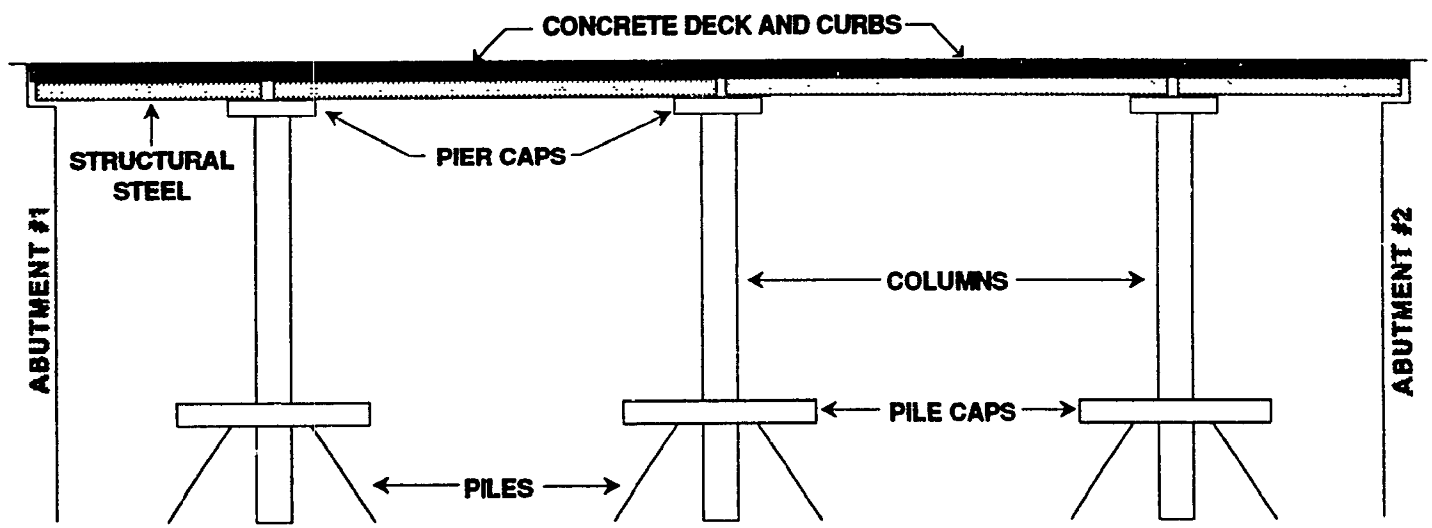


Fig. 6.10: Elevation View of Bridge.

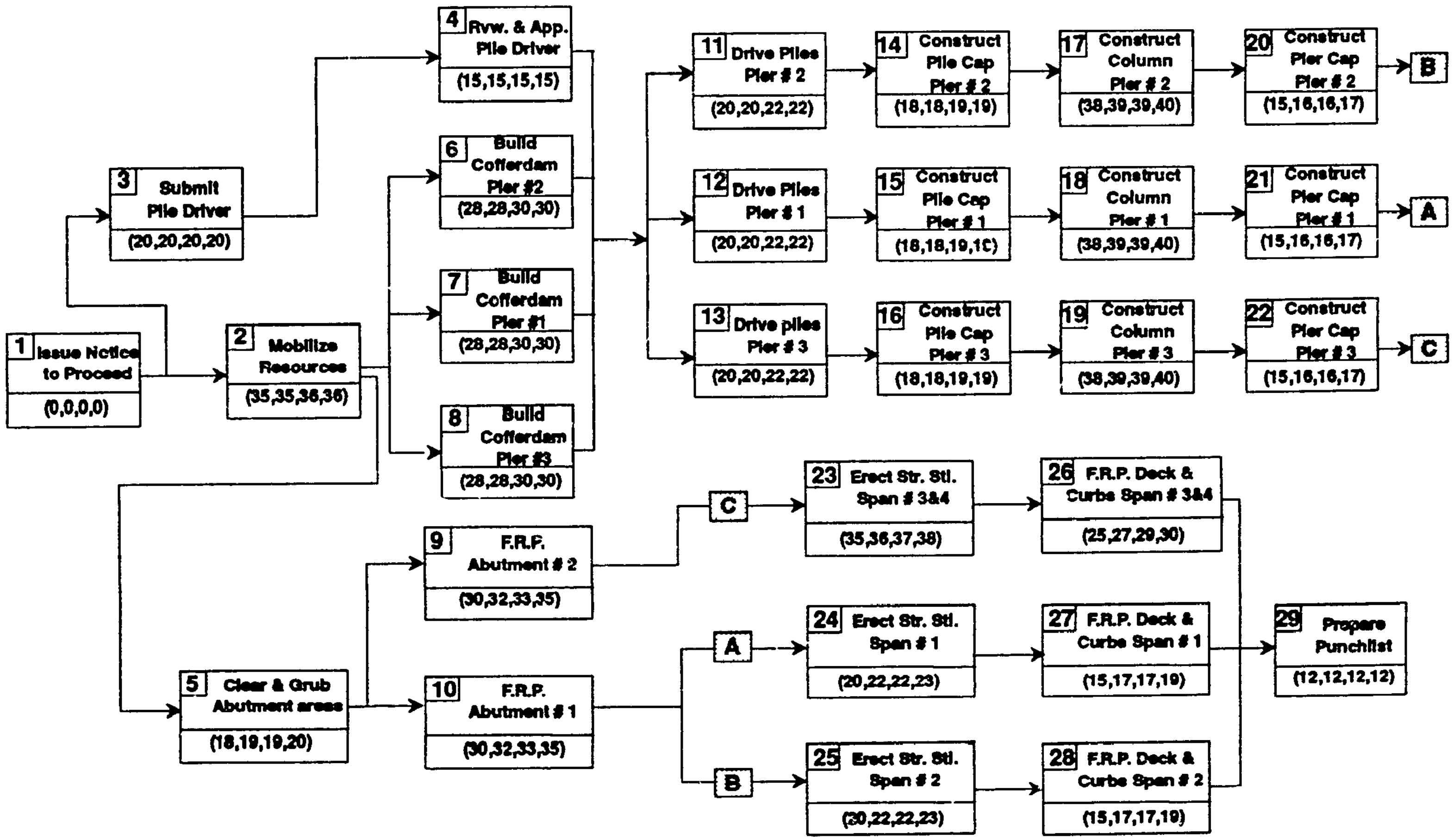


Fig. 6.11: Example Network No. 5.

Table 6.1: Resource Demands and Availabilities for Example Network No. 5.

Activity No.	Resource Types												
	L 16*	C 4*	O 16*	F 8*	I 12*	B 2*	CR 2*	P 2*	G 1*	D 1*	LD 1*	PM 2*	T 5*
1	-	-	-	-	-	-	-	-	-	-	-	-	-
2	8	-	-	-	-	-	-	-	-	-	-	-	-
3	-	-	-	-	-	-	-	-	-	-	-	-	-
4	-	-	-	-	-	-	-	-	-	-	-	-	-
5	2	-	2	-	-	-	-	-	1	1	1	-	1
6	6	-	4	-	-	-	1	1	-	-	-	-	1
7	6	-	4	-	-	-	1	1	-	-	-	-	1
8	6	-	4	-	-	-	1	1	-	-	-	-	1
9	8	1	-	4	2	-	1	1	-	-	-	-	-
10	8	1	-	4	2	-	1	1	-	-	-	-	-
11	4	-	8	-	-	1	1	1	-	-	-	-	-
12	4	-	8	-	-	1	1	1	-	-	-	-	-
13	4	-	8	-	-	1	1	1	-	-	-	-	-
14	2	2	2	2	2	1	1	-	-	-	-	-	-
15	2	2	2	2	2	1	1	-	-	-	-	-	-
16	2	2	2	2	2	1	1	-	-	-	-	-	-
17	2	2	2	2	2	1	1	-	-	-	-	-	-
18	2	2	2	2	2	1	1	-	-	-	-	-	-
19	2	2	2	2	2	1	1	-	-	-	-	-	-
20	2	2	2	2	2	1	1	-	-	-	-	-	-
21	2	2	2	2	2	1	1	-	-	-	-	-	-
22	2	2	2	2	2	1	1	-	-	-	-	-	-
23	-	-	1	-	6	1	1	-	-	-	-	-	-
24	-	-	1	-	6	1	1	-	-	-	-	-	-
25	-	-	1	-	6	1	1	-	-	-	-	-	-
26	3	2	2	4	4	-	1	-	-	-	-	1	-
27	3	2	2	4	4	-	1	-	-	-	-	1	-
28	3	2	2	4	4	-	1	-	-	-	-	1	-
29	-	-	-	-	-	-	-	-	-	-	-	-	-

Legend:
L = Laborer; C = Carpenter; O = Operator; F = Finisher; I = Iron-worker; B = Barge;
CR = Crane; P = Pile-driver; G = Grader; D = Dozer; LD = Loader;
PM = Paving machine; and T = Dump truck
*The available quantities associated with each type of resources.

have been described earlier. After all the data are input, *Scheduler* is invoked to perform unconstrained scheduling. Upon completion of the scheduling efforts, *Report Generator* is executed. Fig 6.12 shows the unconstrained schedule at an α-cut = 1.0 (i.e. the highest possible schedule). The unconstrained project duration at this α-cut level is calculated to be somewhere between Jan. 29, and Feb. 9, 1996. The criticality of project activities are depicted in Fig. 6.13. Activities that have their PM and AI values equal to one are regarded as critical activities. Among these critical activities, those with a higher cumulative degree of criticality should draw the attention of the project team in order to avoid possible schedule slippage. The cumulative degree of criticality can be used as an indicator for selecting activities to be crashed in the situation where the calculated project duration is longer than required.

Once the unconstrained schedule is accepted, resource allocation can begin. The user has two options: automatic or interactive resource allocations. To demonstrate the interactive capabilities of **RECONS**, the interactive mode is selected. Fig. 6.14 shows a typical situation, where, at a current time (63,63,66,66) days, three activities (i.e. activities Build Cofferdam Pier 3, FRP Abutment 1, and FRP Abutment 2) are ready to start. In stead of permitting **RESCONS** to select the best activities, the user can introduce his/her own alternatives by pushing the "User" button shown in Fig. 6.14. In so doing, a new screen is then prompted to the user (see Fig. 6.15). The user can then form different activity sets and evaluate each

RESCONS — REPORT GENERATOR

UNCONSTRAINED SCHEDULE

Project title: BRIDGE — Degree of possibility: 1

Description	ES	EF	LS	LF
Issue Notice to proceed	03-13 , 03-13	03-13 , 03-13	03-13 , 03-13	03-13 , 03-13
Mobilize Resources	03-13 , 03-13	04-28 , 05-01	03-13 , 03-13	04-28 , 05-01
Submit Pile Driver	03-13 , 03-13	04-07 , 04-07	04-24 , 04-24	05-22 , 05-22
Rvw. & App. Pile Driver	04-07 , 04-07	04-28 , 04-28	05-22 , 05-22	06-12 , 06-12
Clear & Grub Abutment areas	04-28 , 05-01	05-25 , 05-26	08-09 , 08-10	09-05 , 09-06
Build Cofferdam Pier 2	04-28 , 05-01	06-07 , 06-12	04-28 , 05-01	06-07 , 06-12
Build Cofferdam Pier 1	04-28 , 05-01	06-07 , 06-12	04-28 , 05-01	06-07 , 06-12
Build Cofferdam Pier 3	04-28 , 05-01	06-07 , 06-12	04-28 , 05-01	06-07 , 06-12
FRP Abutment 2	05-25 , 05-26	07-10 , 07-12	09-05 , 09-06	10-19 , 10-23
FRP Abutment 1	05-25 , 05-26	07-10 , 07-12	10-11 , 10-12	11-24 , 11-28
Drive Piles Pier 2	06-07 , 06-12	07-05 , 07-12	07-13 , 07-18	08-10 , 08-17
Drive Piles Pier 1	06-07 , 06-12	07-05 , 07-12	07-13 , 07-18	08-10 , 08-17
Drive Piles Pier 3	06-07 , 06-12	07-05 , 07-12	06-07 , 06-12	07-05 , 07-12
Construct Pile Cap Pier 2	07-05 , 07-12	07-31 , 08-08	08-10 , 08-17	09-05 , 09-13

Project completion date: 01-29 , 02-09 — Criticality — Return

RESCONS — REPORT GENERATOR

UNCONSTRAINED SCHEDULE

Project title: BRIDGE — Degree of possibility: 1

Description	ES	EF	LS	LF
Construct Pile Cap Pier 3	07-05 , 07-12	07-31 , 08-08	07-05 , 07-12	07-31 , 08-08
Construct Column Pier 2	07-31 , 08-08	09-22 , 10-02	09-05 , 09-13	10-30 , 11-07
Construct Column Pier 1	07-31 , 08-08	09-22 , 10-02	09-05 , 09-13	10-30 , 11-07
Construct Column Pier 3	07-31 , 08-08	09-22 , 10-02	07-31 , 08-08	09-22 , 10-02
Construct Pier Cap Pier 2	09-22 , 10-02	10-16 , 10-24	10-30 , 11-07	11-21 , 11-29
Construct Pier Cap Pier 1	09-22 , 10-02	10-16 , 10-24	10-30 , 11-07	11-21 , 11-29
Construct Pier Cap Pier 3	09-22 , 10-02	10-16 , 10-24	09-22 , 10-02	10-16 , 10-24
Erect Str Steel Span 3&4	10-16 , 10-24	12-05 , 12-14	10-16 , 10-24	12-05 , 12-14
Erect Str Steel Span 1	10-16 , 10-24	11-15 , 11-23	11-21 , 11-29	12-21 , 12-29
Erect Str Steel Span 2	10-16 , 10-24	11-15 , 11-23	11-21 , 11-29	12-21 , 12-29
FRP Deck Span 3&4	12-05 , 12-14	01-11 , 01-24	12-05 , 12-14	01-11 , 01-24
FRP Deck Span 1	11-15 , 11-23	12-08 , 12-18	12-21 , 12-29	01-15 , 01-23
FRP Deck Span 2	11-15 , 11-23	12-08 , 12-18	12-21 , 12-29	01-15 , 01-23
Prepare Punchlist	01-11 , 01-24	01-29 , 02-09	01-11 , 01-24	01-29 , 02-09

Project completion date: 01-29 , 02-09 — Criticality — Return

Fig. 6.12: Unconstrained Schedules for Example Network No. 5.

RESCONS — REPORT GENERATOR

DEGREE OF CRITICALITY

Project: BRIDGE

Description	TF	PM	Sum of PM	AI	Sum of AI
Issue Notice to proceed	0.00	1.00	3.00	1.00	3.00
Mobilize Resources	0.00	1.00	3.00	1.00	3.00
Submit Pile Driver	31.00	0.00	0.00	0.00	0.00
Rvw. & App. Pile Driver	31.00	0.00	0.00	0.00	0.00
Clear & Grub Abutment areas	73.00	0.00	0.00	0.00	0.00
Build Cofferdam Pier 2	0.00	1.00	1.00	1.00	1.00
Build Cofferdam Pier 1	0.00	1.00	1.00	1.00	1.00
Build Cofferdam Pier 3	0.00	1.00	1.00	1.00	1.00
FRP Abutment 2	73.00	0.00	0.00	0.00	0.00
FRP Abutment 1	99.00	0.00	0.00	0.00	0.00
Drive Piles Pier 2	26.00	0.00	0.00	0.00	0.00
Drive Piles Pier 1	26.00	0.00	0.00	0.00	0.00
Drive Piles Pier 3	0.00	1.00	3.00	1.00	3.00
Construct Pile Cap Pier 2	26.00	0.00	0.00	0.00	0.00
Construct Pile Cap Pier 1	26.00	0.00	0.00	0.00	0.00

PM = Possibility Measure; AI = Agreement Index

Return

RESCONS — REPORT GENERATOR

DEGREE OF CRITICALITY

Project: BRIDGE

Description	TF	PM	Sum of PM	AI	Sum of AI
Construct Pile Cap Pier 1	26.00	0.00	0.00	0.00	0.00
Construct Pile Cap Pier 3	0.00	1.00	3.00	1.00	3.00
Construct Column Pier 2	26.00	0.00	0.00	0.00	0.00
Construct Column Pier 1	26.00	0.00	0.00	0.00	0.00
Construct Column Pier 3	0.00	1.00	3.00	1.00	3.00
Construct Pier Cap Pier 2	26.00	0.00	0.00	0.00	0.00
Construct Pier Cap Pier 1	26.00	0.00	0.00	0.00	0.00
Construct Pier Cap Pier 3	0.00	1.00	3.00	1.00	3.00
Erect Str Steel Span 3&4	0.00	1.00	3.00	1.00	3.00
Erect Str Steel Span 1	26.00	0.00	0.00	0.00	0.00
Erect Str Steel Span 2	26.00	0.00	0.00	0.00	0.00
FRP Deck Span 3&4	0.00	1.00	3.00	1.00	3.00
FRP Deck Span 1	26.00	0.00	0.00	0.00	0.00
FRP Deck Span 2	26.00	0.00	0.00	0.00	0.00
Prepare Punchlist	0.00	1.00	3.00	1.00	3.00

PM = Possibility Measure; AI = Agreement Index

Return

Fig. 6.13: Criticality of Activities for Example Network No. 5.

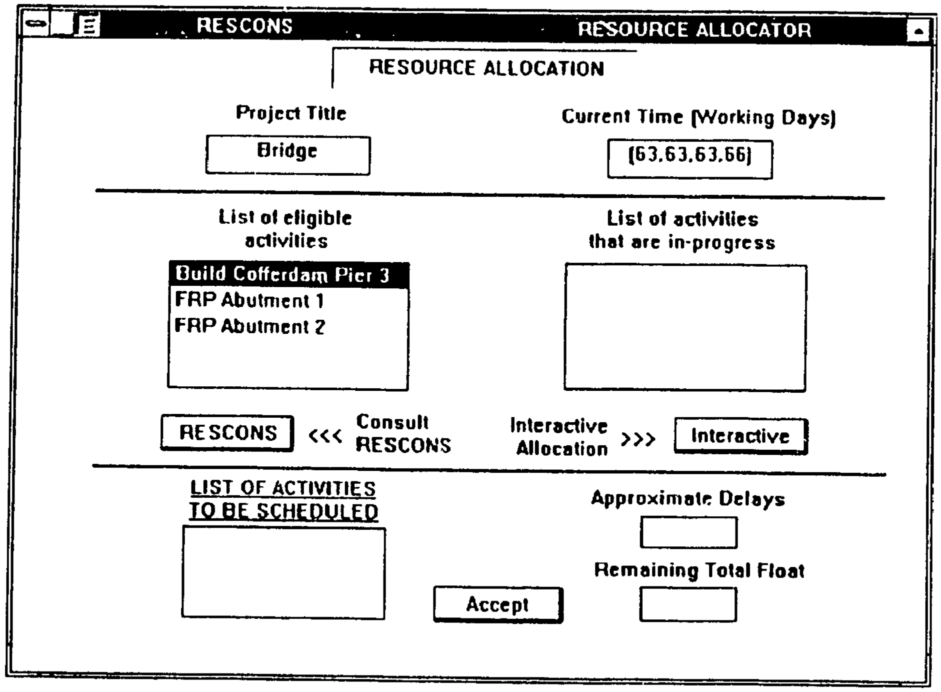


Fig. 6.14 Typical Resource Conflict Situation Screen.

set individually. The user can still seek a recommendation from **RESCONS** by pushing the "RESCONS" button. In the example shown in Fig. 6.15, the system suggests that activities "Build Cofferdam Pier 3" and "FRP Abutment 2" should be scheduled with a remaining total float of 60 days (i.e. no delay occurs).

Suppose that the user, for some reason, would like to schedule activity "FRP Abutment 1" in this period. As a result, two alternatives can be generated from the given eligible set of activities. The first alternative consists of activities "FRP Abutment 1" and "FRP Abutment 2" (Fig. 6.15). The system evaluates this

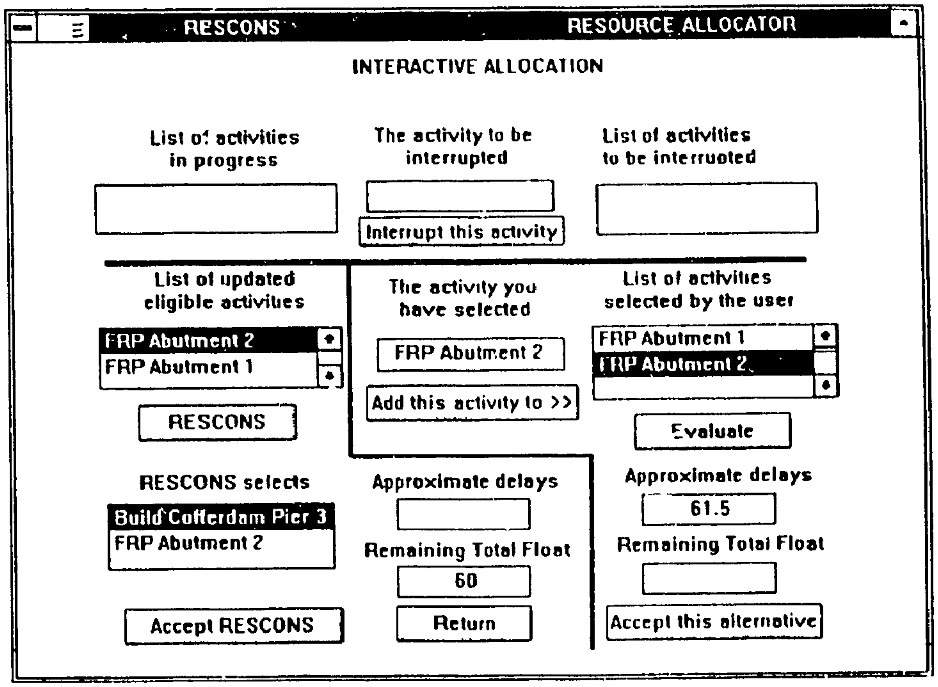


Fig. 6.15: Interactive Resource Allocation (a).

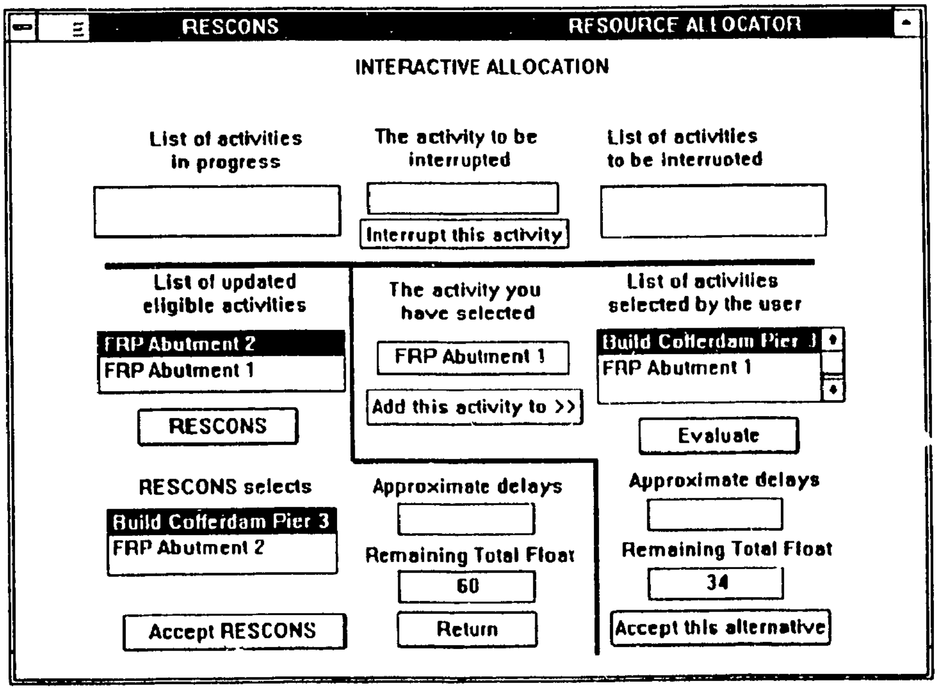


Fig. 6.16: Interactive Resource Allocation (b).

alternative and finds that it is feasible but will extend the project duration for 61.5 days. The user then forms and evaluates the second alternative which includes activities "Build Cofferdam Pier 3" and "FRP Abutment 1" (Fig. 6.16). **RESCONS** finds the second alternative also feasible. The remaining total float of this alternative is calculated to be 34 days (i.e. no delay occurs). Since the second alternative is better, the user then decides to schedule the activities in this alternative.

Another interesting feature of **RESCONS** is to allow the user to interrupt activities that are in-progress. Fig. 6.17 shows an example of how this task can be performed. The situation depicted in Fig. 6.17 involves three activities (i.e. "Drive Piles Pier 2", "Drive Piles Pier 3", and "FRP Abutment 2" ready to start and one activity "FRP Abutment 1", in-progress. Due to the limitation of cranes, only one activity can be scheduled at this time if activity "FRP Abutment 1" continues to progress. The user in this case decides to interrupt activity "FRP Abutment 1" and allows **RESCONS** to select the best alternative. **RESCONS** recommends that two activities (i.e. Drive Piles Pier 2 and Drive Piles Pier 3) be scheduled, causing the approximate total delays of 24 days.

After resource allocation is completed, *Report Generator* is activated. Fig. 6.18 shows the most possible schedule after allocating resources calculated at an α-cut = 1.0. The project duration is then calculated to be between June 27, and July 10

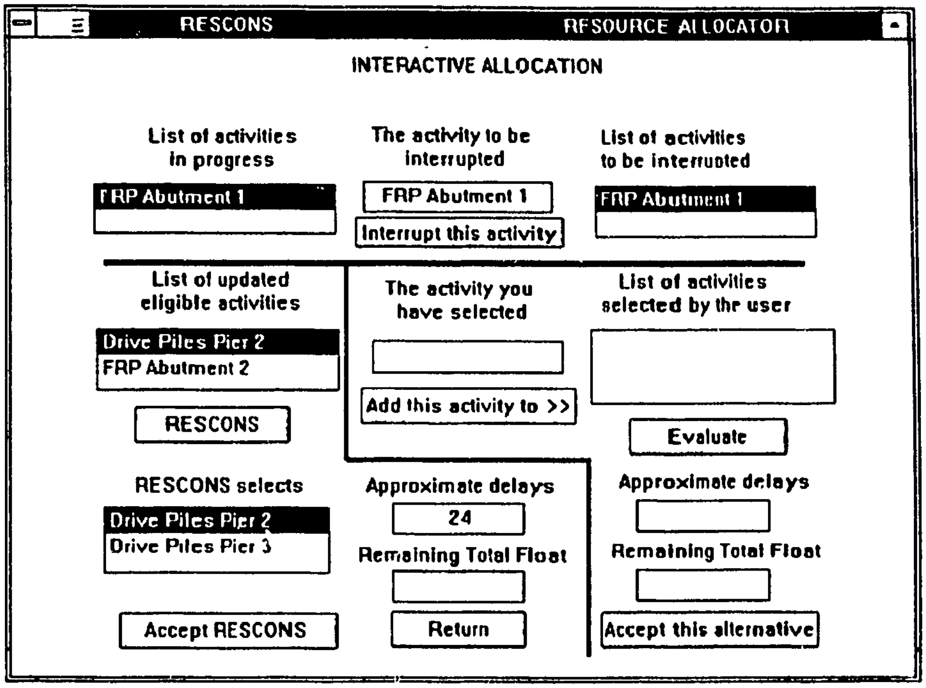


Fig. 6.17: Demonstration of Activity Interruptions.

1996. Fig. 6.19 summarizes the history of resource allocation for this project. It can be stated that the project delays can partially be attributed to the insufficient number of cranes and barges. At some points, five cranes are needed while only two of them are available. This information can be useful in the case where the user would like to accelerate the project execution time.

RESCONS REPORT GENERATOR

AFTER ALLOCATION

Project title: Bridge Degree of Possibility: 1.00

Description	Scheduled Start	Scheduled Finish
Issue Notice to Proceed	03-13 , 03-13	03-13 , 03-13
Mobilize Resources	03-13 , 03-13	04-28 , 05-01
Submit Pile driver	03-13 , 03-13	04-07 , 04-07
Review Pile Driver	04-07 , 04-07	04-28 , 04-28
Clear and Grub	04-28 , 05-01	05-25 , 05-26
Build Cofferdam Pier 1	04-28 , 05-01	06-07 , 06-12
Build Cofferdam Pier 2	04-28 , 05-01	06-07 , 06-12
Build Cofferdam Pier 3	06-07 , 06-12	07-07 , 07-24
FRP Abutment 2	11-01 , 11-13	12-15 , 12-28
FRP Abutment 1	06-07,06-12 10-31,11-15	07-17,07-24 11-06,11-21
Drive Piles Pier 2	07-17 , 07-24	08-14 , 08-23
Drive Piles Pier 1	08-14 , 08-23	09-11 , 09-22
Drive Piles Pier 3	07-07 , 07-24	08-14 , 08-23
Construct Pile Cap Pier 2	09-11 , 09-22	10-05 , 10-19
Construct Pile Cap Pier 1	10-05 , 10-19	10-31 , 11-15

Project completion date after resource allocation: 06-27 , 07-10 Return

RESCONS REPORT GENERATOR

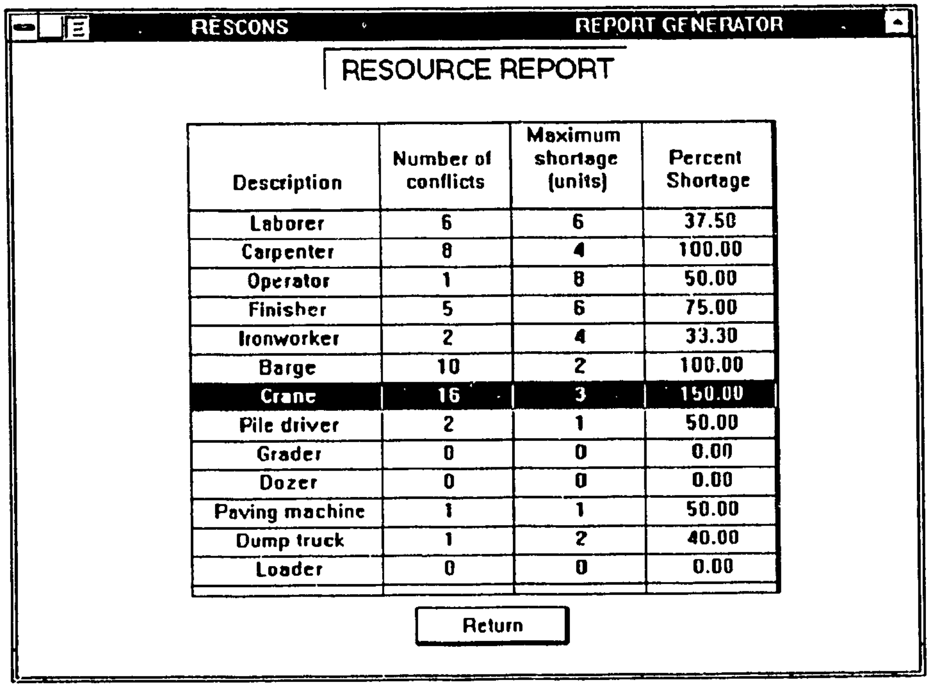
AFTER ALLOCATION

Project title: Bridge Degree of Possibility: 1.00

Description	Scheduled Start	Scheduled Finish
Construct Pile Cap Pier 1	10-05 , 10-19	10-31 , 11-15
Construct Pile Cap Pier 3	08-14 , 08-23	09-07 , 09-19
Construct Column Pier 2	11-28 , 12-13	01-22 , 02-06
Construct Column Pier 1	12-15 , 12-28	02-08 , 02-21
Construct Column Pier 3	09-07 , 09-19	11-01 , 11-13
Construct Pier Cap Pier	03-01 , 03-14	03-25 , 04-05
Construct Pier Cap Pier	02-08 , 02-21	03-01 , 03-14
Construct Pier Cap Pier	11-06 , 11-21	11-28 , 12-13
Erect Str Stl 3&4	01-22 , 02-06	03-12 , 03-28
Erect Str Stl 1	03-12 , 03-28	04-11 , 04-29
Erect Str Stl 2	03-25 , 04-05	04-25 , 05-07
FRP Deck 3&4	04-11 , 04-29	05-20 , 06-07
FRP Deck 1	04-24 , 05-07	05-17 , 05-30
FRP Deck 2	05-17 , 05-30	06-11 , 06-24
Prepare Punchlist	06-11 , 06-24	06-27 , 07-10

Project completion date after resource allocation: 06-27 , 07-10 Return

Fig. 6.18: Schedule After Resource Allocation for Example Network No. 5.



Description	Number of conflicts	Maximum shortage (units)	Percent Shortage
Laborer	6	6	37.50
Carpenter	8	4	100.00
Operator	1	8	50.00
Finisher	5	6	75.00
Ironworker	2	4	33.30
Barge	10	2	100.00
Crane	16	3	150.00
Pile driver	2	1	50.00
Grader	0	0	0.00
Dozer	0	0	0.00
Paving machine	1	1	50.00
Dump truck	1	2	40.00
Loader	0	0	0.00

Fig. 6.19: Resource Conflict Report.

6.5.2 EPC Projects

The second example is also adapted from the literature (Oberlender 1993). It is used for illustrating the application of **RESCONS** in a multiple project environment. Suppose that a small A/E firm is already committed to a design project (i.e. Project A) with a scheduled target completion date has been established. Let us assume that the design activities of project A are scheduled to start on March 13, 1995 and to finish no later than July 3, 1995. This delivery date is established in order to comply with the bidding, procurement, and construction commitments as specified by the owner. At the mean time, the management of this company is considering whether or not to accept a new design project (i.e. Project B) which will also be carried out using the in-house resources. If project B is undertaken it is specified that the start date is set on February 13, 1995 and the completion of all design activities must be finished no later than June 9, 1995.

Fig. 6.20 shows the precedence network diagrams and their estimated activity durations. Resource demands and availabilities for the two projects are summarized in Table 6.2. **RESCONS** is used first to perform unconstrained network calculations for both projects. The results are presented in Fig. 6.21a. In the case of project A, the possible early completion dates of the design work is calculated to be between June 1 and June 15, 1995. It is clear that project A could be finished well ahead of its contractual completion date (i.e. July 3, 1995). At this point it is important to investigate whether the A/E firm could complete the work of the two projects simultaneously without any contractual delays. For that, the completion date of project A is considered to coincide with its contractual completion date which is used in this case as the beginning point for backward pass calculations. As a result, most of the activities in project A will have floats. These floats can be utilized for the scheduling of project B. In the case of project B, the completion date resulting from the forward pass calculation [i.e. May 19 and June 8 1995 (see Fig. 6.21b)] is very close to that stipulated in the contract (i.e. June 9, 1995). It is decided that the calculated completion dates are used in the backward pass calculation. This is done to illustrate the flexibility of the system in

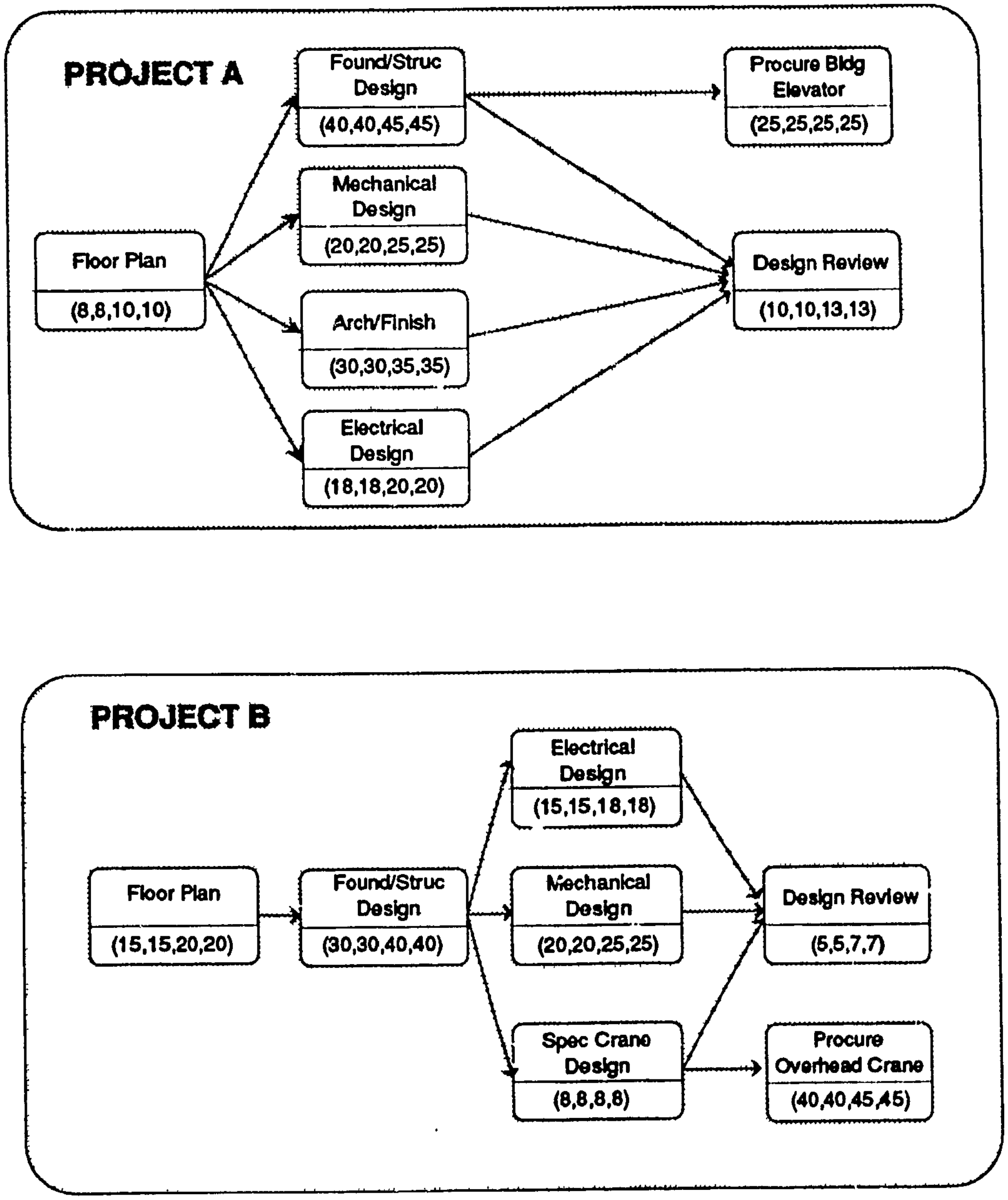


Fig. 6.20: Example Network No. 6.

Table 6.2: Resource Demands and Availabilities for Example Network No. 6.

Activities	Architects A,B,C	Civil Engineers D,E	Mechanical Engineers F,G,H	Electrical Engineers I,J,K	Draftsmen 15
PROJECT A					
Floor plan	A	-	-	-	1
Found/Struct design	-	D	-	-	2
Arch/Finish	A,B	-	-	-	2
Mechanical design	-	-	F,G	-	3
Electrical design	-	-	-	I,J	2
Procure Bldg. elevator	-	-	-	-	-
Design review	C	E	H	K	1
PROJECT B					
Floor plan	A	-	-	-	2
Found/Struct design	-	E	-	-	2
Mechanical design	-	-	H,G	-	2
Electrical design	-	-	-	J,K	2
Spec Crane design	-	-	F	-	2
Design review	B or C	D	F	I	1
Procure overhead crane	-	-	-	-	-

RESCONS — REPORT GENERATOR

UNCONSTRAINED SCHEDULE

Project title: EPC Project A — Degree of posibility: 1.00

Description	ES	EF	LS	LF
Floor Plan	03-13, 03-13	03-23, 03-27	03-13, 03-13	03-23, 03-27
Found/Struc Design	03-23, 03-27	05-18, 05-29	03-23, 03-27	05-18, 05-29
Mechanical Design	03-23, 03-27	04-20, 05-01	05-08, 05-10	06-05, 06-14
Arch/Finish Design	03-23, 03-27	05-04, 05-15	04-24, 04-26	06-05, 06-14
Electrical Design	03-23, 03-27	04-18, 04-24	05-15, 05-17	06-08, 06-14
Proc Bldg Elevator	05-18, 05-29	06-22, 07-03	05-18, 05-29	06-22, 07-03
Design Review	05-18, 05-29	06-01, 06-15	06-05, 06-14	06-19, 07-03

Project completion date: 06-22, 07-03 — Criticality — Return

Fig. 6.21 Unconstrained Schedule of EPC Project (a).

RESCONS — REPORT GENERATOR

UNCONSTRAINED SCHEDULE

Project title: EPC Project B — Degree of posibility: 1.00

Description	ES	EF	LS	LF
Floor Plan	02-13, 02-13	03-03, 03-10	02-13, 02-13	03-03, 03-10
Found/Struc	03-03, 03-10	04-14, 04-25	03-03, 03-10	04-14, 04-25
Electrical Design	04-14, 04-25	05-05, 05-19	04-25, 05-04	05-16, 05-30
Mechanical Design	04-14, 04-25	05-12, 05-30	04-14, 04-25	05-12, 05-30
Spec Crane Design	04-14, 04-25	04-26, 05-05	04-14, 04-25	04-26, 05-05
Design Review	05-12, 05-30	05-19, 06-08	05-12, 05-30	05-19, 06-08
Proc Overhead	04-26, 05-05	06-21, 07-07	04-26, 05-05	06-21, 07-07

Project completion date: 06-21, 07-07 — Criticality — Return

Fig. 6.21: Unconstrained Schedule of EPC Project (b).

permitting the user to make such judgement.

Resource Allocator is then activated to perform resource allocation. The scheduler has three options: 1)automatic allocation without activity splitting, 2)automatic allocation with interruption, and 3) interactive allocation. These options are shown in Fig. 6.22.

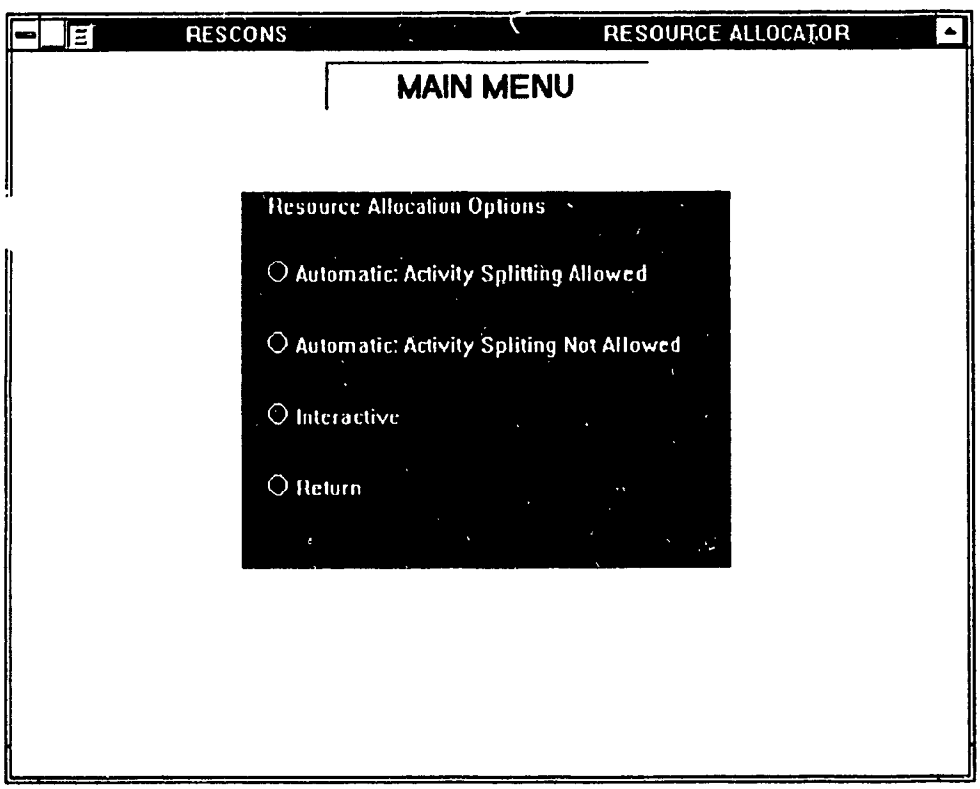


Fig. 6.22: Resource Allocation Options.

Since the interactive resource allocation has already been demonstrated in the first example, the automatic options are used herein. It is interesting to examine first the schedule in which activity splitting is not permitted. Fig. 6.23 illustrates the results of projects A and B after resource allocation. It can be seen from Fig. 6.23a that, without activity splitting, project A can be completed anytime between June 1 and 22, 1995, well ahead the contractual date of July 3, 1995. On the contrary, project B is calculated to be finished between May 25 and June 14, 1995 (see Fig. 6.23b). Accordingly, there is a possibility that project B could be finished behind its contractual date.

Another scheduling scenario is therefore examined, this time allowing activity interruptions. Figs. 6.24 a and b show the schedules of the two projects for the scenario in which activity interruptions are permitted. With the interruptions of activities "Mechanical design" and "Electrical Design", project A can be finished between June 8 and July 4, 1995 (see Fig. 6.24a). Also, there is a slight possibility that project A could be finished one day behind its contractual date (July 3, 1995). With this option, however, project B can be completed between May 19 and June 8, 1995 (see Fig. 6.24b), before its contractual date of June 9, 1995.

The constraint satisfaction reports for the nonsplitting and splitting scenarios shown respectively in Figs. 6.25 a and b can be used to evaluate the merits of these two schedules. If activity splitting is not permitted there is a possibility that the

RESCONS — REPORT GENERATOR

AFTER ALLOCATION

Project title: EPC Project A — Degree of posibility: 1.00

Description	SCHDULED START	SCHEDULED FINISH
Floor Plan	03-13 , 03-13	03-23 , 03-27
Found/Struc Design	03-23 , 03-27	05-18 , 05-29
Mechanical Design	03-23 , 03-27	04-20 , 05-01
Arch/Finish Design	03-23 , 03-27	04-18 , 04-24
Electrical Design	03-23 , 03-27	04-18 , 04-24
Proc Bldg Elevator	05-18 , 05-29	06-22 , 07-23
Design Review	05-18 , 06-05	06-01 , 06-22

Project completion date after resource allocation: 06-22 , 07-23 — Return

Fig. 6.23a: Schedule After Resource Allocation of EPC Project (No Interruptions).

RESCONS — REPORT GENERATOR

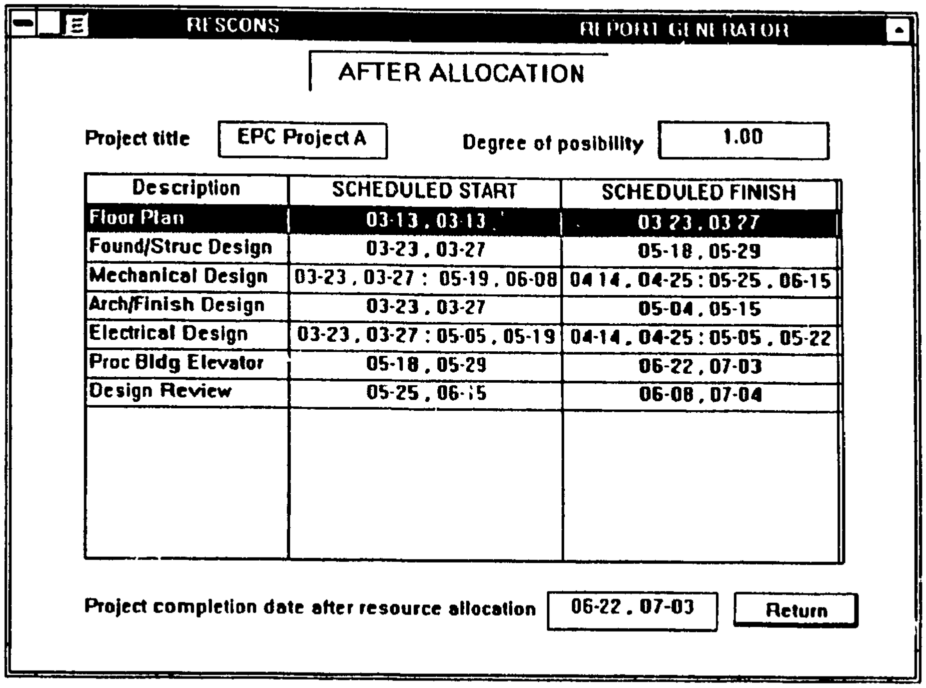
AFTER ALLOCATION

Project title: EPC Project B — Degree of posibility: 1.00

Description	SCHEDULED START	SCHEDULED FINISH
Floor Plan	02-13 , 02-13	03-03 , 03-10
Found/Struc Design	03-03 , 03-10	04-14 , 04-25
Electrical Design	04-18 , 04-24	05-09 , 05-18
Mechanical Design	04-20 , 05-01	05-18 , 06-05
Spec Crane Design	04-20 , 05-01	05-02 , 05-11
Design Review	05-18 , 06-05	05-25 , 06-14
Proc Overhead Crane	05-02 , 05-11	06-27 , 07-13

Project completion date after resource allocation: 06-27 , 07-13 — Return

Fig. 6.23b: Schedule After Resource Allocation of EPC Project (No Interruptions).



RESCONS — REPORT GENERATOR

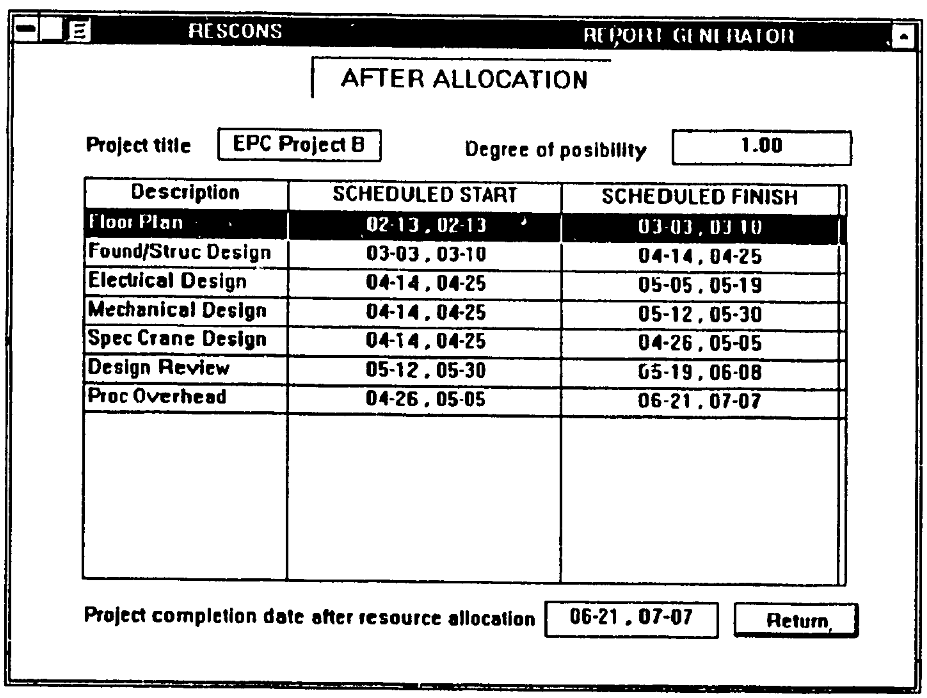
AFTER ALLOCATION

Project title: EPC Project A — Degree of posibility: 1.00

Description	SCHEDULED START	SCHEDULED FINISH
Floor Plan	03-13 , 03-13	03-23 , 03-27
Found/Struc Design	03-23 , 03-27	05-18 , 05-29
Mechanical Design	03-23 , 03-27 : 05-19 , 06-08	04-14 , 04-25 : 05-25 , 06-15
Arch/Finish Design	03-23 , 03-27	05-04 , 05-15
Electrical Design	03-23 , 03-27 : 05-05 , 05-19	04-14 , 04-25 : 05-05 , 05-22
Proc Bldg Elevator	05-18 , 05-29	06-22 , 07-03
Design Review	05-25 , 06-15	06-08 , 07-04

Project completion date after resource allocation: 06-22 , 07-03 — Return

Fig. 6.24a: Schedule After Resource Allocation of EPC Project (Interruptions).



RESCONS — REPORT GENERATOR

AFTER ALLOCATION

Project title: EPC Project B — Degree of posibility: 1.00

Description	SCHEDULED START	SCHEDULED FINISH
Floor Plan	02-13 , 02-13	03-03 , 03-10
Found/Struc Design	03-03 , 03-10	04-14 , 04-25
Electrical Design	04-14 , 04-25	05-05 , 05-19
Mechanical Design	04-14 , 04-25	05-12 , 05-30
Spec Crane Design	04-14 , 04-25	04-26 , 05-05
Design Review	05-12 , 05-30	05-19 , 06-08
Proc Overhead	04-26 , 05-05	06-21 , 07-07

Project completion date after resource allocation: 06-21 , 07-07 — Return

Fig. 6.24b: Schedule After Resource Allocation of EPC Project (Interruptions).

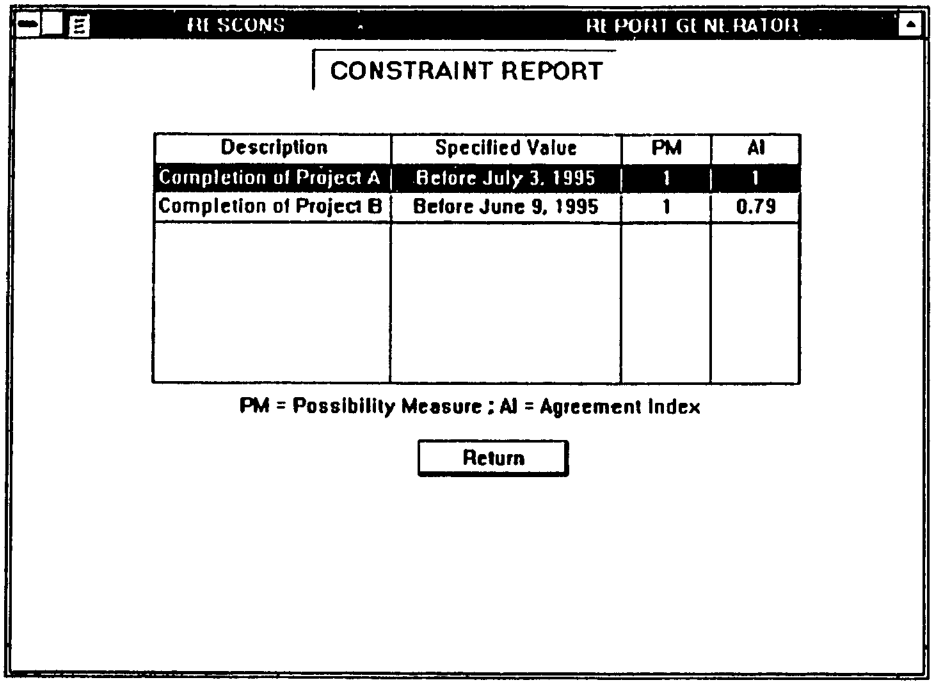


Fig. 6.25a: Constraints Report (No Interruptions).

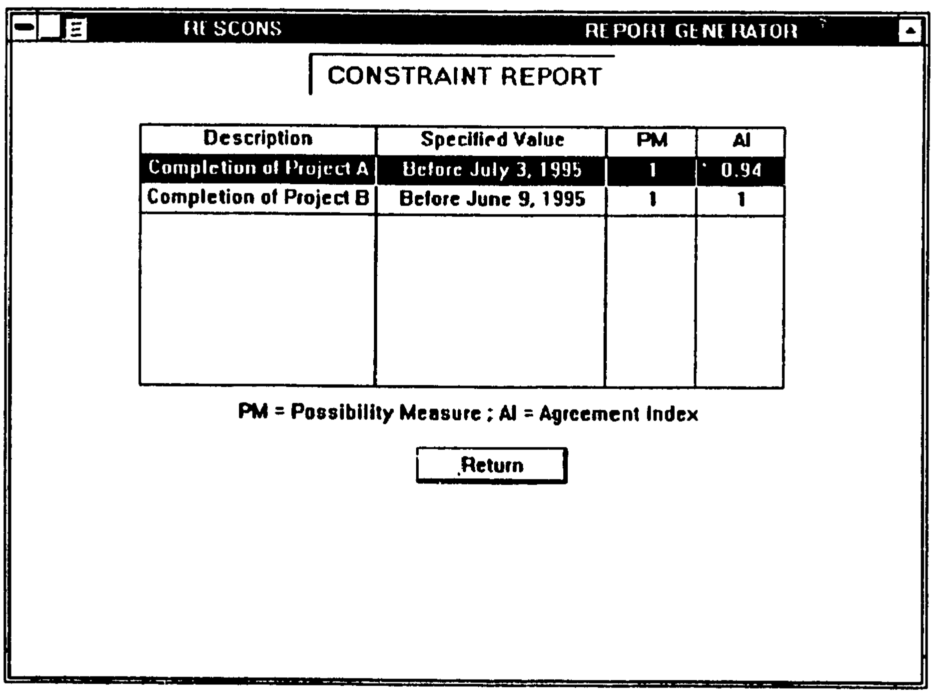


Fig. 6.25: Constraints Report (Interruptions).

contractual date of project B will not be satisfied. The agreement index for the constraint "completion of project B" is calculated to be 0.79 (see Fig. 6.25a). On the other hand, if activity splitting is allowed, there is a small possibility that project A will not be finished before July 3, 1995. The agreement index for the constraint "completion of project A" is calculated to be 0.94 (see Fig. 6.25b). This agreement index is greater than that of the nonsplitting scenario.

Based on the results of this analysis, the management of this A/E firm might be in favor of accepting the new design project (project B). If so, the schedule that allows activity splitting should be adopted since the possibility of completing both projects on their respective contractual dates is higher than the schedule that does allow activity splitting.

CHAPTER 7

CONCLUSIONS

7.1 Conclusions

The problem of project scheduling under resource constraints has been studied and the limitations associated with traditional solution techniques have been identified. The nature and the complexity of this problem have rendered themselves to be best analyzed using heuristic-based solutions. Conventional heuristic-based techniques, however, have been proven inadequate to provide practical schedules. The major limitations associated with those techniques can be attributed to: 1) their inconsistent performance, 2) their inability to account for project uncertainties, 3) their rigid heuristics, and 4) their overlooking of project specific knowledge.

To overcome the above shortcomings, a new heuristic-based method for resource allocation has been developed. The method aims at providing practical solutions to the nondeterministic resource-constrained scheduling problem. The method incorporates an efficient resource allocation model and a suitable method for modeling the uncertainty associated with temporal variables in the project network. The proposed method has been computerized for use as a decision support prototype system capable of assisting schedulers in allocating resources to project activities.

Resource conflicts are resolved using a decomposition technique that forms and evaluates various sets of activities (i.e. alternatives) experiencing conflicts. An evaluation function is used for assessing the negative impact incurred as a result of scheduling each of the generated alternatives. Resources are then allocated to the activities in the set that has a minimum negative impact on the overall project duration. The performance of the model is tested against those of the widely used heuristic rules, considering three objectives: project durations, resource utilization, and resource interruptions. The results indicate that the proposed model statistically outperforms the traditional heuristic techniques in all of the three objectives measured. In addition, the proposed model tends to generate schedules that satisfy, simultaneously, the three objectives more than any of the traditional heuristic rules.

This study has also demonstrated the feasibility of using fuzzy set theory as a tool to model the uncertainty associated with temporal variables such as the duration of project activities, and resource availability periods in the project network. For those situations when the manager cannot exactly specify activity durations either as deterministic numbers or probabilistic random variables, the activity durations can naturally and realistically be expressed as fuzzy numbers. Fuzzy set theory itself is designed to model human perceptions about real-life problems. The theory, by definition, does not require historical data which, in practice, is difficult to compile and maintain. In the present study, imprecise temporal variables are represented using the linear approximation such as the trapezoidal fuzzy numbers expressed in the quadruples form. This format is found to be simple, effective, and practical for representing various degrees of temporal imprecisions normally encountered in construction scheduling. It also facilitates direct fuzzy network calculations.

Fuzzy bounds are used to overcome a major limitation of the fuzzy subtraction which, in the past, hindered successful applications of fuzzy set theory to complete network scheduling. The possibility measure and the agreement index concepts have been proven effective to assist the scheduler in interpreting the calculated fuzzy results in a meaningful way. The fuzzy results generated from this method are tested against probabilistic scheduling methods including PERT and Monte Carlo simulation. The results produced by FNET are found to be reasonably close to those obtained using Monte Carlo simulation. The calculations performed using the proposed FNET are, however, simpler and more transparent. The fact that FNET provides a direct solution, makes it more computationally efficient than Monte Carlo simulation.

A decision support system **"RESCONS"** has been developed in this thesis incorporating the methods described earlier. **RESCONS** acts as an assistant to the scheduler for allocating resources to project activities. The prototype is highly interactive, combining the advantages of full automation and human interactions through the scheduling process. This flexibility may prove advantageous in certain applications, yielding solutions better than those rendered by either one alone. Features of the RESCONS also serve to illustrate the benefits of man/machine interactions. Two numerical examples are presented to demonstrate the effectiveness and practicality of the system features. It has been shown that the system can be used as a decision support tool for improving and facilitating the generation of schedules under nondeterministic resource constraints in single and multiple project environments. The fact that the system allows much flexibility in decision-making pertaining to resource allocation, and provides users with a wealth of information and other supports for their own decision-making, may prove useful in generating realistic schedules. Schedules generated using **RESCONS** can be used in the timely procurement of labor, materials and equipment in addition to project time control.

7.2 Contributions

The contributions of this study are:

1) The development of the least impact model (LIM) for resource allocation.

The development of LIM has significantly improved the quality of the schedules generated using heuristic solutions.

2) The development of a fuzzy network scheduling method (FNET).

FNET overcomes the major limitation experienced by many researchers.

3) The development of flexible fuzzy resource-constrained scheduling method.

The method integrates the developments made in 1 and 2 above. It provides a practical means for schedulers to allocate resources in the situation where the duration of project activities and the resource availabilities are not precisely known.

4) The development of a prototype computer system **RESCONS.**

The system has a number of interesting features. It incorporates the theoretical developments made in this thesis and provides a unique man-machine interactions throughout the resource allocation process.

7.3 Recommendations for Future Work

This study has successfully demonstrated the feasibility of applying fuzzy set theory to construction network scheduling. However, there are a number of potential improvements that could be made to the present study. Some of the most important concerns are:

1) The process of resource allocation proposed in this study can be improved by providing more flexibility with respect to resource sharing, different modes of operations, and permitting activities to be executed using partial resources, and

thereby extending the activity's execution time.

2) The FNET method can be extended to include the cost of project activities. In order to do that the relationships between fuzzy activity durations and their associated fuzzy costs would have to be established. Fuzzy time-cost tradeoff is another promising research avenue worthy of investigations.

3) The process of estimating activity durations supported in **RESCONS** can be enhanced through an integration with a local weather database. This database should provide historical data to facilitate the estimation of the number of working days loss due to precipitation, wind speed, humidity, etc for the given time period.

REFERENCES

AbouRizk, S.M. and Halpin, D.W. "Probabilistic Simulation Studies for Repetitive Construction Processes," **Journal of Construction Engineering and Management,** ASCE, Vol. 116, No. 4, 1990, pp. 575-593.

AbouRizk, S.M., Halpin, D.W., and Wilson, J.R. "Visual Interactive Fitting of Beta Distributions," **Journal of Construction Engineering and Management,** ASCE, Vol. 117, No. 4, 1991, pp. 589-605.

AbouRizk, S.M. and Halpin, D.W. "Statistical Properties of Construction Data," **Journal of Construction Engineering and Management,** ASCE, Vol. 118, No. 3, 1992, pp. 525-544.

AbouRizk, S.M., Halpin, D.W., and Wilson, J.R. "Fitting Beta Distributions Based on Sample Data," **Journal of Construction Engineering and Management,** ASCE, Vol. 120, No. 2, 1994, pp. 288-305.

AbouRizk, S.M. and Sawhney, A. "Subjective and Interactive Estimation," **Canadian Journal of Civil Engineering,** Vol. 20, No. 3, 1993, pp. 457-470.

Adrian, J.J. "Quantitative Methods in Construction Management, New York, NY, Elsevier, 1973.

Ahuja, H.N. and Arunachalam, V. "Risk Evaluation in Resource Allocation," **Journal of Construction Engineering and Management,** ASCE, vol. 110, No. 4, 1984, pp. 324-336.

Ahuja, H.N. and Nandakumar, V. "Simulation Model to Forecast Project Completion Time," **Journal of Construction Engineering and Management,** ASCE, vol. 111, No. 4, 1986, pp. 325-341.

Alagar, V.S. and Peryasami, K. "A Methodology for Deriving and Object-Oriented Design from Functional Specifications, "**Software Engineering Journal,** 1992, July, pp. 247-265.

Alkayyali, O., Minkarah, I., and Sukarieh, G. "Resource Allocation: Toward a Practical Solution," **Civil Engineering Systems,** Vol. 11, 1994, pp. 19-41.

Allam, S.I.G. "Multi-Project Scheduling: A New Categorization for Heuristic Scheduling Rules in Construction Scheduling Problems," Construction Management and Economics, Vol. 6, 1988, pp. 93-115.

Al Sarraj, Z.M. "Formal Development of Line-Of-Balance," **Journal of Construction Engineering and Management,** ASCE, Vol. 116, No. 4, 1990, pp. 689-704.

Ang, A.H.S., Abdelnour, J. and Chaker, A.A. "Analysis of activity Networks Under Uncertainty," **Journal of The Engineering Mechanics Division,** ASCE, August, 1975, pp. 373-387.

Arditi,D. and Albulak, Z.M. "Line-of-Balance Scheduling in Pavement Construction,"**Journal of Construction Engineering and Management,** ASCE, Vol. 112, No. 3, 1985, pp. 411-424.

Ayyub, B.M. and Haldar, A. "Project Scheduling Using Fuzzy Set Concepts," **Journal of Construction Engineering and Management,** ASCE, Vol. 110, No. 2, 1984, pp. 189-204.

Badiru, A.B. "Quantitative Models for Project Planning, Scheduling and Control," West Port, CT, Quorum Books, 1993.

Baker, K.R. "Introduction to Sequencing and Scheduling," New York, N.Y., Wiley, 1974.

Barrie, D.S. and Paulson, B.C. "Professional Construction Management," New York, NY, McGraw-Hill, 1984.

Baxendale, T. "Construction Resource Models by Monte Carlo Simulation," **Construction Management and Economics**, vol. 2, 1984, pp. 201-217.

Bellman, R.E. and Zadeh, L.A. "Decision-making in a Fuzzy Environment," **Management Science**, Vol. 17, No. 4, 1970, pp. 141-164.

Brand, J.D., Meye, N.L., and Shaffer, L.R. "The Resource Scheduling Problem in Construction," Construction Research Series No. 5, Department of Civil Engineering, Urbana, I.L., 1964.

Busch, D.H. "The New Critical Path method," Chicago, I.L., Probus Publishing Company, 1991.

Chanas, S., and Kamburowski, J. "The use of fuzzy variables in PERT." **Fuzzy Sets and Systems**, Vol. 5, 1981, pp. 11-19.

Chang, T.C. and Ibbs, C.W. "Priority Ranking: A Fuzzy Expert System for Priority Decision Making in Building Construction Resource Scheduling," **Building and Environment**, Vol. 25, No. 3, 1990, pp. 253-267.

Chang, T.C., Ibbs, C.W. and Crandall, K.C. "Network Resource Allocation With Support of a Fuzzy Expert System," **Journal of Construction Engineering and Management,** ASCE, Vol. 116, No. 2, June, 1990, pp. 239-260.

Chen, S.J., and Hwang, C.L. "Fuzzy multiple attribute decision making: methods and applications." **Lecture Notes in Economics and Mathematical Systems.** Springer-Verlag., Berlin, Germany, 1992.

Christofides, N., Alvarez-Valdes, R. and Tamarit, J.M. "Project Scheduling with Resource Constraints: A Branch and Bound Approach," **European Journal of Operation Research,** 29, 1987, pp.262-273.

Davis, E.W. and Heidorn, G.E. "Optimal Project Scheduling Under Multiple Resource Constraints," **Management Science**, Vol. 17, No. 12, 1971, pp. B803-B816.

Davis, E.W. "Project Scheduling Under Resource Constraints: Historical Review and Categorization of Procedures," **AIIE Transactions**, Vol. 7, 1973, pp. 297-313.

Davis, E.W. and Patterson, J.H. "A Comparison of Heuristic and Optimum Solutions in Resource Constrained Project Scheduling," **Management Sciences**, Vol. 21, 1975, pp. 944-955.

De La Garza, J.M. and Ibbs, C.W. "Issues in Construction Scheduling Knowledge Representation," In Proceedings of the 5th International Symposium on Building Economics and Construction Management, London, England, 1987.

Diaz, C.F. and Hadipriono, F.C. "Nondeterministic network methods.", **Journal of Construction Engineering and Management**, ASCE, 119(1), 1993, pp.40-57.

Dumond, J. "In a Multi-resource environmant, how much is enough," **International Journal of Production Research**, Vol. 30, No. 2, 1992, pp. 395-410.

Dubois, D. and Prade, H. "**Possibility theory: An approach to computerized processing of uncertainty**," Plenum Press, New York, N.Y., 1988.

Easa, S.M. "Resource Leveling in Construction by Optimization, **Journal of Construction Engineering and Management**, ASCE, vol. 115, No. 2, 1989, pp. 320-336.

Eckel, B. "Using C++," Berkeley, CA, McGraw-Hill, 1989.

Froese, M.T. and Ohbayashi, B.C. Jr. "Object-oriented Programming for Project Management Software," Proceedings of the 7th ISARC, June, Bristol, England, 1990.

Gong, D. "Risk Analysis Oriented Network Scheduling Backward Pass Calculation & Time Disturbance Analysis," Ph.D. Thesis, Department of Civil Engineering, Norwegian Institute of Technology, Trondheim, Norway, 1993.

Gong, D. and Hugsted, R. (1993). "Time-Uncertainty Analysis in Project Networks with a New Merge-Event Time-Estimation Technique," **International Journal of Project management**, Vol. 11, NO. 3, 1993, pp. 165-173.

Hadavi, K., Hsu, W., Chen, T. and Lee, C. "An Architecture for Real-Time Distributed Scheduling," **AI Magazine**, Fall, 1992, pp. 46-56.

Harris, R. "Precedence and Arrow Networking Techniques for Construction," John Wiley & Sons, Inc., 1978.

Hadipriono, C. and Sun, K. "Angular Fuzzy Set Models for Linguistic Values," **Civil Engineering Systems**, Vol. 7, No. 3, 1990, pp. 148-156.

Hakim, M. and Garrett, Jr. J.H. "Object-oriented Techniques for Representing Engineering Knowledge and Data: Pros and Cons," Proceedings of the AIENG, July, Waterloo, Canada, 1992.

Harris, R. "Precedence and Arrow Networking Techniques for Construction," John Wiley & Sons, Inc., 1978.

Harris, R.B. "Packing Method for Resource Leveling (PACK)," **Journal of Construction Engineering and Management**, ASCE, vol. 116, No. 2, 1990, pp. 331-350.

Hastings, N.A.J. "On Resource Allocation in Project Networks," **Operational Research Quarterly**, Vol. 23, No. 2, 1972, pp. 217-221.

Hendrickson, C. and Au, T. "Project Management for Construction," Prentice Hall, Englewood Cliffs, N.J., 1989.

Johnson, R.V. "Resource Constrained Scheduling Capabilities of Commercial Project Management Software," **Project Management Journal**, Vol. 22, No. 4, 1992, pp. 39-43.

Kaufmann, A., and Gupta, M.M. "Introduction to fuzzy arithmetic theory and applications," Van Nostrand Reinhold, New York, N.Y., 1985.

Kaufmann, A., and Gupta, M.M. "Fuzzy mathematical models in engineering and management Science," North-Holland, Amsterdam, 1988.

Keller, G., Warrack, B., and Bartel, H. "Statistics For Management and Economics: A Systematic Approach," Belmon, C.A., Wadsworth, 1988.

Khashenas, S. and Haber, D. "Economic Optimization of Construction Project Scheduling," **Construction Management and Economics**, Vol. 8, 1990, pp. 135-146.

Kim, S.O. and Schniederjans, M.J. "Heuristic Framework for the Resource Constrained Multi-project Scheduling Problem," **Computer Operations Research**, Vol. 16, No. 6, 1989, pp. 541-556.

Kurtulus, I. and Davis, E.W. "Multiproject Scheduling Categorization of Heuristic Rule Performance," **Management Science**, Vol. 28, 1982, pp. 161-172.

Lee, S.M., Park, O.E., and Economides, S.C. "Resource Planning for Multiple Projects," **Decision Sciences**, Vol. 9, 1978, pp. 49-67.

Lawrence, S.R. "Resource Constrained Project Scheduling: A Computation Comparison of Heuristic Scheduling Techniques," Technical Report, GSIA, Carnegie-Mellon University, 1985.

Logcher, R.D. "Adding Knowledge Based Systems Technology to Project Control Systems," Project Controls: Needs and Solutions, Ibbs, C., Ashley, D. Editors, NewYork, NY, 1987.

Lorterapong, P. "A Fuzzy Heuristic Method for Resource-Constrained Project Scheduling," **Project Management Journal**, Vol 25. , No 4. 1994, pp. 12-18.

Lorterapong, P. and Moselhi, O. "Project-Network Scheduling Using Fuzzy Set Theory," Manuscript submitted for possible publication in **Journal of Construction Engineering and Management**,1995.

Lumsden, P. "The Line of Balance Method," Pergamon Press Limited, Industrial Training Division, 1968.

McCahon, C.S. "Fuzzy Set Theory Applied to Production and Inventory Control," Ph.D. Dissertation, Kansas State University, Manhattan, Kansas, 1987.

McCahon, C.S. "Using PERT as an Approximation of Fuzzy Project-Network Analysis," **IEEE Transactions on Engineering Management**, Vol. 40, No. 2, 1993, pp. 146-153.

Milton, J.S. and Arnold, J.C. "Probability and Statistics in the Engineering and Computer Sciences, New York, N.Y., McGraw-Hill, 1986.

Moder, J.J. and Rogers, E.G. "Judgement Estimates of the Moments of PERT Type Distributions," **Management Science**, Vol. 15, No.2, 1968.

Mohanty, R.P. and Siddiq, M.K. "Multiple Projects-Multiple Resource-Constrained Scheduling: Some Studies," **International Journal Production Research**, Vol. 27, No. 2, 1989, pp. 261-280.

Moselhi O. and Lorterapong, P. "Least Impact Algorithm for Resource Allocation," **Canadian Journal of Civil Engineering**, Vol. 20. No. 2, 1993a, pp. 180-188.

Moselhi O. and Lorterapong, P. "Near Optimal Solution for Resource-Constrained Scheduling Problems," **Construction Management and Economics**, Vol. 11, 1993b, pp. 293-303.

Moselhi, O. and Lorterapong, P. "Scheduling of Multi-Storey Buildings in a Constrained Environment," Proceedings of the 5th International Conference on Civil and Building Engineering, ASCE, Anaheim, CA, 1993c.

Moselhi O. and Lorterapong, P. "A Decision Support System for Resource-Constrained Scheduling," Proceedings of the Canadian Society for Civil Engineering Annual Conference, CSCE, Fredericton, NB, June, 1993d.

Moselhi O. and Lorterapong, P. "Fuzzy vs Probabilistic Scheduling," Manuscript accepted for publications in the 12th ISARC Conference Proceedings, Warsaw, Poland, 1995.

Oberlender, G.D. "Project Management for Engineering and Construction," New York, NY, McGraw-Hill, 1993.

Padilla, E.M. and R.I. Carr, "Resource Strategies for Dynamic Project Management" **Journal of Construction Engineering and Management**, vol. 117, no.2, 1991, pp. 279-293.

Panwalker, S.S. and Iskander, W. "A Survey of Scheduling Rules," **Operation Research**, Vol. 25, No. 1, 1977, pp. 45-61.

Patterson, J.H. "A Comparison of Exact Approaches for Solving the Multiple Constrained Resource Project Scheduling Problem," **Management Sciences**, Vol. 30, No. 7, 1984, pp. 854-867.

Perry, C. and Greig, I.D. "Estimating the Mean and Variance of Subjective Distributions in PERT and Decision Analysis," **Management Science**, Vol.21, No. 12, 1975.

Pritsker, A.A. Watters, L.J. and Wolfe, P.M., "Multi-Project Scheduling with Limited Resources: A Zero-one Programming Approach," **Management Science**, Vol. 16, 1969, pp. 93-108.

Rahbar, F.F. and Rowing, J.E. "Repetitive Activity Scheduling Process," **AACE Transactions, 36th Annual Meeting**, Orlando, FL, 1992, pp. O.5.1 - O.5.8.

Russel, A.D. and Wang, W.C.M. "New Generation of Planning Structures," **Journal of Construction Engineering and Management**, ASCE, Vol. 119, No. 2, 1993, pp. 196-214.

Savin, D., Alkass,S. and Fazio, P. "A Neural Network Model for Construction Resource Leveling," Proceedings of CSCE Annual Conference for Civil Engineering, June, 1994, Winipeg, M.N., pp.393-402.

Shaked, O. and Warszawski, A. "CONSCHED: Expert System for Scheduling of Modular Construction Projects," **Journal of Construction Engineering and Management**, ASCE, Vol. 118, No. 3 ,1992, pp. 488-506.

Shanmuganayagum, V. "Current Float Techniques for Resource Scheduling," **Journal of Construction Engineering and Management**, ASCE, Vol. 115, No. 3 ,1989, pp. 401-411.

Selinger, S. "Construction Planning for Linear Projects," **Journal of Construction Division**, ASCE, Vol. 106, No. CO2, 1980, pp. 195-205.

Stinson, J.P., Davis. E.W., and Khumawala, B.M. "Mutiple Resource-Constrained Scheduling Using Branch and Bound," **AIIE Transactions**, Vol. 10, No. 3, 1978, pp. 252-259.

Suarez, L.F. "Resource Allocation: A Comparative Study," **Project Management Journal**, Vol. 18, No. 1, 1987, pp. 68-71.

Talbot, F.B. and Patterson, J.H. "An Efficient Integer Programming Algorithm With Network Cuts and for Solving Resource-Constrained Scheduling Problem," **Management Science**, Vol. 24, No. 11, 1978, pp. 1163-1174.

Talbot, F.B. and Patterson, J.H. "Optimal Methods for Scheduling Projects Under Resource Constraints," **Project Management Quarterly**, Vol. 10 No. 4, 1979, pp. 26-33.

Tavakoli, A. and Riachi, R. "CPM Use in ENR Top 400 Contractors," **Journal of Management in Engineering,** ASCE, vol. 6, No. 3, 1994, pp. 282-295.

Teixeira, J.C. and Bezelga, A. "Resource Allocation in Strategic Planning," Proceedings of the CIB Conference, March, Sydney, Australia, (1990).

Thamhain, H.J. and Wilemon, D.L. "Criteria for Controlling Projects According to Plan," **Project Management Journal**, Vol. 17, No. 2, 1986, pp. 75-81

Touran, A. "Discussion of Current Float Techniques for Resource Scheduling," **Journal of Construction Engineering and Management,** ASCE, Vol. 117, No. 3, 1990, pp. 574-575.

Trauner, T.J. "managing the Construction Project: A Practical Guide for the Project Manager," New York, NY, John Wiley & Sons, 1993.

Wu, R.W. and Hadipriono, F.C. "Fuzzy Modus Ponens Deduction Technique for Construction Scheduling," **Journal of Construction Engineering and Management,** ASCE, Vol. 120, No. 1, March, 1994, pp. 162-179.

Weist, J.D. and Levy, F.K. "A Management Guide to PERT/CPM, Englewood Cliff, NJ, 1977.

Zadeh, L.A. "Fuzzy sets.", **Information Control,** vol. 8, 1965, pp. 338-353.

Zadeh, L.A. "Fuzzy Sets as a Basis for a Theory of Possibility," **Fuzzy Sets and Systems,** Vol. 1, No. 1, 1978, pp. 3-28.

Zimmermann, H.J. "Fuzzy set theory and its applications," Kluwer Academic, Boston, M.A., 1991.

APPENDIX I: Wilcoxon Signed Rank Sum Test for Project Durations.

Network	Project Duration			A - B	|A - B|	Ranks		A - C	|A - C|	Ranks	
	LIM (A)	MINSLK (B)	MINLFT (C)			T^-	T^+			T^-	T^+
1	47	49	48	-2	2	14.0		-1	1	7.5	
2	47	47	51	0	0			-4	4	32.5	
3	47	47	51	0	0			-4	4	32.5	
4	40	43	47	-3	3	21.5		-7	7	38	
5	45	43	44	2	2		14.0	1	1		7.5
6	39	40	40	-1	1	5.0		-1	1	7.5	
7	39	40	40	-1	1	5.0		-1	1	7.5	
8	36	36	36	0	0			0	0		
9	37	40	40	-3	3	21.5		-3	3	26	
10	93	94	94	-1	1	5.0		-1	1	7.5	
11	84	85	85	-1	1	5.0		-1	1	7.5	
12	82	82	82	0	0			0	0		
13	82	82	82	0	0			0	0		
14	78	81	81	-3	3	21.5		-3	3	26	
15	80	81	81	-1	1	5.0		-1	1	7.5	
16	78	81	81	-3	3	21.5		-3	3	26	
17	76	74	74	2	2		14.0	2	2		19
18	68	73	70	-5	5	33.0		-2	2	19	
19	68	68	68	0	0			0	0		
20	45	49	49	-4	4	28.0		-4	4	32.5	
21	36	40	40	-4	4	28.0		-4	4	32.5	
22	37	39	38	-2	2	14.0		-1	1	7.5	
23	63	67	57	-4	4	28.0		6	6		37
24	50	52	53	-2	2	14.0		-3	3	26	
25	38	42	42	-4	4	28.0		-4	4	32.5	
26	37	41	41	-4	4	28.0		-4	4	32.5	
27	33	34	34	-1	1	5.0		-1	1	7.5	
28	35	39	37	-4	4	28.0		-2	2	19	
29	33	34	34	-1	1	5.0		-1	1	7.5	
30	58	57	62	1	1		5.0	-4	4	32.5	

Wilcoxon Signed Rank Sum Test for Project Durations (Continued).

Network	Project Duration LIM (A)	MINSLK (B)	MINLFT (C)	A - B	|A - B|	Ranks T^-	T^+	A - C	|A - C|	Ranks T^-	T^+
31	31	31	31	0	0			0	0		
32	34	31	34	3	3		21.5	0	0		
33	26	26	26	0	0			0	0		
34	22	22	22	0	0			0	0		
35	30	35	31	-5	5	33		-1	1	7.5	
36	27	32	29	-5	5	33		-2	2	19	
37	28	28	28	0	0			0	0		
38	23	24	24	-1	1	5		-1	1	7.5	
39	39	41	41	-2	2	14		-2	2	19	
40	37	41	41	-4	4	28		-4	4	32.5	
41	35	37	37	-2	2	14		-2	2	19	
42	35	37	37	-2	2	14		-2	2	19	
43	35	37	37	-2	2	14		-2	2	19	
44	37	37	37	0	0			0	0		
45	55	55	54	0	0			1	1		7.5
46	50	50	47	0	0			3	3		26
47	45	48	45	-3	3	21.5		0	0		
48	43	43	42	0	0			1	1		7.5
49	46	46	44	0	0			2	2		19
50	34	34	34	0	0			0	0		
51	32	32	32	0	0			0	0		
						$\Sigma = 540.5$	$\Sigma = 54.5$			$\Sigma = 617.5$	$\Sigma = 123.5$

T_{A-B} (for $\alpha = 0.005$, and $n = 34$) $= 149^* > 54.5$ ---> Reject the null hypothesis
T_{A-C} (for $\alpha = 0.005$, and $n = 38$) $= 195^* > 123.5$ ---> Reject the null hypothesis
*Taken from Keller et al. 1988.

APPENDIX II: Wilcoxon Signed Rank Sum Test for Resource Utilization.

Network	Resource Utilization (%)			A - B	|A - B|	Ranks		A - C	|A - C|	Ranks	
	LIM (A)	MINSLK (B)	MINLFT (C)			T⁻	T⁺			T⁻	T⁺
1	79.25	79.23	68.46	0.2	0.2		2	10.79	10.79		43
2	85.57	85.57	81.54	0	0			4.03	4.03		28.5
3	85.28	85.28	81.25	0	0			4.03	4.03		28.5
4	73.28	76.05	68.21	-2.77	2.77	20		5.07	5.07		34
5	74.84	67.99	65.51	6.85	6.85		35	9.33	9.33		42
6	77.28	74.17	77.55	3.11	3.11		21	-0.27	0.27	3.5	
7	69.23	66.11	66.11	3.12	3.12		22	3.12	3.12		26
8	54.48	54.48	54.48	0	0			0	0		
9	78.01	68.61	70.14	9.4	9.4		38	7.87	7.87		41
10	41.69	40.88	43.18	0.81	0.81		10	-1.49	1.49	18	
11	40.26	40.61	40.61	-0.35	0.35	6		-0.35	0.35	6	
12	36.63	36.63	36.63	0	0			0	0		
13	31.52	31.52	31.52	0	0			0	0		
14	29.20	28.43	28.43	0.77	0.77		9	0.77	0.77		13
15	33.50	33.54	33.54	-0.04	0.04	3		-0.04	0.04	2	
16	31.40	33.54	33.54	-2.14	2.14	15		-2.14	2.14	22	
17	58.30	60.54	60.54	-2.24	2.24	16		-2.24	2.24	23	
18	59.98	56 69	59.14	3.29	3.29		25	0.84	0.84		14
19	56.20	56.20	56.20	0	0			0	0		
20	62.82	58.09	58.09	4.73	4.73		30	4.73	4.73		32
21	67.79	61.58	61.52	6.21	6.21		34	6.27	6.27		39
22	62 39	59.16	62.75	3.23	3.23		23	-0.36	0.36	7	
23	63.80	59 98	69 98	3.82	3.82		27	-5.88	5.88	36	
24	65 37	60.85	59.72	4.52	4.52		29	5.65	5 65		35
25	69.98	64.94	64.94	4.94	4.94		31	4.94	4.94		33
26	67.98	61.87	61.87	6.11	6.11		33	6.11	6.11		38
27	64.69	65.20	65.20	-0.51	0.51	7		-0.51	0.51	10	
28	65.31	62.72	64.90	2.59	2.59		17	0.41	0.41		8
29	62.70	61.69	61.69	1.01	1.01		11	1.01	1.01		16
30	62.69	62.36	58.27	0 33	0.33		5	4.42	4.42		30

Wilcoxon Signed Rank Sum Test for Resource Utilization (Continued).

Network	Resource Utilization: PRO (A)	Resource Utilization: MINSLK (B)	Resource Utilization: MINLFT (C)	A - B	|A - B|	Ranks: T^-	Ranks: T^+	A - C	|A - C|	Ranks: T^-	Ranks: T^+
31	77.95	77.95	78.55	0	0			-0.6	0.6	11	
32	70.09	73.58	70.58	-3.49	-3.49	26		-0.49	0.49	9	
33	77.50	80.77	75.54	-3.27	-3.27	24		1.96	1.96		21
34	70 30	70.30	70.30	0	0			0	0		
35	61.31	50.10	57.75	11.21	11.21		40	3.56	3.56		27
36	57.15	47.22	52.71	9.93	9.93		39	4.44	4.44		31
37	53.59	53.59	53.59	0	0			0	0		
38	51.59	49.88	49.88	1.71	1.71		13	1.71	1.71		19
39	52.52	58.54	58 54	-6.02	-6.02	32		-6.02	6.02	37	
40	51 40	58.54	58.54	-7.14	-7.14	36		-7.14	7.14	40	
41	51.39	51.38	51.38	0.01	0.01		1	0.01	0.01		1
42	46.59	44.76	44.76	1.83	1.83		14	1.83	1.83		20
43	41.85	41.58	41.58	0.27	0.27		4	0.27	0.27		3.5
44	54.52	55.56	55.56	-1.04	-1.04	12		-1.04	1.04	17	
45	54.89	52.22	52.22	2.67	2.67		18	2.67	2.67		24
46	55.82	53.10	56.77	2.72	2.72		19	-0.95	0.95	15	
47	56.96	52.77	56.65	4.19	4.19		28	0.31	0.31		5
48	45.36	44.62	44.62	0.74	0.74		8	0.74	0.74		12
49	78.33	85.58	81.06	-7.25	-7.25	37		-2.73	2.73	25	
50	84.95	84.95	84.95	0	0			0	0		
51	57.58	57.58	57.58	0	0			0	0		
						$\Sigma = 234$	$\Sigma = 586$			$\Sigma = 281.5$	$\Sigma = 664.5$

T_{A-B} (for $\alpha = 0.01$, and $n = 40$) = $238^* > 234$ ---> Reject the null hypothesis
T_{A-C} (for $\alpha = 0.025$, and $n = 43$) = $311^* > 281.5$ ---> Reject the null hypothesis
*Taken from Keller et al. 1988.

APPENDIX III: Wilcoxon Signed Rank Sum Test for Resource Interruptions.

Network	Resource Interruption			A - B	|A - B|	Ranks		A - C	|A - C|	Ranks	
	LIM (A)	MINSLK (B)	MINLFT (C)			T^-	T^+			T	T^+
1	0.50	0.50	0.25	0.00	0.00			0.25	0.25		24
2	0.25	0.25	0.25	0.00	0.00			0	0		
3	0.00	0.00	0.00	0.00	0.00			0	0		
4	0.00	0.20	0.20	-0.20	0.20	23.5		-0.2	0.2	20	
5	0.00	0.17	0.17	-0.17	0.17	19		-0.17	0.17	16	
6	0.17	0.17	0.17	0.00	0.00			0.00	0.00		
7	0.14	0.14	0.14	0.00	0.00			0.00	0.00		
8	0.00	0.00	0.00	0.00	0.00			0.00	0.00		
9	0.00	0.14	0.14	-0.14	0.14	15.5		-0.14	0.14	12.5	
10	3.00	3.14	3.14	-0.14	0.14	15.5		-0.14	0.14	12.5	
11	2.56	2.44	2.44	0.12	0.12		11.5	0.12	0.12		8.5
12	2.36	2.55	2.55	-0.19	0.19	21.5		-0.19	0.19	18.5	
13	1.92	2.08	2.08	-0.16	0.16	18		-0.16	0.16	15	
14	1.92	1.92	1.92	0.00	0.00			0.00	0.00		
15	2.00	2.33	2.33	-0.33	0.33	30		-0.33	0.33	30.5	
16	1.71	2.33	2.33	-0.62	0.62	35		-0.62	0.62	38	
17	2.71	2.76	2.59	-0.05	0.05	3		0.12	0.12		8.5
18	3.32	2.58	2.37	0.74	0.74		37	0.95	0.95		39
19	2.65	2.65	2.75	0.00	0.00			-0.10	0.10	6.5	
20	2.11	1.89	1.89	0.22	0.22		25.5	0.22	0.22		21.5
21	1.29	1.10	1.48	0.19	0.19		21.5	-0.19	0.19	18.5	
22	1.30	1.06	1.22	0.24	0.24		27	0.08	0.08		3.5
23	2.84	3.26	2.53	-0.42	0.42	32		0.31	0.31		29
24	1.83	1.79	1.75	0.04	0.04		2	0.08	0.08		3.5
25	1.18	1.21	1.33	-0.03	0.03	1		-0.25	0.25	24	
26	1.20	1.33	1.43	-0.13	0.13	13.5		-0.13	0.13	10.5	
27	1.11	1.37	1.37	-0.26	0.26	28		-0.26	0.26	26	
28	1.00	1.22	1.04	-0.22	0.22	25.5		-0.04	0.04	1	
29	0.74	0.89	0.89	-0.15	0.15	17		-0.15	0.15	14	
30	1.33	1.22	1.72	0.11	0.11		10	-0.39	0.39	34	

Wilcoxon Signed Rank Sum Test for Resource Interruptions (Continued).

Network	Resource Interruption			A - B	|A - B|	Ranks		A - C	|A - C|	Ranks	
	LIM (A)	MINSLK (B)	MINLFT (C)			T^-	T^+			T^-	T^+
31	1.19	1.19	0.81	0				0.38	0.38		
32	1.25	1.05	1.30	0.20	0.20			-0.05	0.05	2	
33	0.48	0.48	0.48	0				0			
34	0.28	0.16	0.50	0.12	0.12			-0.22	0.22	21.5	
35	2.30	2.40	2.70	-0.1	0.1	8		-0.4	0.4	35	
36	1.77	1.85	2.15	-0.08	0.08	4		-0.38	0.38	32.5	
37	2.45	2.45	2.45	0				0			
38	1.80	2.07	2.07	-0.27	0.27	29		-0.27	0.27	27	
39	1.89	2.33	2.33	-0.44	0.44	33		-0.44	0.44	36	
40	1.80	2.33	2.33	0.53	0.53	34		-0.53	0.53	37	
41	2.00	2.10	2.10	0.1	0.1	8		-0.10	0.10	6.5	
42	1.91	1.82	1.82	0.09	0.09		5.5	0.09	0.09		5
43	1.07	1.07	1.07	0				0			
44	1.73	1.91	1.91	-0.18	0.18	20		-0.18	0.18	17	
45	2.00	2.67	2.33	-0.67	0.67	36		-0.33	0.33	30.5	
46	2.30	2.20	2.00	0.10	0.10		8	0.3	0.3		28
47	2.36	2.27	2.36	0.09	0.09		5.5	0			
48	1.73	1.60	1.60	0.13	0.13		13.5	0.13	0.13		10.5
49	1.25	0.88	1.00	0.37	0.37		31	0.25	0.25		24
50	0.58	0.58	0.58	0				0			
51	0.29	0.29	0.29	0				0			
						$\Sigma = 470$	$\Sigma = 233$			$\Sigma = 542.5$	$\Sigma = 237.5$

T_{A-B} (for $\alpha = 0.05$, and $n = 37$) $= 242^{*} > 233$ —> Reject the null hypothesis
T_{A-C} (for $\alpha = 0.025$, and $n = 38$) $= 271^{*} > 237.5$ —> Reject the null hypothesis
*Taken from Keller et al. 1988.