NOTE TO USERS

This reproduction is the best copy available.

UMI®

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

A COMPUTERIZED MULTI-DIMENSIONAL SUPPORT SYSTEM (CM-DSS)

FOR

PLANNING, SCHEDULING AND CONTROLLING CONSTRUCTION PROJECTS

A dissertation submitted to the

Division of Graduate Studies and research
at the University of Cincinnati

in partial fulfillment of the
requirements for the degree of

DOCTOR OF PHILOSOPHY

in the Department of Civil and Environmental Engineering

by

Osama Jawad Alkayyali

B.Sc. Civil Engrg., Al-Azhar University, Cairo, Egypt
June 1976

M.Sc. Construction Engineering and Management, University of Cincinnati, Cincinnati, Ohio
June 1990

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

UMI Number: DP16642

INFORMATION TO USERS

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleed-through, substandard margins, and improper alignment can adversely affect reproduction.
In the unlikely event that the author did not send a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

UMI®

UMI Microform DP16642
Copyright 2009 by ProQuest LLC
All rights reserved. This microform edition is protected against unauthorized copying under Title 17, United States Code.

ProQuest LLC
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106-1346

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

ABSTRACT

A deeper understanding is needed of the practices used by the construction industry to plan and control their projects. The work presented in this dissertation contributes an additional step toward this understanding by articulating some of the industry's concerns.

This research suggests a computerized multi-dimensional support system (CM-DSS) for planning, scheduling, and controlling construction projects. The major objective of this system is to further the use of CPM by the construction industry by making it more indispensable for project management practices. In the planning phase, a planner can use CM-DSS to develop four scheduling alternatives for the project under study and to evaluate these alternatives and select a target schedule. CM-DSS, then, provides the planner with a comprehensive network analysis for the project. This analysis can be used to measure the criticality of the target schedule as well as its activities, and to forecast any potential deviation.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

CM-DSS can be used by planners and construction managers to plan an activity under variable productivity rates and to examine different scheduling alternatives. It can be also used to tune up the logic of the target schedule by running What-If analyses. In addition, CM-DSS allows the planner to incorporate control criteria into the project schedule during the planning process. CM-DSS can be used during both the planning and control phases.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

بسم الله الرحمن الرحيم

In The Name Of God The Most Compassionate The Most Merciful

Read in the name of thy Lord who createth. — إقرأ باسم ربك الذي خلق .

Createth man from a clot. — خلق الإنسان من علق .

Read; and thy Lord is the most bounteous. — إقرأ وربك الأكرم .

Who teacheth by the pen. — الذي علم بالقلم .

Teacheth man that which he knows not. — علم الإنسان ما لم يعلم .

(The Holly QUR'AN;Ch.97;Vs.1-5.)

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

ACKNOWLEDGEMENTS

All praise is due to God. Words cannot express my appreciation and thanks to my academic advisor, Professor Issam A. Minkarah, for his constructive advice and fatherly treatment. The idea of this research owes him a debt of honor. Without his belief in my performance, I doubt that this work would have been accomplished. A very special thank-you goes out to the members of my doctoral committee. Their suggestions and recommendations have been most helpful and are greatly acknowledged.

I would like to express my sincere gratitude to the Department of Civil and Environmental Engineering for the warm treatment, family atmosphere and support at all levels from enrollment to graduation.

I would like to express my deep appreciation of the love and support my parents, sisters, sons and all family members have given me. Their patience, trust and encouragement were a spring of inspiration throughout my life. This thesis is dedicated to all of them.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

DEDICATION

To my sons: Tarek and Hussam.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

TABLE OF CONTENTS

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

CHAPTER FOUR

CM-DSS Computer Program

CHAPTER FIVE

Computer Session With CM-DSS

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

LIST OF TABLES

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

LIST OF FIGURES

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

CHAPTER ONE

INTRODUCTION

1.1 PLANNING

In general terms, planning is the process of breaking a project down into specific activities, and defining the sequence in which those activities can or must be performed.

Construction planning is the most time-consuming and critical aspect of project management[1]. Typically, it involves defining the project activities, estimating the required resources and duration for each activity, determining precedence relationships among these activities, choosing appropriate technology, and preparing a graphic presentation for the plan in the form of a Bar Chart or a network. In addition, a sound project plan must take into consideration practical constraints such as resource availability, approval of shop drawings, material delivery to the site, safety requirement, weather conditions, and so on.

To represent these relationships and constraints, it is a common practice to set a time schedule as a primary guideline for implementing the project at hand.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

1.2 SCHEDULING

Once a plan has been developed, the next step is to consider the element of time, i.e., the project schedule. A project schedule, which serves as the principal guideline for project execution, is a projected timetable of construction activities. It is a time-based arrangement of activities planned to take place in order to complete the project within the preset duration[25,29].

Contractors and subcontractors need a time-based schedule to know when and how many crews are needed; when they can start executing their work, etc. Vendors also need a schedule to know when to deliver material. Several network-based scheduling techniques such as the Critical Path Method (CPM), Program Evaluation and Review Techniques (PERT), and Graphic Evaluation and Review Techniques (GERT) are available for construction scheduling.

The network-based scheduling technique is a method that is used for scheduling activities by presenting the interrelationships among them in a parallel or series fashion as designed by a planner. The major use of the network techniques is to provide a mathematical-based network that represents the construction process with respect to time. It also provides the planner with a means to analyze the impact of changing different parts of the network on the project[25].

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

A realistic and workable network-based schedule can be accomplished by[1,25]:

1. Estimating the duration required to execute each activity;
2. Calculating the overall project completion time;
3. Determining the start and finish dates for each activity;
4. Identifying critical and major activities;
5. If necessary, shortening the project duration at the least possible cost if it deems necessary (usually, when the project completion date exceeds the contract preset date); and
6. Recalculating the start and finish dates of some chosen activities, according to the surplus times that some activities have. This new schedule is called the target schedule.

1.3 CONTROL

Control is the last phase of the implementation cycle of planning-scheduling-control. It is the process of establishing objectives, developing measurements, and providing feedback in order to measure progress toward the preset objectives, to detect any deviation from these objectives and to modify measurements as required and to take corrective action when needed. Project control is a process which most contractors implement to study the project during execution. This documentation provides a reliable source of information for planning future projects.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Typical project control procedures include three major functions: monitoring, evaluating and updating. Monitoring allows the project manager to determine the progress of the project and to predict the likelihood of meeting the target schedule. When it becomes clear that the current level of performance will result in overruns in cost or time, the target schedule is adjusted, or updated, to reflect the current status, and to meet contract provisions. The essential component of monitoring is to get feedback on the progress of the project from the field staff[46].

Once the feedback from the field staff has been acquired, the project manager/planner must evaluate the data and decide what should be done to maintain the target completion date. The evaluation process usually starts by producing schedule tables from feedback information. The project manager/planner examines these tables, and compares the current start and finish dates with the ones that have been targeted. It then becomes clear where the difficulties may lie. Bar Charts, S-Curve charts (percentage of time vs. percentage of cost to date), computer printouts, etc., are usually used to compare the current status of the project with the target plan[46].

The final phase of a sound project control is updating. This process involves the correction of the target schedule in order to achieve the preset objectives, i.e., completion within budgetary

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

and time limits. Updating consists of replanning and rescheduling the remaining work. It is an expensive and time-consuming process. The decision to update the project must be weighed against ensuing costs. Effective project control cannot be achieved unless updating is carried out.

1.4 THE CRITICAL PATH METHOD (CPM)

One of the first network-based models is the Critical Path Method (CPM). CPM was developed in 1957 by James E. Kelly of the Remington Rand Corporation (UNIVAC), and Morgan Walker of the Du Pont Company, thus called at the time the Kelley-Walker method. Later, it was known as the Arrow Diagram Method (ADM)- see figure (1.1). The original task was to come up with a scheduling tool for the construction and maintenance of chemical processing plants. In the early 1960s, Professor John Fondahl of Stanford University established a second version of CPM which is called the Precedence Diagram Method (PDM) - see figure (1.2). Similar research was underway at the same time in Europe. In England, S.P.S. Andrew developed the "controlling sequence duration" for scheduling maintenance in 1955; and the Central Electricity Generating Board established the "minimum irreducible sequence" method in 1957. In France, Roy came up with the "method of potentials" in 1958 and finalized it by 1960. Nevertheless, it is widely known that the CPM concept was generated in the United States[2].

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

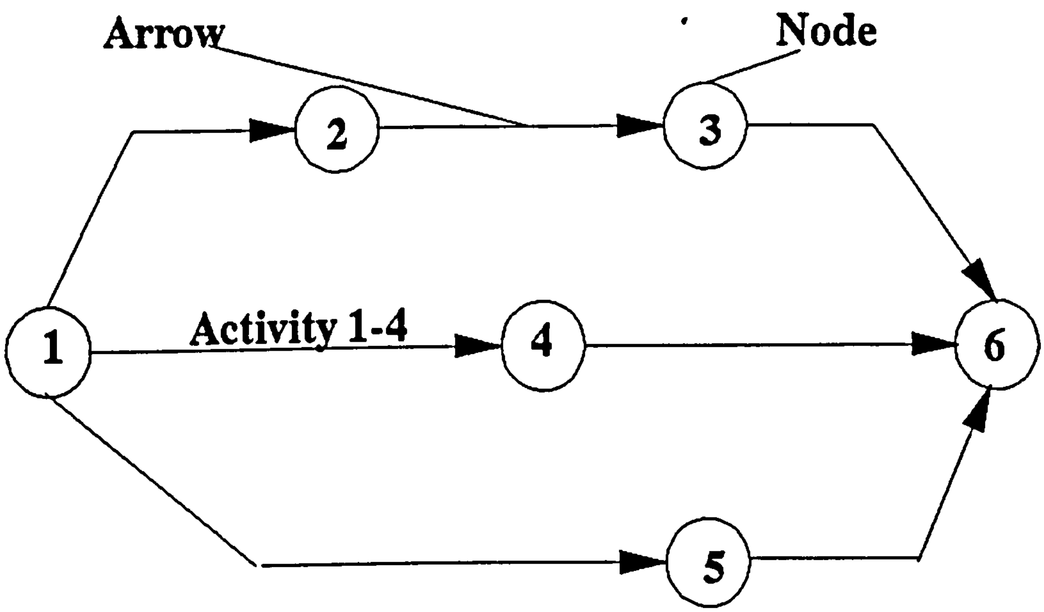
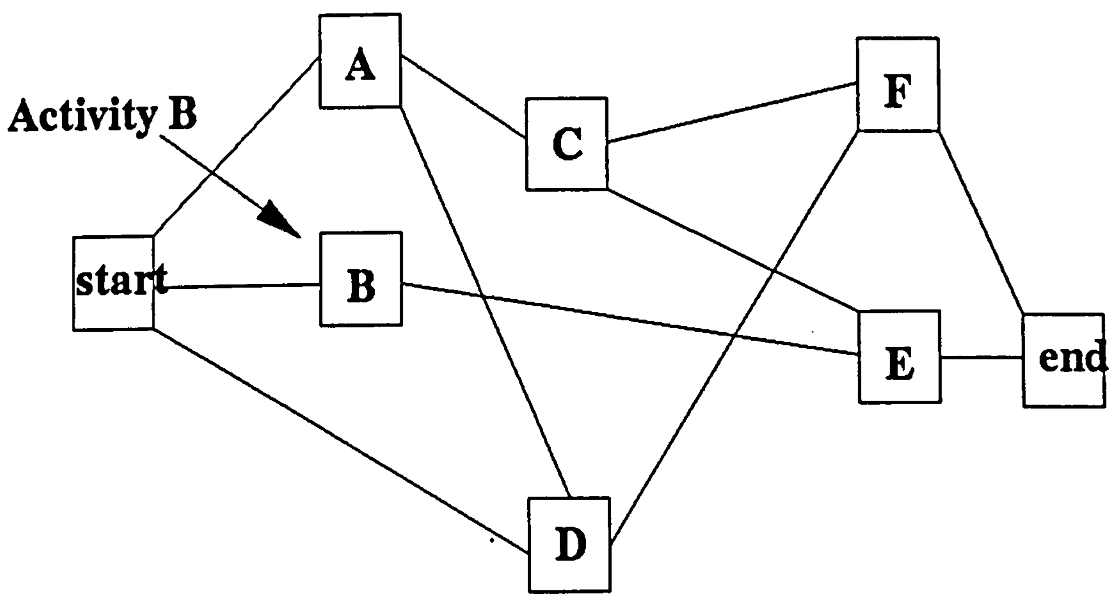


Figure (1.1).--Arrow Diagram Method (ADM)



Figure(1.2).--Precedence Diagram Method (PDM)

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Generally, the Critical Path Method (CPM) is a networking technique in which a network is prepared and sequences of activities are defined in such a way that at least one sequence is characterized by having the longest duration. This sequence of activities, which determines the project completion time, is called the critical path, and each activity of that sequence is called a critical activity[1,2,28]. If the completion of any critical activity is delayed, the completion of the project will also be delayed. During the execution of a project, the CPM schedule is frequently updated. Each new schedule is then compared to the original one in order to control the project and to measure any deviation from the established schedule.

1.4.1 CPM ASSUMPTIONS

When CPM is used to prepare a project schedule, it assumes a deterministic project network structure. It also considers activities to be independent and to have deterministic durations. During the scheduling phase, CPM considers only those constraints resulting from the precedence relationships or technological logic among activities. It ignores any resource constraints[1,41]. Conventionally, it assumes that the availability of resources is unlimited and that labor can be hired and fired at any time.

1.4.2 CPM CALCULATIONS

There are two approaches to represent a project network: arrow diagram (AD) and precedence diagram (PD). AD, sometimes called an

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

activity-on-arrow diagram, represents project activities by using arrows. PD, sometimes called an activity-on-node or precedence diagram, represents project activities by using circles or squares. Throughout this research the focus will be only on the former.

In the following sections, calculations for conventional CPM will be presented and illustrated by using an adopted example[29]. These calculations will help explain some of the terms such as the project completion time, early start, early finish, total float, free float, etc.

1.4.3 CPM NOTATIONS

Act.: An activity is a portion of the project that consumes time or resources and has a beginning and an end.

Dummy: It is an activity with zero time duration used to represent a dependency of one activity upon another, and to provide unique numbering for each activity.

Event (or Node): A connection between two or more activities.

Dur: Duration in workdays, or workhours, etc.

i: Beginning node of an activity

j: Ending node of an activity

ES: Early Start (ES = latest EF of all preceding ACTs)

EF: Early Finish (EF = ES + Dur)

LS: Late Start (LF - Dur)

LF: Late Finish (LF = earliest LS of all succeeding ACTs)

FF: Free Float

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

TF: Total Float

INTF: Interfering Float

INDF: Independent Float

1.5 EXAMPLE

The calculation process is divided into two major steps (a) Forward Pass, and (b) Backward pass. An example taken from Ref. (29) is used to illustrate the CPM calculations, see figure (1.3). Assume that the project starts at the end of day 0. In this case, ES = 0 for all beginning activities.

1.5.1 FORWARD PASS: It is a process to find earliest start times and earliest finish times for all activities.

1. A starts on day 0 (ES = 0) and finishs on days 3 (EF = ES + Dur = 0 + 3).
2. B, C and D start on day 3 and finish on day 6, 8, and 10, respectively.
3. E, F, and d1 start on day 6 and finish on days 10, 12, and 6, respectively.
4. G starts on day 8 (after C) and finishes on day 12.
5. I and d3 start on day 10 and finish on days 10 and 19, respectively.
6. H and d2 start on day 10 and finish on days 16 and 10, respectively.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

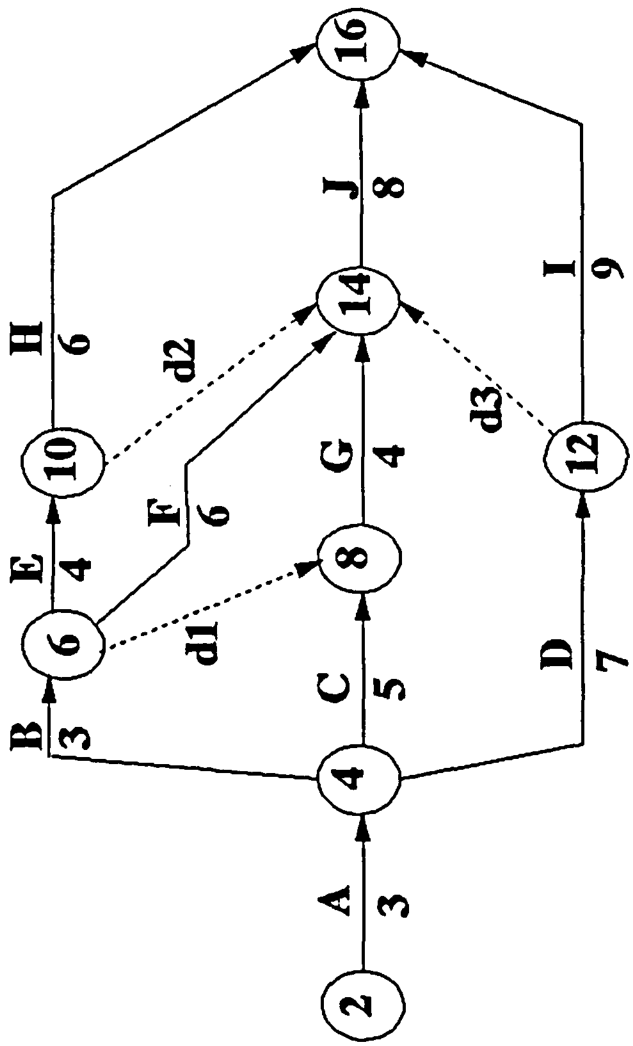


Figure (1.3).--NETWORK EXAMPLE

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

7. J starts on day 12 (after d2, F, G, and d3) and finishes on day 20.

The earliest time activity J can finish is day 20; therefore, the shortest time needed to complete the project is 20 days.

<u>**1.5.2 BACKWARD PASS:**</u> A process to find latest start times and latest finish times for all activities.

The backward pass will determine a late-finish schedule--the latest each activity can finish without delaying the completion of the project, in this case on day 16. It is assumed that the project should be finished as soon as possible, which is usually the case.

1. H, I and J must finish on day 20 (LF of J) in order for the project to finish on day 20, so they must start on days 14, 12, and 11, respectively (LF - Dur).
2. d2, F, G, and d3 must finish by day 12, so they must start by days 12, 6, 8, and 12, respectively.
3. E must finish by day 12 (as determined by the LS of d2), so it must start by day 8.
4. D must finish by day 11 (as determined by the LS of I), so it must start by day 4.
5. C and d1 must finish by day 8, so they must start by days 3 and 8, respectively.
6. B must finish by day 6 (as determined by the LS of F), so it must start by day 3.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

7. A must finish by day 3 (as determined by the LS of B and C), so it must start by day 0.

1.5.3 TOTAL FLOAT: Total float as defined by Ref. (29) is "the maximum amount of time an activity can be delayed without delaying project completion" --see figure (1.4.a). Total Float is calculated by equation (1.1).

$$TF_{ij} = LS_{ij} - ES_{ij} = LF_{ij} - EF_{ij} \qquad [1.1]$$

For the project at hand, the total float values are:

1. A : $TF_{2\text{-}4}$ = (LS) 0 - (ES) 0 = (LF) 3 - (EF) 3 = 0
2. B : $TF_{4\text{-}6}$ = 3 - 3 = 0.
3. C : $TF_{4\text{-}8}$ = 3 - 3 = 0.
4. D : $TF_{4\text{-}12}$ = 4 - 3 = 1.

The remaining results are shown in Table (1.1) and figure (1.5).

1.5.4 THE CRITICAL PATH: Any continuous path, from the starting node to the finishing node, that contains only activities with TF = 0 is a critical path (Ref. 28). Each network project should have at least one critical path. The critical path in this example is [B-E].

It is important to notice that there can be any number of critical paths through the network. One path can spread out into a

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

number of paths. A number of critical paths can converge into one. However, the critical path(s) must be a continuous chain of activities.

1.5.5 FREE FLOAT: Free Float as defined by Ref. (29) is "the maximum amount of time that an activity can be delayed without delaying the earliest time any succeeding activity can start" -- see figure (1.4.a)

$$FF_{ij} = ES_{jk} - EF_{ij} = LS_{jk} - LF_{ij} \qquad [1.2]$$

1. A: $FF_{2\text{-}4}$ = $(ES_{4\text{-}6})$ 3 - $(EF_{2\text{-}4})$ 3 = 0.
2. B: $FF_{4\text{-}6}$ = 3 - 3 = 0.
3. C: $FF_{4\text{-}8}$ = 3 - 3 = 0.

The remaining values are displayed in Table (1.1) and figure (1.5).

Note: Even though d1 goes between critical paths, it is not critical because its ES = 6 and its LS = 8.

To check CPM calculations, four steps are needed:

1. Start and finish activities must be on a critical path.
2. All critical paths must be continuous from start to finish.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

3. Total float must be greater than or equal to free float for every activity.

4. Any activity with TF = 0 must be on a critical path.

1.5.5 INTERFERING FLOAT: It is the span of time in which the completion of an activity may occur without delaying the termination of the project, but within which its completion will delay the start of some following activities - see figure (1.4.a)

$$INTF_{ij} = TF_{ij} - FF_{ij} \text{ or } LF_{ij} - ES_{jk} \qquad [1.3]$$

1.5.7 INDEPENDENT FLOAT: Independent Float as defined by Ref. (46) is "The time span in which the completion of an activity may occur and not delay the termination of the project, not delay the start of any following activity, and not be delayed by any preceding activity" -- see figure (1.4.a).

$$INDF_{ij} = ES_{jk} - LF_{hi} - Dur_{ij} \qquad [1.4]$$

Note that INDF is always less than FF.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

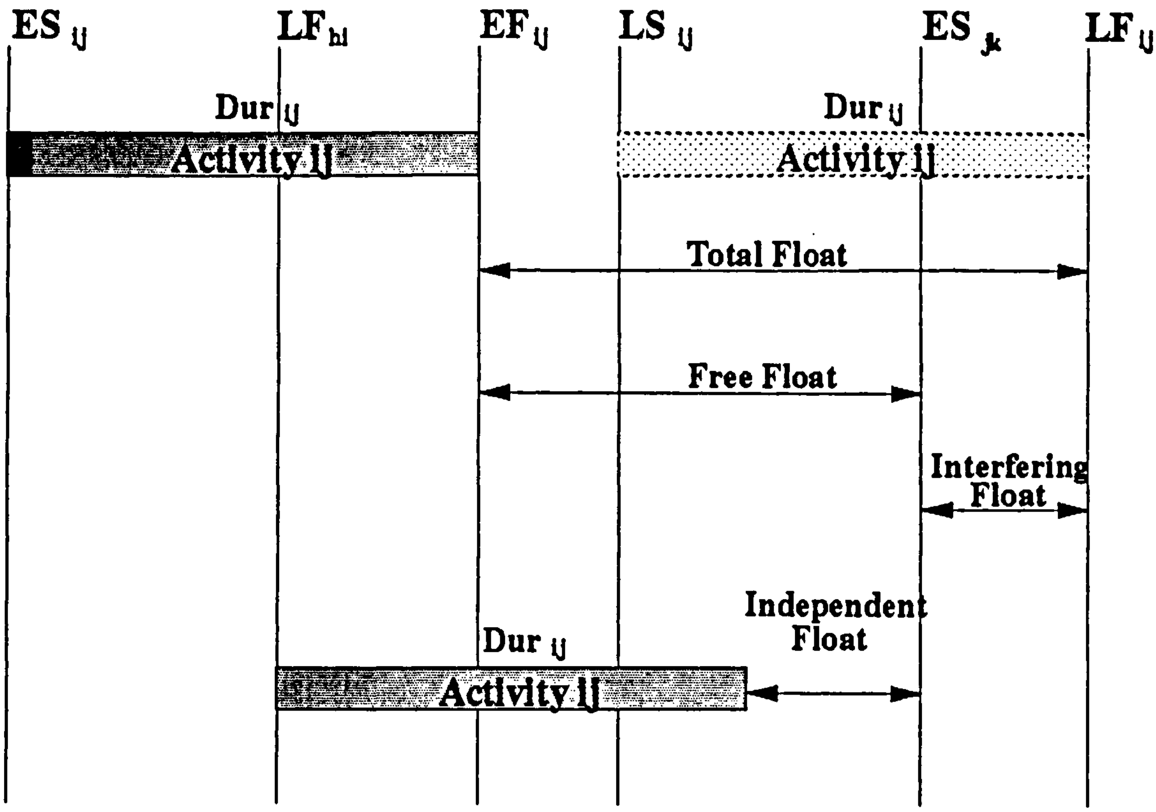
1.5.8 ALLOCATED FLOAT: is obtained by dividing up the total float along a path and allocating the parts to individual activities along that path. Total Float is often allocated in proportion to the duration of the activities.

1.5.9 EXTERNAL FLOAT Suppose that the contractual completion time is 210 workdays and the forward pass shows an early completion on day 190. If the backward pass starts on day 210, it ends up with critical activities having TF = 20 and noncritical activities having TF > 20. Such total float is sometimes called External Float. It indicates that the project can finish 20 days earlier than its contractual completion time. The External Float may be used for allocating priorities when several separate projects are competing on a scarce resource, for example -- see figure (1.4.b).

1.5.10 NEGATIVE TOTAL FLOAT[5]: To demonstrate the concept of negative total float, assume that the contractual completion time for a given project is 180 workdays and the early completion time as set by the forward pass is 190 workdays. If the backward pass starts at day 180 (contractual completion date), it arrives back at the project start on day (-10). All critical activities will then have a TF = - 10 and noncritical activities a TF > - 10 -- see figure (1.4.c).

Since total float can be either negative, zero, or positive, the critical activities are those with the smallest TF.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.



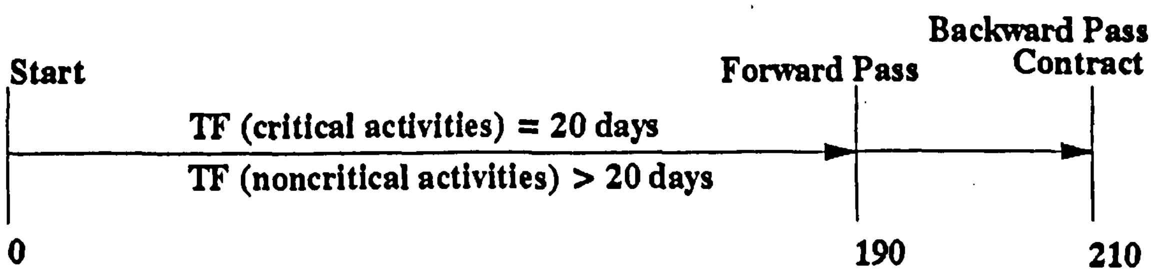
Figure(1.4.a).--CPM Basic Floats

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

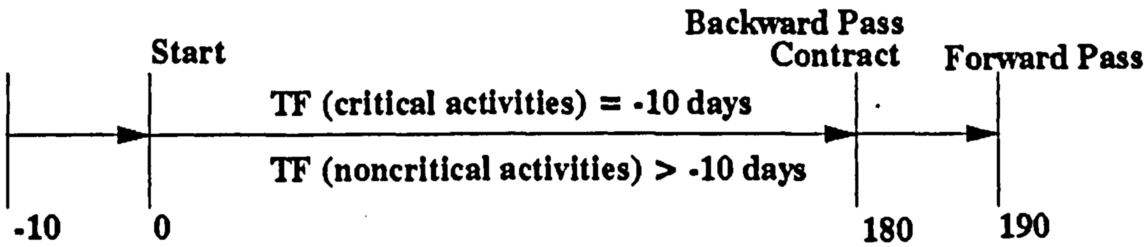
TABLE (1.1).--CPM CALCULATIONS

Act.	i-j	Duration	ES	EF	LS	LF	TF	FF	INTF	INDF
A	2-4	3	0	3	0	3	0	0	0	0
B	4-6	3	3	6	3	6	0	0	0	0
C	4-8	5	3	8	3	8	0	0	0	0
D	4-12	7	3	10	4	11	1	0	1	0
E	6-10	4	6	10	8	12	2	0	2	0
F	6-14	6	6	12	6	12	0	0	0	0
G	8-14	4	8	12	8	12	0	0	0	0
H	10-16	6	10	16	14	20	4	4	0	2
I	12-16	9	10	19	11	20	1	1	0	0
J	14-16	8	12	20	12	20	0	0	0	0
d1	6-8	0	6	6	8	8	2	2	0	2
d2	10-14	0	10	10	12	12	2	2	0	0
d3	12-14	0	10	10	12	12	2	2	0	1

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.



Figure(1.4.b).--External Float



Figure(1.4.c).--Negative Float

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

1.6 CONVENTIONAL OPTIMAL SCHEDULE

Initially, CPM calculations provide a networking-based plan, and a set of earliest and latest start and finish times for each activity making up the project. The earliest finish time for the network terminal event determines the "normal" project duration time based on "normal" activity durations. In this context, "normal" means that these durations are estimated based on the assumption that the availability of resources is unlimited. It is essential at this stage to check whether or not the project duration satisfies time constraints imposed on the project, or to evaluate the project duration in terms of cost.

Time constraint usually arises when the owner requires a specific project completion time; when the original time constraint changes after a project has started, requiring new project planning; or, when delays occur in the early stages of a project. The planner, in this case, will try to save time at minimum cost. This means, of course, that the durations of some activities may be shortened (crashed) by allocating more resources to them. Crash time is defined as the shortest time within which an activity can be accomplished by using more crews, overtime, extra shifts, or any combination of these three. Since there is a cost increase involved in shortening an activity duration from normal to crash, these resources should be estimated and translated into a direct cost per

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

time unit. The objective, here, is to crash the project's duration at a minimum increase in the project direct cost. To accomplish this goal, the least-cost method is used.

The least-cost method is an optimization process whereby some project activities are crashed in order to shorten the overall project length[2,25,29]. When activities are shortened, their direct costs increase. When the overall project is shortened, its overhead costs decrease. Table (1.2) shows the normal time and cost as well as the crash time and cost for each activity in the network, figure (1.5). The least-cost method assumes the following:

1. The planned time of an activity can be any whole day value between normal and crash time.
2. The direct cost of an activity is linear between normal and crash times.
3. The overhead cost is linear during the entire project.

1.6.1 Example

Another example from Ref. (29) will be used to illustrate the least-cost method. All values were kept the same except for the crash cost of activity E which was changed from $600 to $800. This was done to improve the example so as to get a definite minimum point in the total cost. The network is shown in figure (1.5) and the cost calculations in table (1.2). The steps used in the calculations are as follows: as follows:

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

TABLE (1.2).--Crash Information for Project XYZ

Activity	Duration		Cost, $		△	△	△
	Normal	Crash	Normal	Crash	Cost	Days	Cost / Days
A	4	2	400	500	100	2	50
B	8	5	800	980	180	3	60
C	3	2	600	700	100	1	100
D	10	6	500	800	300	4	75
E	8	6	800	950	150	2	75
F	7	4	700	1000	300	3	100

Project Duration	16 days
Direct Cost	$3,800
Overhead Cost	$1,600
TOTAL COST	$5,400

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

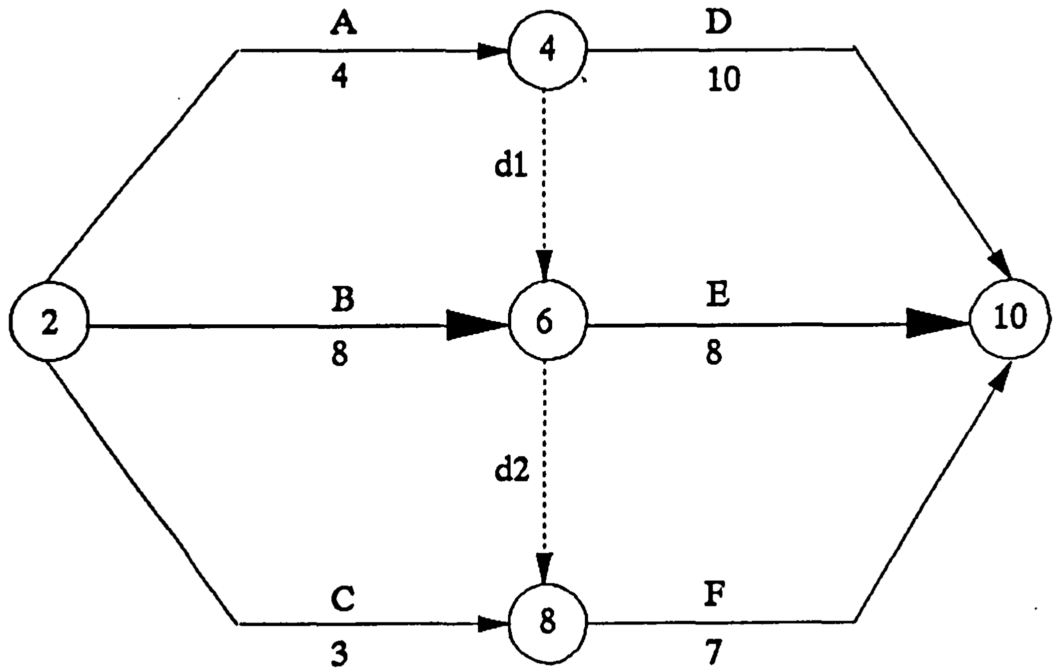


Figure (1.5).--Project XYZ

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Step 1: Determine the normal durations and the normal costs of the network activities.

Step 2: Determine the crash durations and the crash costs of the network activities. Table (1.2) shows these values.

CYCLE 1

Step 3: Determine the activities that will shorten the project completion time, i.e., the critical activities. These are activities B and E.

Step 4: From these activities, select the activity or activities that will shorten the project at the least cost. From Table (1.2), activity B is selected since it can be shortened @ $60/day rather than $75/ day for E.

Step 5: Shorten activity B by 1 day. Accordingly, activity B has a new duration of 7 days.
Step 6: Update the network. The project's new duration is 15 days, and the new cost is $5,360 ($5,400 + 1 x $60/day - 1 x $100/day = $5,360).

CYCLE 2

By repeating Steps 3 through 6, the project can be shortened by one more day.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Step 3: Activities B and E

Step 4: Activity B @ $60

Step 5: Shorten activity B by 1 day. Its new duration is 6 days.

Step 6: The project's new duration is 14 days, and the new cost is $5,320 ($5,360 + 1 x $60/day - 1 x $100/day = $5,320).

CYCLE 3

In this cycle there are multiple critical paths to shorten.

Step 3: Activities (B or E) and (A or D)

Step 4: B and A @ ($60 + $ 50) = $110 is the cheapest choice.

Step 5: Shorten both B and A by one day. B has a new duration of 5 days and A has a new duration of 3 days.

Step 6: The project's new duration is 13 days, and the new cost is $5,305 ($5,320 + $110/day - $100/day = $5,330).

By repeating the same steps, Table (1.3) can be obtained, and the optimal network (in terms of cost) can be determined as shown in figure (1.6).

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

TABLE (1.3).--Crash Analysis for Project XYZ

Activity	Δ Cost / Days	Days Shortened				
A	50			1	1	
B	60	1	1	1		
C	100					
D	25					1
E	75				1	1
F	100					
Days Cut		1	1	1	1	1
Proj. Dur.	16	15	14	13	12	11
Inc. Cost/Day		60	60	110	125	150
Dir. Cost	3,800	3,860	3,920	4,030	4,155	4,305
Ovhd. Cost	1,600	1,500	1,400	1,300	1,200	1,100
TOTAL COST	5,400	5,360	5,320	5,330	5,355	5,405

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

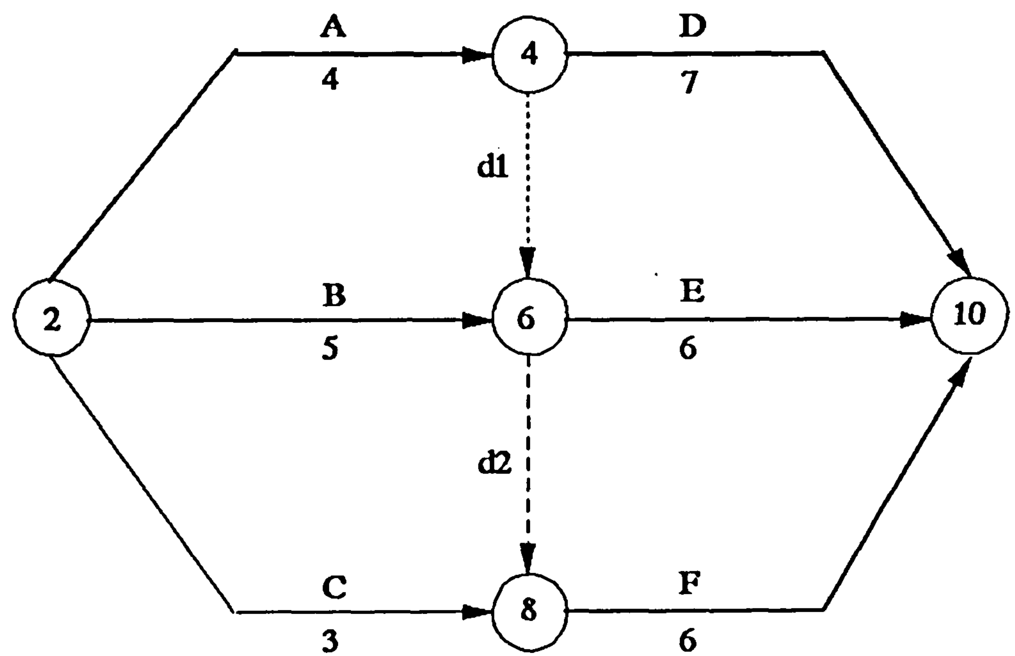


Figure (1.6).--Final Network for Project XYZ

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

1.7 EMBELLISHMENTS OF CPM

Over the last three decades, certain CPM-based techniques have been developed to improve the method. These techniques use different approaches in order to determine an optimal schedule which has the minimum possible time at the minimum cost. Some significant techniques are discussed in the following sections.

1.7.1 DECISION CRITICAL PATH METHOD (DCPM)

Unlike CPM, DCPM introduces decision nodes in the network. Each decision node leads to possible alternative methods for executing an activity. For example, an activity might have a constraint labor, equipment, material, or weather which may lead to different decision alternatives for executing this activity. Example network from Ref. (37) is used to illustrate this method.

1.7.1.1 DCPM ALGORITHM -- The activities of the DCPM network, shown in figure (1.7), are defined by circular nodes, and decision nodes are represented by triangles. These nodes are mutually exclusive. Each decision alternative j of activity i, i.e. A_{ij}, is associated with a time, t_{ij}, a cost, c_{ij}, and an integer, d_{ij}= 0,1. The value of the integer is set at 1 for the selected alternative and 0 for all the others. The mutually exclusive requirement indicates that d_{ij}= 1, i.e., each activity can be performed by using only one alternative.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Assume a network with **n** activities and activity A_i can be executed in r(i) alternative ways, where r(i) = integer => 1. Also assume that **p** is the penalty per time unit, **w** is the reward per time unit, and **D** is the completion time. The following integer linear programming algorithm is used to select the optimal network and its critical path:

Objective Function:

$$Minimize\ Total\ Cost = \sum_{i=1}^{n} \sum_{j=1}^{r(i)} d_{ij}\, c_{ij} - w\, \Delta^{-}_{F} + p\, \Delta^{+}_{F} \qquad [1.5]$$

Subject To:

$$S_i + t_i \leq S_m \qquad [1.6]$$

for all A_i << A_m, where << indicates a precedence relation and in which A_i and A_m are activities without alternatives.

$$-M\,(1 - d_{ij}) + S_{ij} + t_{ij} =< S_m \quad \forall\ activities\ with\ alternatives \qquad [1.7]$$

$$\sum_{j=1}^{r(i)} d_{ij} = 1; \quad for\ \ i = 1, \ldots, n \qquad [1.8]$$

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

where:

D = scheduled completion time.

TF= earliest finish time as determined from the critical path.

$$\Delta^-_F = D - T_F \quad if \quad T_F \leq D \tag{1.9}$$

$$\Delta^+_F = T_F - D \quad if \quad T_F > D \tag{1.10}$$

S_i= early start of activity A_i, if without alternatives

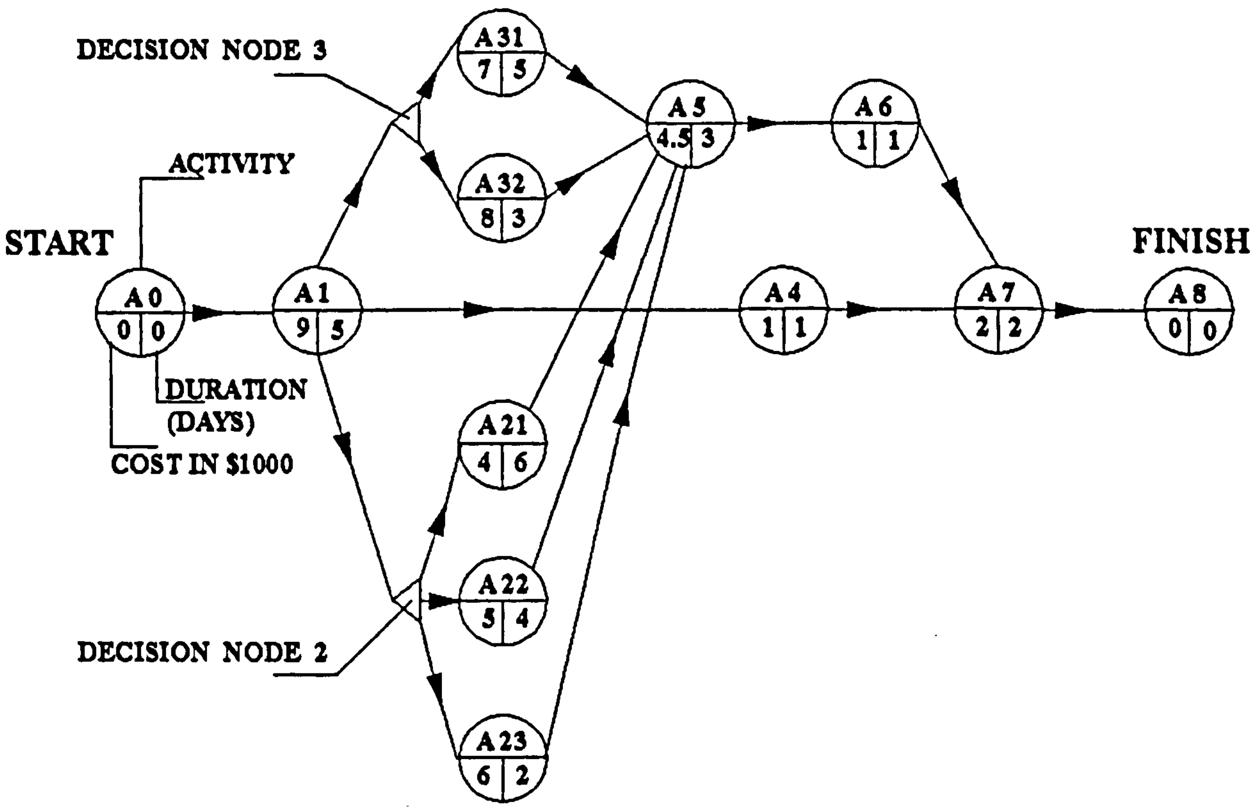
t_i= duration of activity A_i, if without alternatives

S_{ij}= early start of alternative j, of activity i, i.e., A_{ij}

t_{ij}= duration of A_{ij}

M = large number such that if A_m is an activity without alternatives, and A_{ij} is an activity with alternatives, the inequality holds only if d_{ij} = 1.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.



Alternatives	Completion Time Days	Direct Cost in $1000	Total Cost	
A21 - A31	17	28.5	31.5	
A22 - A31	16	29.5	31	
A22 - A32	15	30.5	30.5	BEST
A23 - A31	16	30.5	32	
A23 - A32	14	31.5	31	

Total Cost = Direct Cost + Penalty - Reward
Penalty = $1,500/day
Reward = $500/day

Note: It is assumed that if activity A21 is chosen, then activity A32 cannot be chosen concurrently.

Figure(1.7).--Example of Decision CPM Network

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

1.7.2 SOFT LOGIC IN NETWORK ANALYSIS (SOFTCPM)

SOFTCPM is a computerized CPM-based algorithm. It deals with a soft logic network, whereby the activities in an established network can be arranged in a variety of logical sequences. This situation is usually encountered, during the control phase, when some technically independent activities are constrained by external factors such as limited resources. In this case, SOFTCPM deals with all possible combinations as a result of the available constraints.

SOFTCPM starts by establishing a network logic. Next, the algorithm distinguishes between two sets of activities: regular (noninterchangeable) and interchangeable. An interchangeable set is one whose activities are technically independent and can be performed simultaneously. A subset, which is a branch of a set, includes all activities in a particular set that are in series. The total number of subsets equals the total number of activities that can be performed at one time.

To be considered for scheduling, activity A, for instance, must satisfy two conditions. It must be scheduled and not completed, and all its preceding activities must be completed. Next, SOFTCPM considers the initial resource constraints imposed on activity A and ensures that resource utilization will not exceed this limit. As a result, the activity set and subset are determined automatically. Before activity A can start, SOFTCPM conducts certain transformations:

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

1. It tries to exchange activity A with an incomplete activity within the same subset. This requires that none of the activities in the subset is in progress.

2. If one activity has already started, activity A should be exchanged with another activity from a different subset. This exchange may change the project completion time and the critical path. To minimize this effect, SOFTCPM attempts to exchange the first incomplete activity (within the set) with the activity having the minimum change in duration, and then to exchange this activity with activity A.

3. If the last activity in one of the other subsets is a complete activity, activity A must be deleted from its position and added to the later subset after a complete activity.

CPM computations are performed after each updating period, and a new project completion date and critical path are determined. This method will be explained further by the use of an example from Ref. (50) shown in figure (1.8)

1.7.2.1. EXAMPLE

The project network shown in figure (1.8.a) is to be updated at the end of days 2, 6, 13 and 15. Assume that after finishing activity 1, the contractor decided to execute activities 4, 5, and 10.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

End of Day 2.--SOFTCPM exchanges activities 2 and 4, then activities 8 and 10. Since the exchanges are done between activities in series (same subset), they have no effect on either the project completion date or on the critical path. The new logic is demonstrated in figure (1.8.b).

End of Day 6.--Activity 10 has been completed in four days. Instead of starting activity 9, the contractor has decided to start activity 3. Since activities 4 and 5 are still in progress, SOFTCPM exchanges activity 3 with another activity that is in series with activity 10 and has the same duration (or minimum change in duration) as activity 3. Activity 9, which follows activity 10, requires 2 more days than activity 8. Therefore, activities 8 and 9 are initially exchanged, then activities 8 and 3 have been changed. Even though this exchange has been made between activities that are not in series (i.e. different subsets), it has no effect on either the project completion date or the critical path since the change in duration between exchanged activities is zero. The new logic is shown in figure (1.8.c).

End of Day 13.--Activity 3 has encountered an unexpected delay, and its new duration becomes 9 days. Activity 6 has been completed in 4 days instead of 5 days, and activity 8 has been completed as scheduled. The contractor has chosen to start activity 2 and 7 while performing activity 3. Figure (1.8.d) shows the up-to-date

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

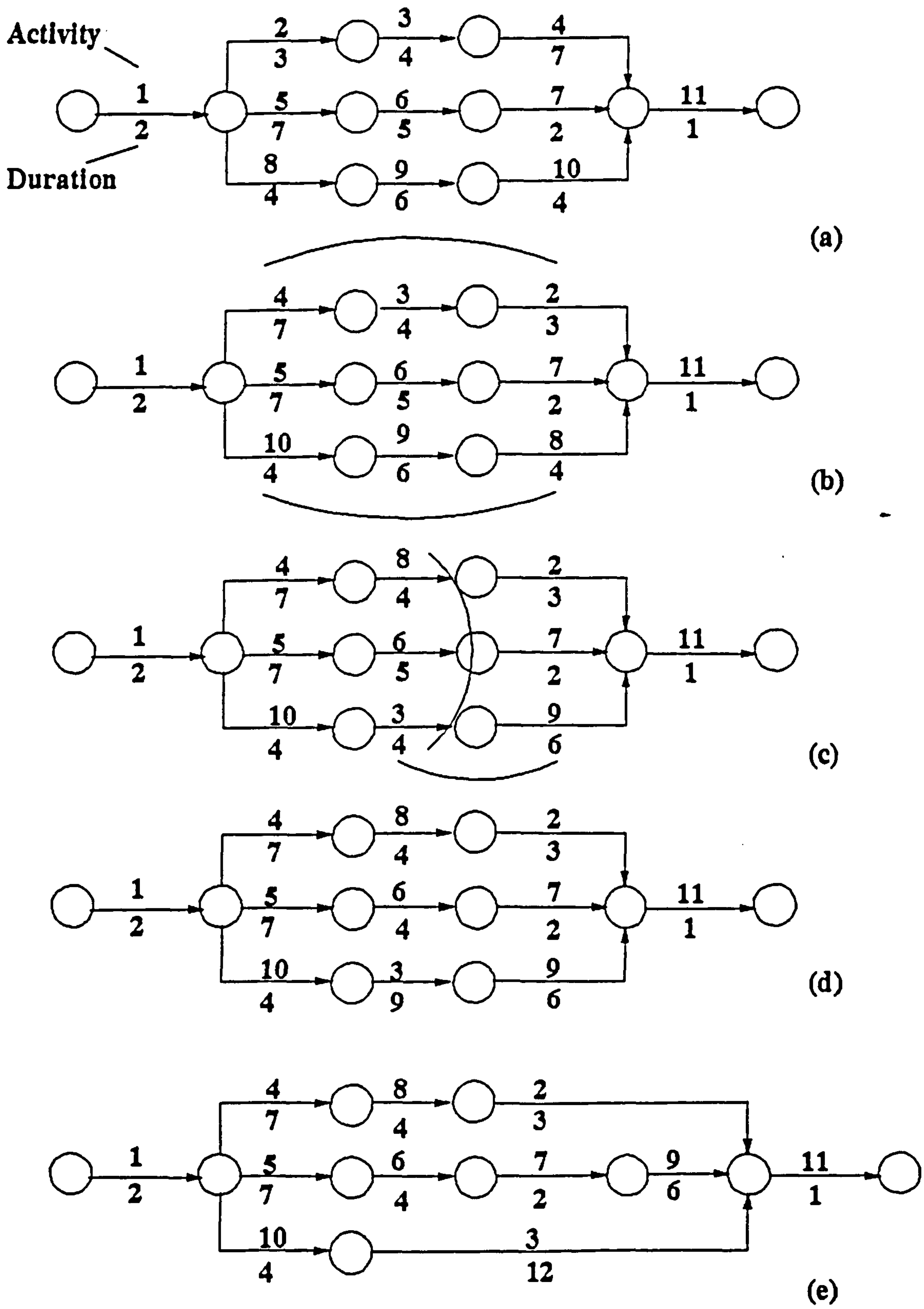


Figure (1.8).--Networks for Example Project During Different Updating Periods

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

logic. Consequently, the new project completion date becomes 22 days, i.e., five days of delay.

End of Day 15.--Activity 7 has been finished as originally scheduled and activity 3 is still in progress. Its new estimated duration is 12 days. The decision has been made to start activity 9 while performing activities 2 and 3. Since activity 7 has been completed, activity 9 has been deleted from its position and added after activity 7 to satisfy the transformation conditions. Figure (1.8.e) shows the final updated network.

1.7.3 LINEAR PROGRAMMING SOLUTION TO NETWORK COMPRESSION

The following method is based on chain-bar chart and Linear Programming. It starts by computing the CPM schedule for a hypothetical project. Next, a chain-bar chart similar to figure (1.9) is drawn based on the earliest start time of all activities. Then, using Linear Programming this chart is compressed. The minimum cost incurred as a result of compressing the network plus the total direct cost for the uncompressed network gives the final cost of the project. An example from Ref. (44) will be used next to illustrate this method.

1.7.3.1 FORMULATION OF LINEAR PROGRAMMING MODEL

The formulation of a Linear Programming model to compress the network is accomplished by using the chain-bar chart, maximum crash

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

durations, and the cost slope of activities. Table (1.4) shows activity durations and their crash costs and the chain-bar chart for the project is shown in figure (1.9).

OBJECTIVE FUNCTION: The objective is to minimize the overall cost of crashing the activities in order to reduce the project completion time. From Table (1.4),

Minimize: COST(Z) = 60A + 10B + 20C1 + 20C2 + 5D1 + 5D2 + 30E + 50F + 15G1 + 15G2 + 12H + 6I + 14J + 9K + 11L + 13M [1.11]

SUBJECT TO:

(a) Crash Duration Constraints [1.12]

A<=2; (i.e., crash duration for activity A <= 2 days)

B<=7; C1<=1; C2<=3; D1<=2; D2<=8; E<=5; F<=2; G1<=1; G2<=6;

H<=9; I<=7; J<=6; K<=10; L<=8; and M<=4

(b) Critical Path Constraint [1.13]

A + B + C2 + D2 + E + F = 20

(c) Loop Constraints [1.14]

The loop constraints take into consideration the impact of compressing certain activities in one chain on activities in other chains. In case of having more than one critical path, the loop constraint will take care of the activities in the other critical paths.

B - C1 <= 7

B + C2 - D1 <= 6

B + C2 - G1 <= 5

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

D2 - G2 <= 3

B + H - J - K <= 2

C2 + D2 - H - I <= 2

C2 + D2 + E - H - I - M <= 2

I - L <= 1

Loop constraints can be only formulated for a valid loop. This means that there can be floats either before or after the loop but not both. For example, the loop containing activities C2, D2, I and H is a valid loop, since it has no float before its begining, and the loop containing D1 and G1 is not since it has floats on both sides.

As a final step, the previous Linear Programming problem can be solved by using any available computer program to obtain the crash schedule.

The previous techniques have improved the use of CPM in several regards:

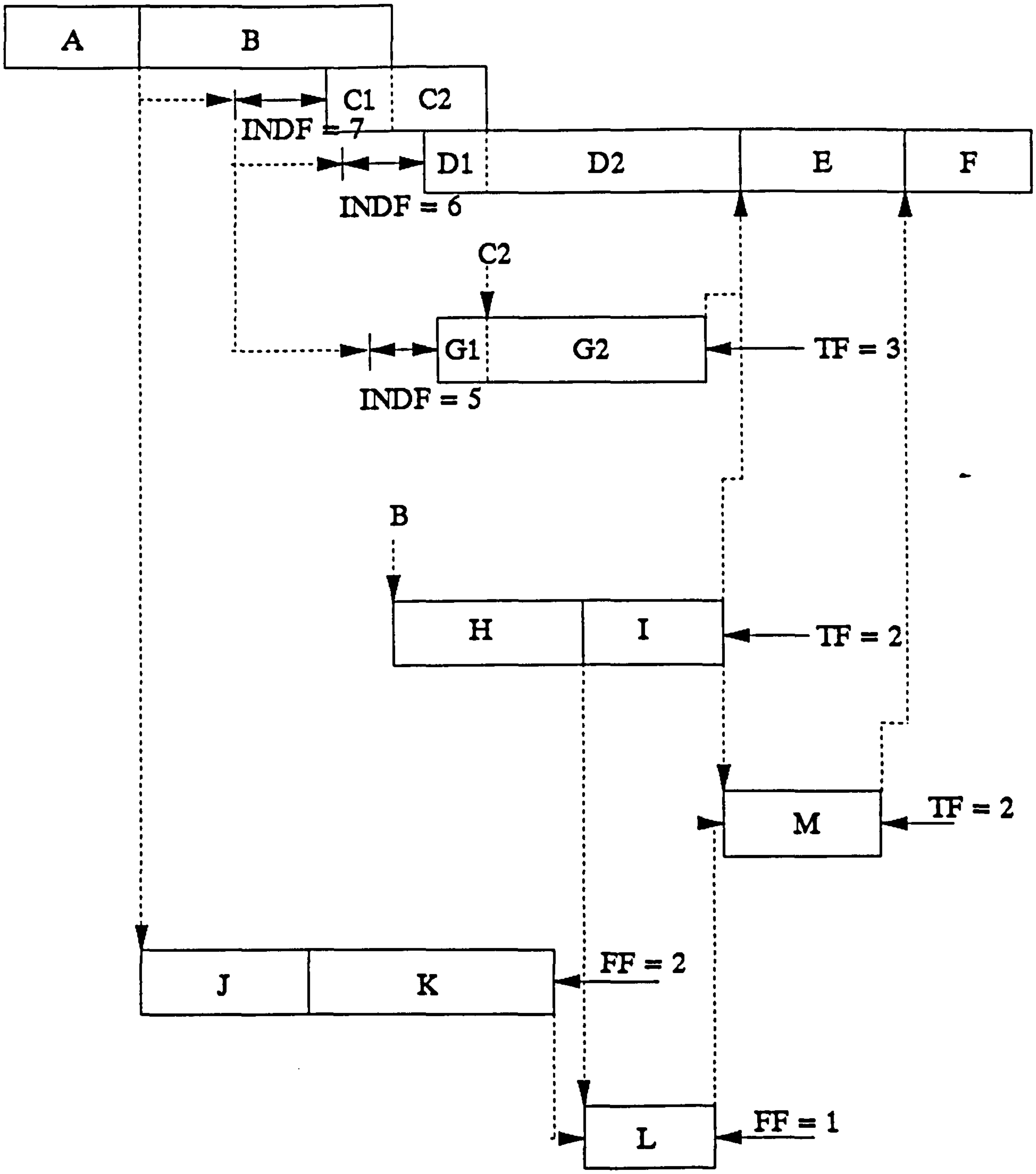
1. The idea of introducing decision alternatives and attempting to minimize the total cost of the project based on the available alternatives reduces the rigidity of the deterministic structure of conventional CPM. This allows the planner to develop a more realistic schedule.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

TABLE (1.4).--Details of Activity Durations and Crash Costs

Activity	Durations, in days		Maximum crash duration	Cost of crashing per day in dollars
	Normal	Crash		
A	15	13	2	60
B	30	23	7	10
C1	5	4	1	20
C2	10	7	3	20
D1	8	6	2	5
D2	25	17	8	5
E	20	15	5	30
F	15	13	2	50
G1	7	6	1	15
G2	22	16	6	15
H	18	9	9	12
I	15	8	7	6
J	20	14	6	14
K	26	16	10	9
L	14	6	8	11
M	20	16	4	13

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.



Figure(1.9) Chain Bar Chart for Hypothetical Project

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

2. SOFTCPM enhances the flexibility of CPM by allowing the exchange of similar activities under resource constraints.

3. SOFTCPM deals with resource constraint only.

4. The chain-bar chart along with Linear Programming technique are used to produce optimal schedule. It is a modified version of the least-cost method.

5. It considers time and cost only to obtain the so called optimal schedule.

1.7.4 COMMENTS ON THE EMBELLISHMENTS OF CPM

The previous techniques have improved the use of CPM in several regards:

1. The idea of introducing decision alternatives and attempting to minimize the total cost of the project based on the available alternatives reduces the rigidity of the deterministic structure of conventional CPM. This allows the planner to develop a more realistic schedule.

2. SOFTCPM deals with resource constraint only.

3. SOFTCPM enhances the flexibility of CPM by allowing the exchange of similar activities under resource constraints.

4. The chain-bar chart along with Linear Programming are used to produce an optimal schedule. This method is a modified version of the least-cost method and considers only time and cost to obtain the so called optimal schedule.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

1.8 CRITICISM OF CPM

In recent years, CPM has been criticized for its inability to cope with the dynamic nature and variability of construction projects. Basically, CPM was developed for the construction and maintenance of chemical processing plants. It is imposed on the construction industry by people who are not familiar or concerned with labor scheduling. This makes the integration of control functions such as cost control, resource management and productivity improvement with CPM extremely difficult, expensive, and nonproductive[3,7,12,21,47].

Peer[41] discusses major shortcomings that CPM suffers from. "CPM's capabilities are constrained by its fundamental unrealistic assumptions of unlimited resources that can be hired and freed, and by independent activities of fixed duration that can be shifted freely between earliest start and latest finish. The need for creating working continuity and balancing the whole process into an integrated production system is completely neglected." Conventional CPM is expensive to run and to update as well[3]. Its status reports take time to reach managers and decision makers; by the time they receive these reports, the information contained in them tends to be out of date[6,12,21].

CPM is a time-consuming technique. The time spent in preparing its network using dates, activity periods, and logic could be used to perform different scheduling related operations[3,23].

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Birrel[7] states that "...the basic critical path network technique is neither a true model nor the best approximate model of the construction process..... The concept of PERT/CPM was created in the military/industrial environment where United States national security rightly put a low weight on cost control and efficient use of resources."

Although the methods discussed in section (1.7) have advanced the use of CPM in certain regards, they all have shortcomings:

1. The assumptions of these techniques do not take advantage of the use of the fundamental information available on the project, such as productivity rate, availability of resources, etc.
2. A practical and simple tool is needed to schedule and control construction projects. This is not the case with these methods.
3. The solution provided by DCPM network is lengthier and more costly than that of CPM[37].
4. The use of the chain-bar chart along with Linear Programming is time consuming, since it involves the calculations of CPM, the formation of the chain-bar chart, and the use of a computer program to solve the Linear Programming objective function. This solution would be more effective if a computer package that integrates these three steps were available.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

5. SOFTCPM is applicable only to activities in series that have the same type of operation. It is accurate only when the change in duration between the interchangeable activities is small. It also does not take dependencies into consideration.

It is thus obvious that a new planning method for construction projects based on heuristic analysis is needed for the construction process[7,32].

1.9 CPM-BASED SCHEDULING SOFTWARE

CPM-based scheduling computer programs have been developed to facilitate scheduling practices for construction managers. They perform CPM computations, and provide these managers with vast graphical and tabular representations of project scheduling information. These capabilities enhance and facilitate the effectiveness of scheduling practices. In addition, these programs are capable of presenting the network schedule in a time-scaled diagram, and/or a bar chart. This helps project managers in the preparation of job progress reports and other related information to top management as well as owners. Moreover, many programs allow the user to plan the utilization of resources in a project by generating a crew schedule, and material and equipment reports[36].

1.10 WHAT IS NEEDED ?

In a paper titled "Basic Research Needs in Construction

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Engineering", Carr and Maloney[9] presented and discussed basic research needs in the construction industry. The paper was based on the results of a workshop sponsored by the National Science foundation and the Construction Research Council of the American Society of Civil Engineers. They recommended the development of the following areas for future research.

1. A detailed micro level planning system that takes into account the flow of planning information within and between the constructor's and owner's organizations;
2. Techniques for analyzing construction engineering management problems and operations analysis;
3. Planning, scheduling, and updating tools which include dynamic and uncertain nature of activity dependencies and durations; and
4. Decision-making techniques for construction decisions.

In his paper, Peer[41] indicates that "For construction process to be balanced into a comprehensive system of production, future research must be devoted to the development of construction planning techniques based on input of fundamental production data, e.g., quantities of work, production rates, and other production characteristics." It is important to put emphasis on the need for establishing and studying the appropriate control scheme during the planning phase of a project. A planner should design a comprehensive plan for the project. Having established the logic of the project, the planner then moves on to the scheduling phase. In

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

this phase, CPM should be fully utilized for developing several scheduling alternatives, selecting the target schedule, and employing all kinds of floats to forecast potential deviation from the target schedule. This is something which CPM in its present format does not offer.

From the previous discussion, it becomes obvious that an effective planning-scheduling-control scheme should:

1. be able to measure the criticality of the target schedule network, and to adjust the network accordingly;
2. be able to make use of all kinds of floats, and to measure the degree of criticality of all project activities;
3. be flexible, effective, and practical to operate at minimal cost;
4. be able to accurately incorporate the project's conditions and constraints;
5. be capable of anticipating deviations from the target schedule; and
6. be simple in format in order to effectively communicate, the network plan.

1.10.1 THE LEARNING CURVE EFFECT[39]

It is well recognized that repetition of the same operation results in less time or effort expended on that operation per unit of production. The observed characteristic of the improved

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

performance is called "learning" which can be represented mathematically by the learning curve model.

The learning curve model is based on the following assumptions:

1. The amount of time and cost required to complete a unit of product is reduced each time the task is repeated.
2. The unit time will decrease at a decreasing rate.
3. The relationship between direct labor hours and the number of units produced follows a specific estimating model:

$$Y_n = Y_1 \; n^{-b} \qquad [1.15]$$

where:

Y_n = the effort (hours/unit) it will take to produce the nth unit

n = unit number

Y_1 = the effort it will take to produce the first unit

b = improvement factor (log r)/(log 2)

r = learning factor (percent learning curve)

b is negative because the effort decreases with increasing production.

It is assumed that an 80 percent learning curve represents many large-scale repetitive operations, i.e., a 20 percent reduction in worker-hours per operation unit between doubled units.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

For example, if it takes 100 hours to produce unit 1, it would take 100(0.8) = 80 hours to produce unit 2. Similarly, unit 4 would take 80(0.8) = 64 hours, and so on.

Although the learning curve model has been available for a long time, its impact on improving productivity has not been widely considered by the construction industry. This is confirmed by results of a planning questionnaire that was recently conducted[3]. In fact, it has been found that only 16.66% of the interviewed construction companies use the Learning Curve during the project life cycle.

In this research, it is intended to incorporate the learning curve model into the new system.

1.11 OBJECTIVES AND SCOPE OF RESEARCH

A sound planning-scheduling-controlling scheme cannot be only involved with determining a critical path from deterministic activity durations and following the execution of the critical activities. This scheme should utilize fundamental activity information such as productivity rates, crew size, time, and cost. It should be also capable of producing different schedules that incorporate time, crew size, and cost constraints. In addition, the scheme should be able to predict possible future deviations from the target schedule. Moreover, it should be able to measure the

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

criticality of the project network. Conventional CPM as well as the techniques previously presented lack this integrity.

Based on the previous discussion, the prime concern of this research is to advance the use of CPM by the construction industry by making it more indispensable for construction practices. To achieve this goal, the focus should be on the planning-scheduling-controlling cycle. Consequently, the research objectives of this thesis are:

1. to develop a planning-scheduling-control technique that:
    a. utilizes fundamental activity information such as productivity rates, crew size, time, and cost;
    b. incorporates the learning curve model;
    c. provides four CPM-based scheduling alternatives under different scenarios and constraints;
    d. analyzes the criticality of scheduling alternatives and their activities;
    e. incorporates control criteria and uses a decision support system that can be used in selecting a target schedule; and
    f. forecasts the potential of each activity to deviate from the target schedule, and
2. to computerize the proposed system so developed.

1.12 RESEARCH METHODOLOGY

The author starts this research by thoroughly reviewing the literature and collecting information related to planning,

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

scheduling and control practices. This step is followed by problem definition.

The information collected is then used to design a planning questionnaire which is mailed to a selected group of construction companies in the United States. Next, the response from the construction companies are studied and analyzed.

The findings, along with the information gathered from literature, are used to determine the practical needs of the construction industry. This information is then used to design a CPM-based technique for planning, scheduling and controlling construction project. As a final step, this technique is computerized and an illustrative example is used to facilitate its use by the construction industry.

1.13 DISSERTATION ORGANIZATION

This dissertation is divided into six chapters. In chapter one, the historical background of CPM, and its shortcomings are thoroughly presented and discussed. The objectives and scope of this research are then defined and stated. Chapter two presents the results and findings obtained from the planning questionnaire. It also discusses the needs of the construction industry in order to

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

enhance the planning and control practices. In chapter three, the algorithm of the computerized multi-dimensional support system (CM-DSS) is presented and explained along with its flow chart. All the sub-algorithms are also presented and discussed. Chapter four presents and discusses the CM-DSS computer model, the major element of the computer program, and the required inputs and outputs. In chapter five, a full step-by-step computer session with CM-DSS is presented. A final output for an illustrative example with a comprehensive analysis is also presented. Chapter six presents final conclusions, limitations, and direction of future research. All the references used in this research are included in the references section. This dissertation includes three appendices. Appendix (I) includes a copy of the planning questionnaire employed in this research. Appendix (II) incorporates a printout of the original source code of CM-DSS. Appendix (III) contains the actual printouts of the illustrative example used in chapter five.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

CHAPTER TWO

QUESTIONNAIRE

2.1 INTRODUCTION

A recent survey was carried out on a sample of 200 construction companies in the United States. The primary goals of the survey are: a) to understand and analyze, by means of a questionnaire, the techniques that are usually used by these companies to plan and control projects, and b) to specify the kind of improvements, if any, that are needed to advance these techniques.

The sample was selected to represent a wide range of areas of specialty and volume of work handled by contractors as listed in DUN's Directory 1991. Of the 200 questionnaires mailed, 60 were returned, for an overall response rate of 30%. The respondents were then divided into three groups according to the companies' annual sales volume: Group A (36.7% or 22) had annual billings of more than $500 million; Group B (38.3% or 23) had annual billings in the range of $100 million to $500 million; and Group C (25% or 15) had annual billings between $50 million and $100 million.

This questionnaire comprises the first major part of this research. In the following sections, the results and findings

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

obtained from the analysis of the questionnaire are presented and discussed along with conclusions.

2.2 PURPOSE OF THE STUDY

In recent years, planning and control techniques have been criticized because of their inability to assume an influential role in running construction projects. Many researchers have addressed the need for improving the available methods and developing better ones[6,7,26,41]. It is believed that the initial step toward developing more efficient techniques must start with understanding planning and control practices used by the industry. As a result, the author conducted a questionnaire survey--see Appendix (I). The major objectives of this survey were to determine:

1. What planning and control (P&C) practices are being used by the construction industry?
2. Which factors are the most important to construction companies when they plan projects?
3. Which criteria are considered to be significant during project control?
4. What sort of improvements, if any, are needed to help advance P&C practices?
5. Do significant differences in P&C practices exist among construction companies?
6. What role does experience play in planning and controlling construction projects?
7. What are, if any, the major shortcomings of CPM?

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Based on these objectives, the questionnaire was divided into six sections. The first section contained questions concerning type of contractor, annual billings, number of permanent engineers, average job size, average project duration and region of operation. The second section targeted the background of the respondent in order to understand the role of experience in P&C practices. In the third and fourth sections, respondents were asked to specify the methods, factors and performance goals that they use to plan and control their projects. Finally, the fifth and sixth sections were designed to understand how construction companies manage human resources and the extent to which computer facilities are being used in P&C practices.

2.3 DEVELOPMENT OF THE QUESTIONNAIRE

This questionnaire is an extended and refined version of the original questionnaire which was developed by the author during his master's program[5]. Its design is based on a personal investigation by the author. After the first draft was completed two copies were sent to Professor George Suckarieh, Department of Construction Science, University of Cincinnati, and Mr. Tom Keciks, Vice President of Messer Construction company, Cincinnati, Ohio, in order to determine:

1. If the terms/language were clear;
2. If the questions were understandable; and
3. if any correction to the questionnaire could be suggested.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

As a result of the validation process, some suggested changes to the questionnaire were adopted.

2.4 CHARACTERISTICS OF RESPONDENTS

Companies involved in this questionnaire were asked to identify their specialty. Table (2.1) gives information on their replies. 63.3% of the respondents are in "Commercial" construction and 48.3% are in both "Industrial" and "Highway & Heavy" construction. Only, 8.3% are in "Residential." The average job size distributions and project durations are also listed in the table. Figure (2.1) displays the geographical regions in the United States where the respondents are operating. Generally, the figure shows a good spread over the regions while only 26.7% operate internationally.

Table (2.2) shows that 68.3% of the respondents have a bachelor's degree, 20.6% have a master's degree, 4.8% have an MBA and 6.3% have a Ph.D. The table also shows that the percentage of master's degree holders is highest in Group C (31.3%). It is interesting to note that more than 8% of the respondents in Groups A and B have a Ph.D. degree, and 8.33% in Group A and 6.25% in Group C have an MBA. Except for Ph.D's, Group C, 50 to 100 M dollars volume, seems to be more progressive with regard to the education background if it employees. In addition, Table (2.2) shows how experts at these companies developed their managerial skills. All respondents in Groups A and B and 86.7% in group C

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

TABLE (2.1).-- Company Information

1. TYPE OF COMPANIES				
FEATURE	GROUP (A) > 500 $M	GROUP (B) 100 to 500 $M	GROUP (C) 50 to 100 $M	COMBINED
TOTAL RESPONDENTS	(22)	(23)	(15)	(60)
1. COMMERCIAL	59.10%	78.26%	46.66%	63.33%
2. INDUSTRIAL	68.18%	52.17%	13.33%	48.33%
3. HIGHWAY & HEAVY CONST.	63.63%	39.13%	40.00%	48.33%
4. RESIDENTIAL	4.54%	17.39%	0.00%	8.33%
2. AVERAGE JOB SIZE				
1. < 1 MILLION	0.00%	8.69%	0.00%	3.33%
2. 1 TO 5 MILLIONS	9.10%	26.08%	20.00%	18.33%
3. 5 TO 10 MILLIONS	22.72%	26.08%	13.33%	21.66%
4. 10 TO 25 MILLIONS	22.72%	39.13%	40.00%	33.33%
5. 25 TO 100 MILLIONS	31.81%	13.04%	6.66%	18.33%
3. AVERAGE PROJECT DURATION				
1. < 1/2 YEAR	0.00%	4.34%	0.00%	1.66%
2. 1/2 TO 1 YEAR	9.10%	34.78%	40.00%	26.66%
3. 1 TO 2 YEARS	68.18%	47.82%	60.00%	43.33%
4. 2 TO 3 YEARS	18.18%	13.04%	0.00%	11.66%
5. > 3 YEARS	4.54%	0.00%	0.00%	5.70%

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

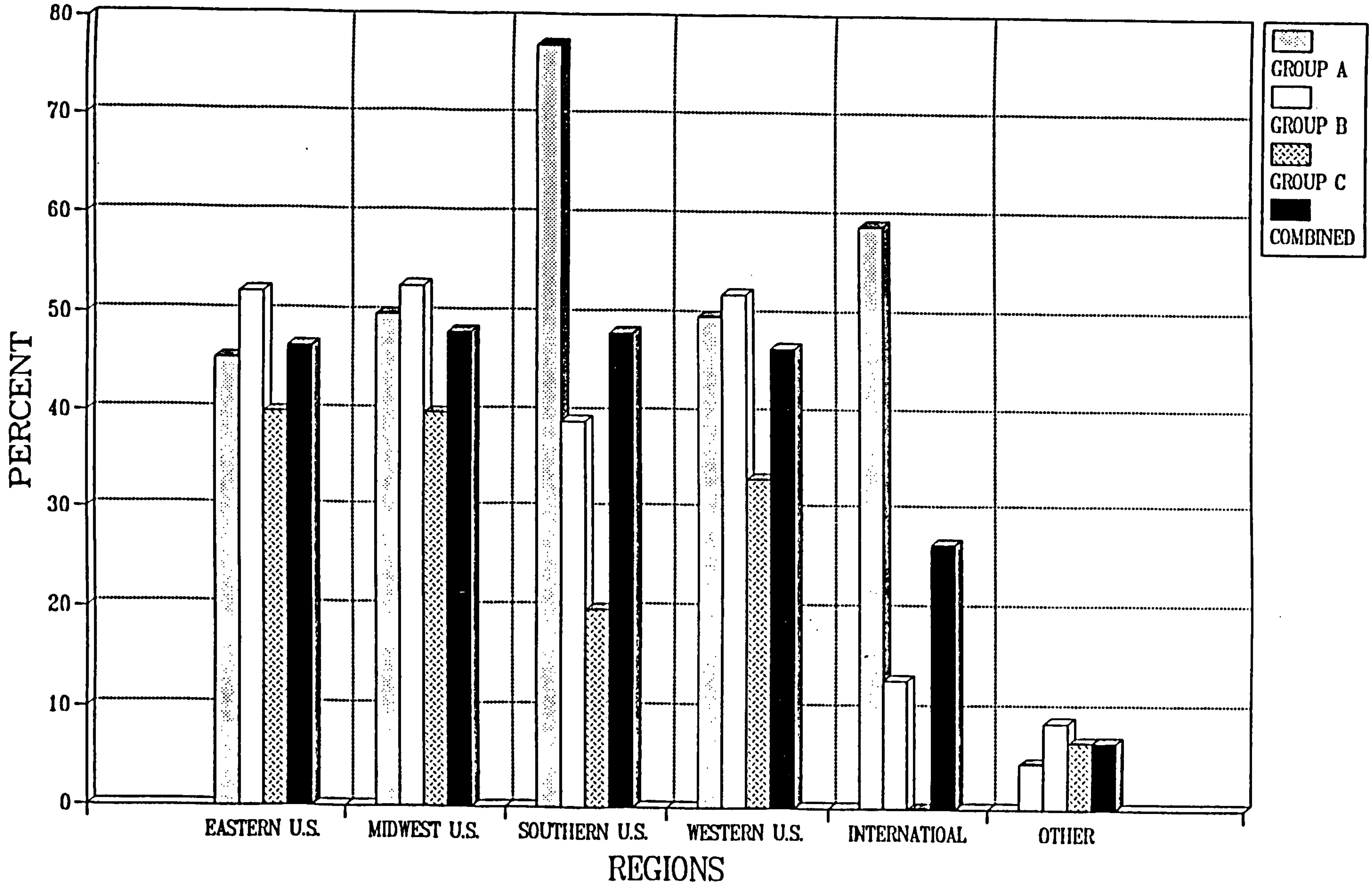


Figure (2.1).--Region Of Operation

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

TABLE (2.2).--Background of Respondents

FEATURE	GROUP (A)	GROUP (B)	GROUP (C)	COMBINED
1. EDUCATIONAL BACKGROUND				
1. BACHELOR	62.50%	78.26%	62.50%	68.25%
2. MASTER	20.08%	13.04%	31.25%	20.63%
3. MBA	8.33%	----	6.25%	4.76%
4. Ph.D.	8.33%	8.70%	----	6.35%
2. MANAGERIAL SKILLS				
1. EXPERIENCE	100.00%	100.00%	86.66%	96.66%
2. TECHNICAL LITERATURE	31.81%	30.43%	33.33%	31.66%
3. CONFERENCES	40.90%	52.17%	60.00%	50.00%
4. OTHER	9.10%	13.04%	13.33%	6.16%

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

stated that they have gained their managerial skills through experience. The lower percentage in group C does not make much sense. It may be due to a misunderstanding of the question by some of the respondents. The combined average of 96.7% is a better indicator. Therefore, it can be stated that contractors gain their managerial skills through experience rather than relying on technical literature and conferences.

2.5 HOW U.S. CONSTRUCTION COMPANIES PLAN AND CONTROL PROJECTS

Table (2.3) indicates that about 60% of the respondents have always been involved in the planning stage of their companies' projects. Their participation typically takes place in the prebidding and after-bidding/before-execution phases. The role of 48.3% of the respondents is to prepare a comprehensive plan.

In the context of planning practices, Table (2.3) shows that CPM is still the dominant planning technique, even though its utilization has dropped from 90% [12] to 79%[3]. The percent utilization is highest in groups A and B while it is only 53.3% in group C. This may be due to a lack of familiarity with CPM by contractors in group C, the impracticality of using CPM for planning and controlling small projects and its cost relative to the size of the project.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

TABLE (2.3).--Planning A Project

1. INVOLVEMENT IN PROJECT PLANNING				
FEATURE	GROUP (A)	GROUP (B)	GROUP (C)	COMBINED
1.NEVER	----	----	----	----
2.SOMETIMES	36.36%	52.17%	26.66%	40.00%
3.ALWAYS	63.64%	47.82%	73.33%	60.00%
2. AT WHAT STAGE DO YOU USUALLY BECOME INVOLVED?				
1.PREBIDDING	86.36%	91.30%	86.66%	88.33%
2.AFTER BIDDING & BEFORE EXECUTION	86.36%	86.95%	80.00%	85.00%
3.DURING EXECUTION	81.81%	78.26%	80.00%	80.00%
4.OTHER	----	21.74%	----	16.66%
3. WHAT IS YOUR ROLE ?				
1.BREAKDOWN THE PROJECT	27.27%	39.13%	13.33%	28.33%
2.PREPARE TECHNOLOGICAL LOGIC	31.81%	21.73%	26.66%	26.66%
3.PREPARE LIST OF RESOURCES	31.81%	13.04%	13.33%	20.00%
4.PREPARE A COMPREHENSIVE PLAN	59.09%	39.13%	46.66%	48.33%
5.OTHER	31.81%	52.17%	40.00%	41.66%
4. PLANNING METHODS				
1.BAR CHART	40.90%	73.91%	53.33%	56.66%
2.CPM	90.90%	82.60%	53.33%	78.33%
3.PERT	4.54%	4.34%	----	3.33%
4.COMPUTER PROGRAMS	72.72%	86.95%	46.66%	71.66%
5.OTHER	----	39.13%	6.66%	8.33%

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Figure (2.2) illustrates the criticism that CPM received from the respondents. As expected, group C is not comfortable with CPM (60%). The respondents in group C also stated that the method is laborious and time consuming (40%), expensive to update (27%) and impractical for field applications (34%). It is interesting to note that contractors in groups A and B are significantly more supportive of the use of CPM than those in group C. However, a good percentage in both groups stated that they are not comfortable with CPM, 27% for group A and 22% for group B and that the method is laborious, time consuming and not flexible.

PERT received a much lower level of acceptance than CPM. This is understandable from an analysis of Figure (2.3). All respondents are not comfortable with the method and believe that it is impractical for field application.

For project control, Table (2.4) again shows that CPM is the most practical method. Its rate of acceptance has increased from 52%[12] to 70%[3]. The Bar Chart is still used extensively by groups B and C at 60.9% and 66.7% respectively while it is used less, 36.4% by group A. Obviously this is directly related to the size of the projects. Most sophisticated methods of project control like learning curves and cost curves are used more by group A than the other groups. All three groups use statistical methods moderately and benefit to cost ratio rarely.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

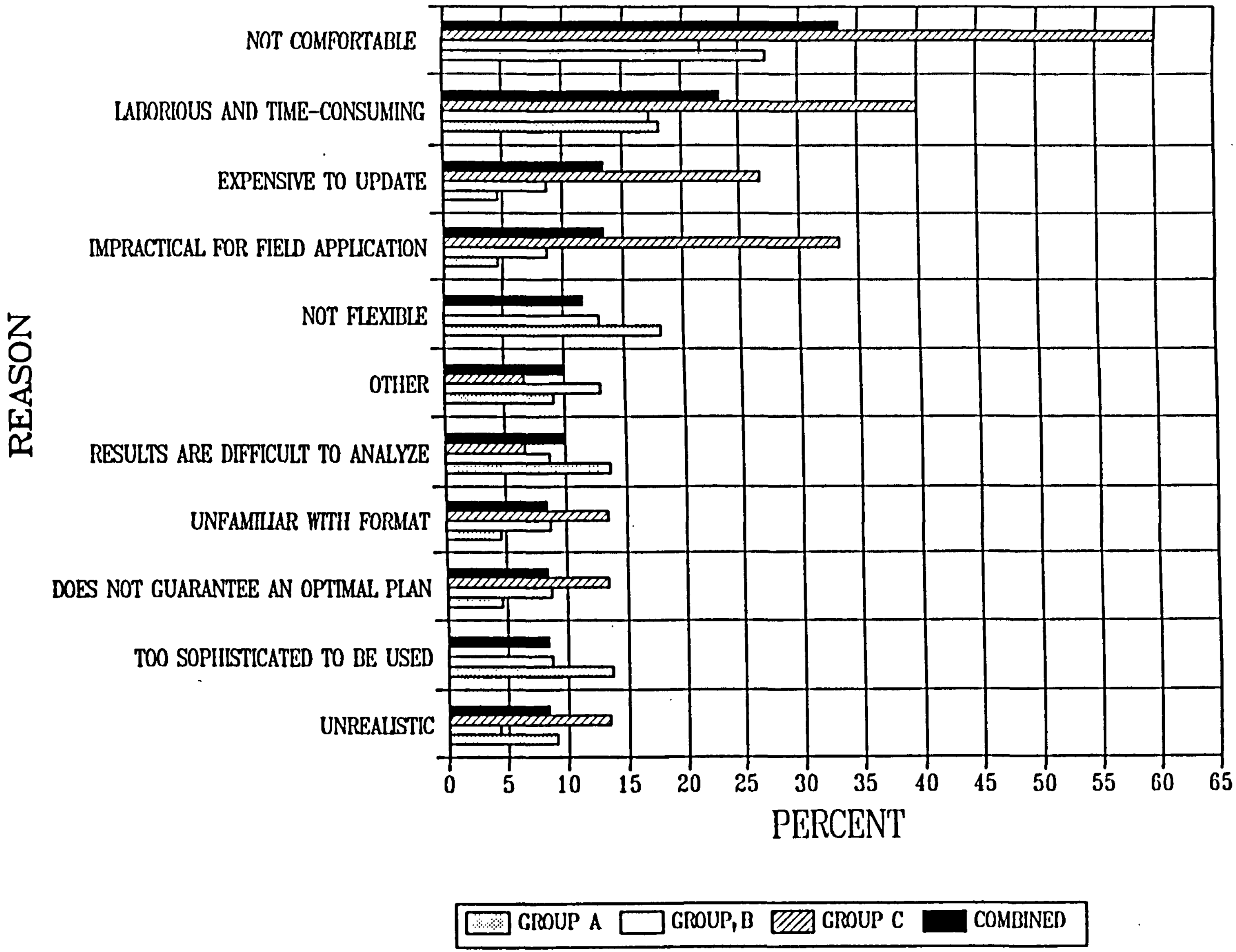


Figure (2.2).--Criticism Of CPM

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Although CPM is the most favored technique for planning and controlling construction projects, it is still criticized by 30% of the respondents, for one reason or another. When the respondents were asked to identify why they are not comfortable with CPM, they listed many reasons as shown in Table (2.5). The most significant ones are (in order of importance):

1. Laborious and time-consuming;
2. Expensive to update;
3. Impractical for field application;
4. Not flexible; and
5. Results are difficult to analyze.

These results confirm previous findings by other researchers[6,7,34,41].

Conventional CPM is expensive to run and to update[12,21,51]. CPM is also impractical for field application since it is not generally understood by subcontractors and the field staff[21,51]. The amount of effort and time required to prepare the initial plan properly is a major disadvantage[2,51]. These results and findings support the need for a more flexible, practical, and effective CPM. This need has been addressed in recent years by many researchers[7,12,21,41,51].

One of the goals of this research was to identify the effect of 11 factors on planning practices. Table (2.6) shows that the type and size of a project are ranked first and second, respectively. The availability of manpower that a planner usually

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

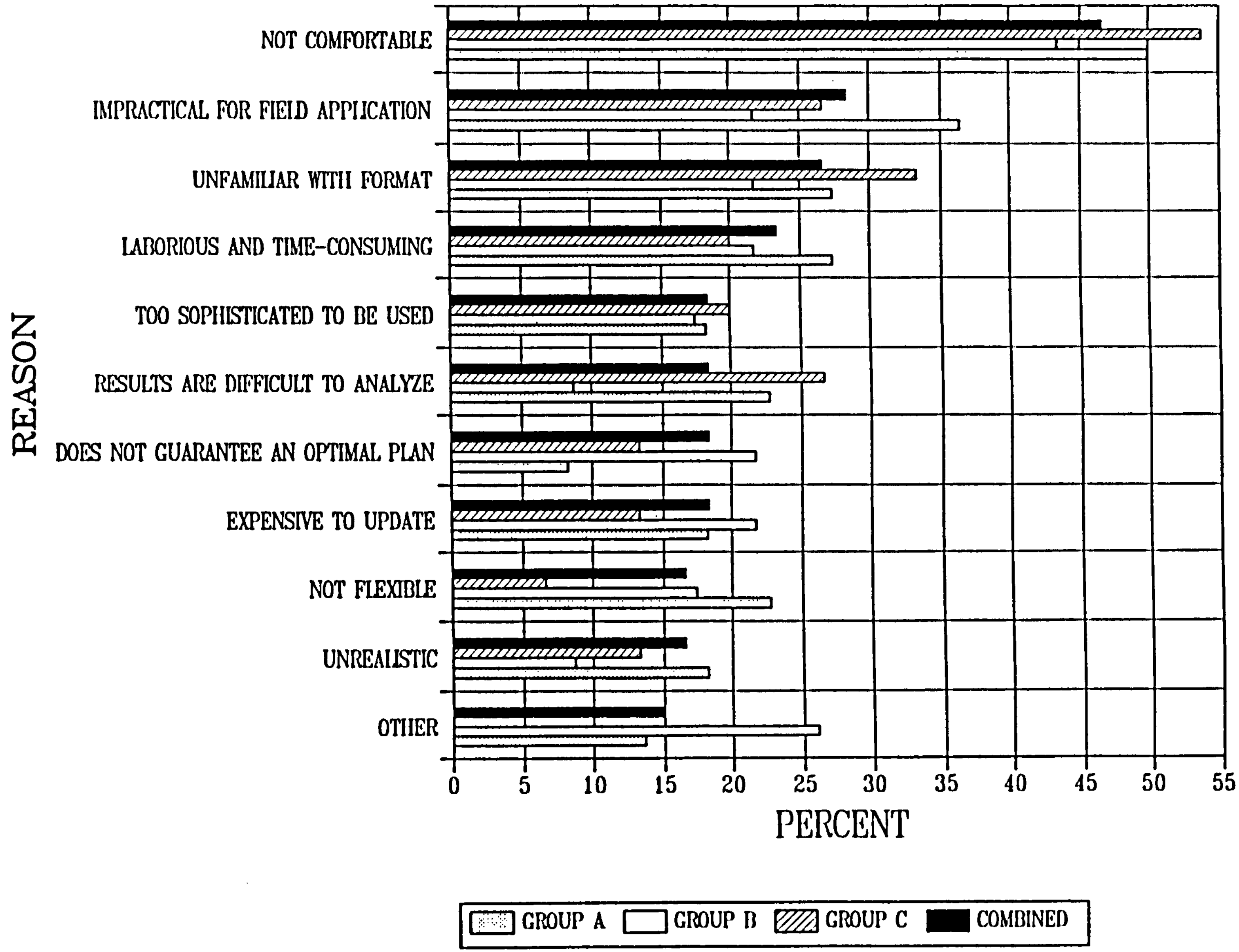


Figure (2.3).--Criticism Of PERT

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

TABLE (2.4).--Project Control

FEATURE	GROUP (A)	GROUP (B)	GROUP (C)	COMBINED
1. FREQUENCY OF SITE VISITS				
1. EVERY DAY	----	----	6.66%	1.66%
2. EVERY WEEK	18.18%	26.08%	20.00%	21.66%
3. BI-WEEKLY	22.72%	17.39%	20.00%	20.00%
4. EVERY MONTH	27.27%	34.78%	3.33%	31.66%
5. OTHER	31.81%	21.74%	6.66%	22.41%
2. METHODS OF CONTROL				
1. BAR CHART	36.36%	60.86%	66.66%	53.33%
2. CPM	72.72%	65.21%	73.33%	70.00%
3. PERT	4.54%	13.04%	----	6.66%
4. COMPUTER PROGRAMS	45.45%	78.26%	53.33%	60.00%
5. OTHER	13.63%	17.39%	----	11.66%
3. USE OF LEARNING CURVE				
1. YES	36.36%	----	15.38%	16.66%
2. SOMETIMES	22.72%	39.13%	26.66%	30.00%
3. NO	40.91%	61.00%	60.00%	53.33%
4. USE OF BENEFIT TO COST RATIO				
1. YES	9.10%	13.04%	6.66%	10.00%
2. SOMETIMES	36.36%	43.47%	33.33%	38.33%
3. NO	54.54%	43.47%	60.00%	51.66%
5. USE OF COST CURVE				
1. YES	68.18%	26.10%	6.66%	36.66%
2. SOMETIMES	27.27%	26.10%	33.33%	28.33%
3. NO	4.54%	47.82%	60.00%	35.00%
6. STATISTICAL METHODS				
1. YES	36.36%	39.13%	20.00%	33.33%
2. SOMETIMES	22.72%	21.74%	33.33%	25.00%
3. NO	40.90%	39.13%	46.66%	41.66%

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

TABLE (2.5).--Criticism Of CPM

REASON	GROUP (A) %	GROUP (B) %	GROUP (C) %	COMBINED %
1. Too sophisticated to be used	14.28	13.04	-----	10.00
2. Impractical for field application	9.53	13.04	20.00	13.33
3. Results are difficult to analyze	14.28	13.04	-----	10.00
4. Unfamiliar with format	9.53	8.70	13.33	10.00
5. Laborious and time-consuming	28.57	17.39	26.66	23.33
6. Expensive to update	19.04	8.70	13.33	13.33
7. Unrealistic	9.53	13.04	-----	8.33
8. Not flexible	19.04	13.04	6.66	13.33
9. Does not guarantee an optimal plan	14.28	8.69	-----	8.33
10.Other	9.53	13.04	6.66	10.00
• NOT COMFORTABLE WITH CPM	28.57	21.74	46.66	30.00

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

considers during the planning phase is ranked third. Another goal of this study was to determine the importance of some control criteria during the execution-control phase of a project. As expected, Table (2.7) shows that securing profit is the most important goal, and, surprisingly, productivity is the least important.

2.6 HUMAN RESOURCE MANAGEMENT

Respondents indicated that human resources should be planned during the prebidding phase (Table 2.8). This table also shows that the availability of skilled labor, foremen, and engineers are sometimes limited and that this period of limited resources lasts for about two weeks to one month. The table indicates that the majority of the respondents prefer to reallocate their human resources based on experience and working overtime.

Most of the respondents believe that their level of experience is sufficient to handle the problem of limited-resources (Table 2.8). This table also shows that 50% of the respondents prefer to use computer scheduling programs at times to plan and control their human resources. In addition, about 70% of the respondents have considered changing the original technological logic of the project when solving the limited-resource problem. 45% of the respondents who have used this approach have encountered "Good" success in their past experience.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

TABLE (2.6).--FACTORS AFFECT PLANNING PROCESS

(SCORE SCALE 1-5; 1 = LOW IMPORTANCE, 5 = HIGH IMPORTANCE)

RANK	FACTOR	Group (A)	Group (B)	Group (C)	Combined
1	SIZE OF PROJECT	4.23	4.52	3.45	4.066
2	TYPE OF PROJECT	4.15	4.52	3.45	4.040
3	AVAILABILITY OF MANPOWER	4.23	4.13	3.45	3.936
4	PERSONAL EXPERIENCE OF PLANNER	3.46	3.95	3.95	3.786
5	AVAILABILITY OF EQUIPMENT	3.31	3.56	4.36	3.743
6	AVAILABILITY OF MATERIAL	3.07	3.65	4.36	3.693
7	PLANNING METHODS AND TECHNIQUES	3.15	3.78	3.81	3.580
8	PROJECT DURATION	3.84	3.65	3.54	3.340
9	PROJECT LOCATION	3.23	3.69	3.27	3.39
10	AVAILABILITY OF FINANCE	2.92	3.56	3.54	3.340
11	COORDINATION WITH OTHER DEPARTMENTS	3.07	3.39	3.31	3.256

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

TABLE (2.7).--FACTORS AFFECT CONTROL PROCESS

(SCORE SCALE 1-5; 1 = LOW IMPORTANCE, 5 = HIGH IMPORTANCE)

RANK	CONTROL CRITERIA	GROUP (A)	GROUP (B)	GROUP (C)	COMBINED
1	SECURE PROFIT	4.772	4.869	4.923	4.854
2	OWNER'S SATISFACTION	4.727	4.695	4.769	4.730
3	MAINTAIN HIGH QUALITY	4.818	4.565	4.769	4.717
4	SAFETY	4.818	4.478	4.769	4.688
5	DURATION	4.818	4.565	4.384	4.589
6	AVOID CLAIM	4.409	4.434	4.307	4.383
7	IMPROVE PRODUCTIVITY	4.409	4.304	4.076	4.263

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

TABLE (2.8).--Human Resource Planning And Control

1. AT WHAT STAGE SHOULD RESOURCES BE SCHEDULED ?				
FEATURE	GROUP (A)	GROUP (B)	GROUP (C)	COMB.
1. PREBIDDING	86.36%	86.95%	73.33%	83.33%
2. AFTER BIDDING AND BEFORE EXECUTION	31.81%	47.82%	40.00%	40.00%
3. DURING PROJECT EXECUTION	13.63%	30.43%	20.00%	21.66
4. OTHER	----	4.34%	----	1.66%
2. HOW OFTEN DO YOU ENCOUNTER LIMITED RESOURCE SITUATION ?				
1. UNSKILLED				
* Rarely	68.20%	69.50%	60.00%	66.66%
* Sometimes	31.80%	30.50%	20.00%	28.33%
* Always	----	----	20.00%	5.00%
2. SKILLED				
* Rarely	9.120%	17.39%	23.10%	15.50%
* Sometimes	86.36%	78.26%	53.84%	75.86%
* Always	4.54%	4.34%	23.10%	8.60%
3. FORMEN				
* Rarely	9.10%	17.39%	15.38%	13.80%
* Sometimes	77.27%	69.56%	69.23%	72.41%
* Always	13.63%	13.04%	15.38%	13.79%
4. ENGINEERS				
* Rarely	22.72%	21.74%	38.46%	25.86%
* Sometimes	68.18%	73.91%	53.84%	67.24%
* Always	9.10%	4.34%	7.70%	6.89%
3. HOW LONG DOES THE LIMITED-RESOURCE SITUATION LAST ?				
1. < 1 WEEK	----	8.69%	6.66	5.00%
2. 1 TO 2 WEEKS	4.54%	13.04%	13.33%	10.00%
3. 2 WEEKS TO 1 MONTH	54.54%	43.47%	40.00%	46.66%
4. > 1 MONTH	40.90%	34.78%	40.00%	38.33%
4. HOW DO YOU USUALLY SOLVE IT ?				
1. REALLOCATION TECHNIQUES	13.63%	30.43%	6.66%	18.33%
2. EXPERIENCE	63.63%	78.26%	40.00%	63.33%
3. WORK OVERTIME	59.10%	60.86%	40.00%	55.00%
4. MORE SHIFTS	13.63%	21.73%	13.33%	16.66%
5. OTHER	27.27%	13.04%	6.66%	16.66%

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

In the final section of this questionnaire, respondents were asked whether a 3-D presentation of project networks would be helpful in planning and controlling their projects. This question did not receive a very positive response, perhaps because it suggests something new without adequate illustration.

2.7 COMMENTS BY THE RESPONDENTS

Some of the respondents' comments that help clarify the attitude of the construction industry toward CPM and other P&C techniques are listed below:

1. "CPM is not understood by all" (Group A);
2. "Project staff must participate in and be responsible for project planning. Scheduling techniques used must be flexible to meet the specific needs of each project's staff and client. Scheduling should never be done simply as a routine requirement" (Group B).
3. "Superintendents and tradesmen do not understand CPM" (Group B).
4. "Planning and control of a project must be capable of easily incorporating changes. This includes changes of scope, work sequence, durations, resource, etc." (Group A).
5. "Much of a project's success is a close coordination of design, engineering, materials control with project management and construction. We use CPM and Project Planning Systems to contribute to this objective" (Group C).

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

6. "Each project is required to establish performance goals (financial, safety, quality, etc.) prior to the actual work starting. These goals are reviewed and updated on a quarterly basis by the project management staff (PM, Superintendent, engineers, foremen, etc.)" (Group B).

7. "Computer applications should be as simple and graphic as possible in order to save the time of construction supervisors and managers who need to spend more time on managing their projects" (Group B).

8. "Our projects are short term in nature (3 months to 18 months). We use Primavera software for scheduling only. We have developed our own cost system for monitoring costs that is not tied directly into our project schedules" (Group C).

9. "Your questionnaire proposes a very technical approach to scheduling and cost/resource control. While this may be appropriate for large firms and larger projects, generally our approach is somewhat less technical. Once a master network is developed for a project schedule, most planning takes place utilizing short term (1-2 week) Bar schedules. Unfortunately, activities in this industry are much too dynamic generally to look much past the near future in planning manpower and scheduling subcontractors. With the best of intention, you try, but realistically you plan week to week" (Group B).

10. "Project scheduling is a product of designers, owner, or CM's instruction" (Group B).

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

TABLE (2.8).--Human Resource Planning And Control
contd.

1. DO YOU CONSIDER YOUR EXPERIENCE ENOUGH TO HANDLE THE LIMITED-RESOURCE PROBLEM ?				
FEATURE	GROUP (A)	GROUP (B)	GROUP (C)	COMBINED
1. YES	27.27%	26.10%	20.00%	25.00%
2. SOMETIMES	40.90%	39.10%	33.33%	38.33%
3. NO	31.81%	34.78%	46.66%	36.66%
2. WOULD YOU PREFER TO USE COMPUTER SCHEDULING SOFTWARE IN PLANNING AND CONTROLLING YOUR HUMAN RESOURCES ?				
1. YES	31.80%	21.73%	26.66%	26.66%
2. SOMETIMES	59.60%	47.82%	40.00%	50.00%
3. NO	9.10%	30.43%	33.33%	23.33%
3. DO YOU CONSIDER CHANGING THE ORIGINAL TECHNOLOGICAL LOGIC OF THE PROJECT, WHEN SOLVING THE LIMITED-RESOURCE PROBLEM ?				
1. YES	68.18%	69.56%	66.66%	68.33%
2. NO	31.82%	30.44%	33.33%	31.66%
4. IF YES, WHAT WAS THE RESULT?				
1. EXCELLENT	----	4.34%	6.66%	3.33%
2. GOOD	59.10%	34.78%	40.00%	45.00%
3. NORMAL	9.10%	30.43%	20.00%	20.00%
4. POOR	----	----	----	----

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

11. "We have anywhere from 5-10 projects at any one time that a computer generated schedule would be useful for. The problem is getting field personnel comfortable with using computers and keeping them updated" (Group C).

2.8 CONCLUSIONS

The focus of this research was to verify the methods and techniques that are used by the construction industry to plan and control projects. As expected, the results confirmed that CPM is the preferred method by contractors of all sizes. PERT is rarely used and the Bar Chart is still used for smaller projects. For project control other sophisticated methods such as Learning Curves and Cost Curves are mainly used by larger contractors, whereas Statistical methods are used by all. It is surprising that Benefit to Cost ratio has not gained acceptance among contractors; 38% of the respondents stated that they use it sometimes and 52% not at all.

Although CPM is the preferred method, most contractors indicated reservations about its use. It is clear that some improvements are needed to make the method more effective and simple to use, especially as a control tool. The findings revealed that there is a lack of understanding of all planning and control techniques that are available to the construction industry. The more sophisticated the method, the less frequently it is used.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Contractors prefer to use a short term (1-2 week) bar schedule which is not adequate for planning or controlling medium to large size projects.

Although construction companies are comfortable to a certain degree with the use of CPM, computer project management programs, and Bar charts, they have demonstrated dissatisfaction with the use of statistical methods, resource allocation techniques, and Learning Curves in controlling their projects. Consequently, they suggest that planning and control methods should be practical, flexible, integral and simple enough to be understood by the field staff.

Once a master plan and a schedule has been developed for a project, the individuals responsible for the implementation of these techniques should be knowledgeable enough to cope with the changes that occur in the field. They need to be able to reschedule and replan the remainder of the work to the best interest of the company.

While planning projects, construction experts regard several factors as prerequisite elements for project success. They assume that successful planning practices can be guaranteed if manpower is carefully planned and the personal experience of the planner is effectively utilized during the planning phase. When dealing with the execution and control of projects, experts prefer to establish

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

control criteria to guide the project management team. They also suggest that these criteria should be reviewed and adjusted to accommodate any changes during the execution.

It becomes evident from the survey data that construction companies have confidence in experience as a principal means in handling P&C issues. These companies believe that a key component for successful project planning is the actual participation of the project management team and the close coordination among this team. Experts from these companies express the need for methods that are practical. Thus, it is fair to say that experience should be integrated in P&C methods.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

CHAPTER THREE

COMPUTERIZED MULTI-DIMENSIONAL SUPPORT SYSTEM (CM_DSS)

FOR

PLANNING, SCHEDULING, AND CONTROLLING CONSTRUCTION PROJECTS

3.1 INTRODUCTION

The problem of existing planning-scheduling-controlling schemes lies in their inability to provide the planner or the project manager with several scheduling alternatives, to select a target schedule, to provide a comprehensive network analysis, and to incorporate criteria for controlling the project. Consequently, a computerized multi-dimensional support system (CM-DSS) for planning, scheduling, and controlling construction projects is proposed to address these concerns.

Designing and developing the CM-DSS algorithm comprises the second major part of this research. In the following sections, the steps followed in developing the CM-DSS algorithm will be presented and discussed.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

3.2 DESIGN OVERVIEW

One of the basic approaches to problem solving is to break a complex problem into subproblems. This helps in solving the problem one step at a time rather than all at once. This approach is often called divide and conquer[15]. It is used in designing the CM-DSS algorithm by dividing the solution into three major steps:

Step 1. Planning.

Step 2. Scheduling.

Step 3. Control.

At each step, part of the solution is achieved. To do this, however, each step is divided into substeps and the fundamental aspects of planning, scheduling, and controlling a project are employed. After solving the problem, an algorithm, which is a list of steps that organizes the solution for the original problem, is established.

3.3 CM-DSS ALGORITHM

Generally, the CM-DSS algorithm consists of three major steps as shown in figure (3.1). These steps are:

Step 1. Planning--Plan Activities.

Step 2. Scheduling--Develop Scheduling Alternatives.

Step 3. Control--Establish Control Plan.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

STEP 1. PLANNING--PLAN ACTIVITIES.

As previously stated in section (1.11), the objective of the planning process is to plan the project under study by (a) utilizing fundamental activity information such as productivity rates, crew size, time, and cost, and (b) considering the impact of the learning curve on the repetitive activities. In the following sections, the planning process is explained and discussed.

__a. Establish Productivity-Cost Database.__ A format similar to what is shown in Exhibit (3.1), page 80, is used to establish a productivity-cost database. When studying a project, it is normal practice to examine different crew sizes in order to increase productivity and/or to satisfy a limited-resource situation. Therefore, four possible crew sizes are considered for each activity to reflect this practice. Column #2 lists the man-hours required to execute activity #1, based on an optimal combination of productivity rate and crew size. Man-hours for other crew sizes are listed in column #3. These values are usually determined by the project manager based on past experience or historical records. Exhibit (3.1) shows that activity # 1 has upper and lower limits of crew size (column #4) for its execution. Typically, each crew size corresponds to a different productivity rate (column #3), which reaches an optimal value at a specific crew size. This variability mainly depends on the type of work to be performed and the method of execution. It is not within the scope of

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

this research to study the relationship between the productivity rate and each crew size, but nevertheless, it is important to emphasize this issue and to design the database based on this variability. Column #5 shows the Adjusted Man-Hour values. These values are obtained by dividing the optimal man-hours by the corresponding productivity. The unit cost for one unit of resource per hour is listed in column #6. For simplicity, this value is considered constant even though it may vary from one type of resource to another and from one activity to another. Column #7 shows the adjusted-hours, for each crew size, which is calculated by dividing the required man-hours by the product of productivity rate and crew size. The total cost required to perform activity #1 equals the product of the Adjusted Man-Hours (AMH), and the unit cost. This is shown in column #8.

EXHIBIT (3.1).--PRODUCTIVITY COST DATABASE

(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
AN	**MH**	**PR**	**CS**	**AMH**	**UC($)**	**AH**	**TC($)**
1		0.79	3	50.63	10	16.87	506.30
		0.88	4	45.45	10	11.36	454.50
	40	1.00	5	40.00	10	8.00	400.00
		0.92	6	43.48	10	7.25	434.80

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

LEGEND:

AN = Activity Number

MH = Man-Hours

PR = Productivity Rate

CS = Crew Size = number of different resources in a crew

AMH = Adjusted Man-Hours = [MH]/[PR]

UC = Unit Cost per Man-Hours

AH = Adjusted Hours = [AMH]/[CS]

TC = Total Cost = [UC] x [AMH]

NOTE: CM-DSS assumes that each activity has four sets. Each set consists of six parameters (i.e.,PR, CS, AMH, UC, AH, and TC).

---

b- Establish Interdependencies. The goal of this step is to establish the Logic to be followed in performing the project at hand. This is a difficult and time-consuming task. Usually, the planner and the project manager meet to study the various methods, scenarios, and alternatives that can be used to feasibly execute the project. Then they use one of the CPM notations the Arrow Diagram Method (ADM) or the Precedence Diagram Method (PDM) to establish the final logic. In this research, the former one is used.

c - Determine Activity Basic Conditions. This step requires a thorough understanding of the contractual and physical

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

conditions that a project may encounter. According to the scope of this research, only three constraints are considered (resource, time, and cost).

d- Use the Learning Curve for Repetitive Activities. The basic idea of the learning curve is that productivity improves when performing repetitive tasks. For instance, the casting of the fifth unit of five identical footings should take less time than the first one. It is realistic to include the learning curve effect when estimating the required man-hours for repetitive activities instead of assuming equal productivity. CM-DSS assumes that one of three learning rates can be used (75%, 80%, or 85%), and each group of repetitive activities has a maximum of 4 repetitive units. As a result, CM-DSS moves on and calculates the new value of man-hours for each activity based on its groups's learning rate, and the order in which this unit will be implemented. This new value of the man-hours will be used instead of the value available in the database in determining the adjusted hours for this activity. The following exhibit presents the learning curve algorithm.

EXHIBIT (3.2).--THE LEARNING CURVE ALGORITHM

Step 0: Determine the number of activities (ACTIVITIES).

Step 1: Initialize the process. Put N := 0.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Step 2: Start with activity N := N+1. Decide whether activity N is repetitive or not. If yes, then go to Step 2. Otherwise, repeat Step 2.

Step 3: Determine the learning rate (75%, 80%, or 85%).

Step 4: Determine the order in which this activity will be implemented.

Step 5: Use the learning curve equation (1.15) or the learning rate chart in Appendix (III) to modify the original man-hours available in the database.

Step 6: Store the modified man-hours in the database.

Step 7: If N < ACTIVITIES, then repeat steps 2 to 7. Otherwise, STOP.

STEP 2. SCHEDULING--DEVELOP SCHEDULING ALTERNATIVES.

In this step, the objective is to select a target schedule as a final schedule for project execution. This can be accomplished by studying several schedules under different scenarios and constraints. This is usually done only if the project is critical to the contractor or if the project manager has enough time to conduct such a study. This study involves, at best, the development of two alternatives and focuses on finishing the project on time. However, comparing two scheduling alternatives based on time only does not provide the project manager with a realistic schedule.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Therefore, the intention of this research is to stress the significance of examining more than one schedule under different scenarios, when planning construction projects, and to consider practical control criteria in the selection of a target schedule. CM-DSS develops the following four scheduling alternatives:

<u>a- Cost-Based Schedule</u>. It is sometimes important to control the cost of the project regardless of other factors such as resource and time. This can be done by developing a cost-based schedule. The Cost-Based Schedule (CBS) is based on the minimum cost of each activity. CM-DSS assigns the least cost, which corresponds to the optimal productivity rate, to each activity, and then uses the corresponding adjusted hours to perform CPM computations. The CBS algorithm is shown in Exhibit (3.3). It should be noted that this schedule is what has been defined as "normal" in section (1.6).

In this algorithm, the planner starts by using the four sets of Man-Hours, Productivity Rate, Crew Size and Unit Cost available in the productivity-cost database established in Step (1.a) to calculate the Adjusted-Hours and Total Cost for each activity. Next, the planner selects the minimum Total Cost and determines the corresponding Adjusted-Hours and Crew Size. Finally, he/she uses the Adjusted-Hours values to run a CPM analysis, and uses the corresponding Total Cost values to calculate the Total Cost for this schedule.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

EXHIBIT (3.3).--THE COST-BASED SCHEDULE ALGORITHM

Step 0 : For each activity, get the four sets of Man-Hours, Productivity Rate, Crew Size, and Unit Cost.

Step 1 : Initialize the process: put N := 0

Step 2 : For activity (N := N + 1), calculate the four Adjusted-Hours and Total Cost values and store these values.

Step 3 : Search the four Total-Cost values and determine the minimum (least) value.

Step 4 : Determine the Adjusted-Hour and Crew Size values that correspond to the least cost value.

Step 5 : Repeat Steps 2 & 3 for all activities.

Step 6 : Use the Adjusted-Hour values to run CPM computations and use the corresponding Total cost values to calculate the Total Cost for this schedule.

---

b- Time-Based Schedule. It is sometimes important to control the time of the project regardless of other factors such as resource and cost. This can be done by developing a time-based schedule. The Time-Based Schedule (TBS) is based on the minimum adjusted hours of each activity. CM-DSS assigns the

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

least adjusted hours to each activity, and then performs CPM computations. Exhibit (3.4) shows the TBS algorithm.

In this algorithm, the planner starts by using the four sets of Man-Hours, Productivity Rate, Crew Size and Unit Cost available in the productivity-cost database established in Step (1.a) to calculate the Adjusted-Hours and Total Cost for each activity. Next, the planner selects the minimum Adjusted-Hours and determines the corresponding Total Cost and Crew Size. Finally, he/she uses the Adjusted-Hours values to run a CPM analysis, and uses the corresponding Total Cost values to calculate the Total Cost for this schedule.

EXHIBIT (3.4).--THE TIME-BASED SCHEDULE ALGORITHM

Step 0 : For each activity, get the four sets of Man-Hours, Productivity Rate, Crew Size, and Unit Cost.

Step 1 : Initialize the process: put N := 0.

Step 2 : For activity (N := N + 1), calculate the Adjusted-Hours and Total Cost values and store these values.

Step 3 : Search the four Adjusted-Hour values and determine the minimum (least) value.

Step 4 : Determine the Total Cost and Crew Size values that correspond to the least time value.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Step 5 : Repeat Steps 2 & 3 for all activities.

Step 6 : Use the Adjusted-Hour values to run CPM computations and use the corresponding Total cost values to calculate the Total Cost for this schedule.

<u>c- Resource-Based Schedule</u>. If the availability of resources is limited, the planner may decide to schedule the project by using a resource-based schedule regardless of factors such as time and cost. The Resource-Based Schedule (RBS) is based on the minimum crew size that can perform each activity. CM-DSS assigns the adjusted hours, which corresponds to the least resource value, to each activity, and then performs CPM computations. The RBS algorithm is shown in Exhibit (3.5).

To develop the Resource-Based Schedule, the planner uses the productivity-cost database established in Step (1.a). For each activity, the planner uses the sets of Man-Hours, Productivity Rate, Crew Size, and Unit Cost to calculate the Adjusted-Hours and Total Cost values. Next, the planner selects the minimum Crew Size and determines the corresponding Adjusted-Hours and Total Cost values. Finally, he/she uses the Adjusted-Hours values to run CPM analysis and the corresponding Total Cost values to calculate the Total Cost for this schedule.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

EXHIBIT (3.5).--THE RESOURCE-BASED SCHEDULE ALGORITHM

Step 0 : For each activity, get the four sets of Man-Hours, productivity Rate, Crew Size, and Unit Cost.

Step 1 : Initialize the process: put N := 0.

Step 2 : For activity (N := N + 1), calculate the four Adjusted-Hours and Total Cost values and store these values.

Step 3 : Search the four Crew Size values and determine the value that corresponds to the least resource value.

Step 4 : Determine the corresponding Adjusted Hour and Total Cost values.

Step 5 : Repeat Steps 2 & 3 for all activities.

Step 6 : Use the Adjusted-Hour values to run CPM computations and use the corresponding Total cost values to calculate the Total Cost for this schedule.

---

d- Feasible Schedule. Each activity in the network could be subject to a different constraint (resource, time, or cost). The Feasible Schedule (FS) takes this variability into consideration when calculating the durations of the activities. CM-DSS imposes the relevant constraint on each

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

activity separately and assigns the adjusted hours to this activity. The adjusted hours are then used to perform CPM computations. Exhibit (3.6) demonstrates the FS algorithm.

In this algorithm, the planner studies each activity and determines a possible constraint (crew size, time, or cost). For example, If an activity has a cost constraint, in this case, the planner searches the Total Cost values available in the productivity-cost database and uses the Adjusted-Hour value that correspond to the cost constraint to run CPM analysis. In case of a time/resource constraint, a similar approach can be used to run CPM analysis. If there is no constraint, the planner uses the values that correspond to the optimal productivity rate to run CPM analysis. This scheduling alternative allows the planner to study each activity under different scenarios and constraints.

EXHIBIT (3.6).--THE FEASIBLE SCHEDULE ALGORITHM

Step 0 : Initialize the process: put N := 0

Step 1 : For activity (N := N + 1), if there is a set of constraint (Crew Size, Time, or Cost), then use its value. Otherwise, go to step 2.

Step 2 : For activity (N := N + 1), search the four values of the Total-Cost and determine the least value.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Step 3 : Determine the Adjusted-Hour and Crew Size values that correspond to the least cost value.

Step 4 : Repeat Steps 1 & 2 for all activities.

Step 5 : Use the Adjusted-Hour values to run CPM computations and use the corresponding Total Cost values to calculate the Total Cost for this schedule.

STEP 3. CONTROL--ESTABLISH CONTROL PLAN.

As previously discussed in Chapter #1, an effective project control system should be capable of analyzing the criticality of scheduling alternatives and their activities, incorporating control criteria and using a decision support system that can be used in selecting a target schedule, and forecasting the potential of each activity to deviate from the target schedule. These issues are considered in the design of CM-DSS. The following four steps are required to achieve effective project control.

a - Analyze the scheduling alternatives. Conventional CPM only allows for limited network analysis. It provides the user with the project duration and four floats for each activity. As a result, CPM has not been fully utilized by project managers for this purpose. To further its use for effective control of construction projects, more information should be incorporated in the analysis. Accordingly, 10 new indices were developed in

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

this study to achieve this objective. The indices are called criticality indices and are divided into three groups: Activity, Schedule, and Network. CM-DSS uses these indices to measure the criticality of each schedule and activity in terms of time, cost, and resource. Then, it analyzes the project technological logic and provides the planner with a logic index that reflects the criticality of the project network. If the planner is not satisfied with the logic index value and wishes to change the logic of the project, he/she can use the three groups of indices to focus on the activities that should be replanned. The analysis of the scheduling alternatives starts by calculating the following indices:

<u>Activity Criticality Indices</u>

<u>1. Activity Criticality Index (ACI).</u> CPM assumes that the activities of any network are either critical or noncritical. This is not always sufficient because the noncritical activities also have a certain degree of criticality based on their duration and total float. For instance, assume that activity A has a duration of 7 days and a total float of 3 days, and activity B has a duration of 6 days and a total float of 3 days--see exhibit (3.7). CPM does not distinguish between them, in terms of criticality, and assumes that each activity is noncritical with a total float value of 3 days. The following equation is formulated to determine which activity is more critical and consequently has a higher tendency to become critical.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

$$\textit{Activity Criticality Index (ACI)} = \left[1 - \frac{TF}{\textit{ActivityDuration} + TF} \right] x \ 100\% \qquad [3.1]$$

where: 0<= ACI <= 100%

By using this equation, activity #1 has an ACI value of 70%, while activity #2 has an ACI value of 66.66%, thus making activity #1 more critical This demonstrates that, for two activities with the same total float and two different durations, the activity that has a longer duration is more critical. This index reflects the criticality of an activity in terms of time. It provides the project manager with a criterion that can be used to closely monitor noncritical activities with, say an ACI of 75% or higher to prevent them from becoming critical.

EXHIBIT (3.7).--ACTIVITY CRITICALITY INDEX

```
            Activity #1

|<---------------------------->|<------------>|   ACI = 70%
     Duration = 7 days              TF = 3 days

            Activity #2

|<------------------------->|<------------>|      ACI = 66.66%
     Duration = 6 days              TF = 3 days
```

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

2. Activity Cost Index (ACOI). This is a simple way of measuring the criticality of an activity in terms of cost. Equation (3.2) represents the relative weight of the cost of an activity with respect to the project total cost.

$$\text{Activity Cost Index (ACOI)} = \left[\frac{\text{Activity Cost}}{\text{Project Total Cost}} \right] x\ 100\% \qquad [3.2]$$

Assume that the activity information shown in exhibit (3.8) belongs to project XYZ. Activities #1, #2, #3, #4, #5, #6, and #7 have ACOI values of 18%, 10%, 20%, 24%, 8%, 11%, and 9%, respectively. This indicates that activity #4 is the most critical activity in terms of cost and therefore should be closely monitored. This index assists the project manager in focusing on the activities that have high indices and accordingly establishing an appropriate cost control policy. The planner may also use this index to reschedule high-cost activities.

EXHIBIT (3.8).--ACTIVITY COST INDEX

Activity	ACOI
#1	18%
#2	10%
#3	20%
#4	24%
#5	8%
#6	11%
#7	9%
TOTAL	100%

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

3. Activity Resource Index (ARI). This index reflects the relative weight of each activity in terms of labor resource requirements. The following equation is used:

$$\textit{Activity Resource Index (ARI)} = \left[\frac{\textit{Activity Man-Hours}}{\textit{Project Man-Hours}} \right] x\ 100\% \qquad [3.3]$$

Assume that the activity information shown in exhibit (3.9) belongs to project XYZ. Activities #1, #2, #3, #4, #5, #6, and #7 have ACOI values of 15%, 13%, 16%, 19%, 6%, 22%, and 9%, respectively. This indicates that activity #6 is the most critical activity in terms of resource requirements. This index provides a project manager with a measure that can pinpoint activities that consume the highest percentage of a critical resource. If the planner decides to change the project network, he/she can fucos on the activities that have high ARI and replan them. A separate ARI may be calculated for each critical resource.

EXHIBIT (3.9).--ACTIVITY RESOURCE INDEX

Activity	ARI
#1	15%
#2	13%
#3	16%
#4	19%
#5	6%
#6	22%
#7	9%
TOTAL	100%

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

4. Activity Interfering Index (AINTI). The Interfering Float may be used to develop another criticality index for noncritical activities that are on the same chain. This index focuses on individual paths and should be considered within that context. The equation for calculating the Activity Interfering Index (AINTFI) is as follows:

$$\textit{Activity Interfering Index (AINTFI)} = \left[\frac{TF - FF}{Duration + TF - FF} \right] x\ 100\% \qquad [3.4]$$

Assume that the information shown in Exhibit (3.10) belongs to three noncritical activities on one path. In order to distinguish between the three activities in terms of criticality, CM-DSS employs this index. As shown in Exhibit (3.10), the three activities have different degrees of criticality even though they have the same total float. By using equation (3.4), a planner can easily determine the degree of criticality of each activity in the path. Exhibit (3.10) shows that activity #1 has an AINTI value of 50%, activity #2 has an AINTI value of 42.86%, and activity #3 0%. This indicates that activity #1 is the most critical activity among the three in terms of this index and would have a higher impact on the project if it is delayed.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

EXHIBIT (3.10).--ACTIVITY INTERFERING INDEX

Activity	Dur.(hours)	TF	FF	INTF	AINTI
#1	3	3	0	3	50.00%
#2	4	3	0	3	42.86%
#3	7	3	3	0	0.00%

```
 Act. 1    Act. 2     Act. 3        TF = 3
|-------->|--------->|----------->|<-------->|

|<---------------------------------------->|
              A noncritical path
```

<u>5. Activity Independent Index (AINDI).</u> The Independent Float may also be used to develop another criticality index for noncritical activities that are on the same path. This index focuses on individual paths and should be considered within that context. The equation for the Activity Independent Index (AINDI) is as follows:

$$\text{Activity Independent Index (AINDI)} = \left[1 - \frac{INDF}{Duration + FF} \right] x\ 100\% \qquad [3.5]$$

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Assume that the information shown in Exhibit (3.11) is for two noncritical activities on one path. Exhibit (3.11) shows that, for two activities with the same Free Float and two different INDF values, the activity that has a smaller INDF is more critical. This means that activity #1 is more sensitive than activity #2 to any slippage in the completion of an activity that precedes it. The project manager can use this index to concentrate on the noncritical activities that have high AINDI to prevent them from becoming critical.

EXHIBIT (3.11).--ACTIVITY INDEPENDENT INDEX

Activity	Dur.(hours)	FF	INDF	AINDI
#1	4	3	2	71.42%
#2	6	3	3	66.665

6. <u>Activity Global Index (AGI).</u> This index reflects the overall criticality of an activity assuming that all indices have the same weight. The AGI equation is as follows:

$$\text{Activity Global Index} = \left[\frac{ACI + ACOI + ARI + AINTI + AINDI}{5} \right] x \ 100\% \qquad [3.6]$$

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Assume that the activity information shown in exhibit (3.12) belongs to project XYZ. This information indicates that activity #3 has the highest AGI value and thus it is the most critical activity in terms of time, cost, and resource. This index is a practical measure that can be used if the planner decides to change the network logic. He/she can employe AGI to replan those activities that have high AGI values.

EXHIBIT (3.12).--ACTIVITY GLOBAL INDEX

ACTIVITY	ACI	ACOI	ARI	AINTI	AINDI	AGI
#1	60%	20%	23%	70%	65%	48.60%
#2	45%	12%	11%	40%	35%	35.60%
#3	80%	15%	14%	17%	50%	34.00%
#4	12%	9%	10%	80%	30%	38.00%
#5	7%	5%	4%	10%	100%	25.20%

Schedule Criticality Indices

1. Schedule Time Index (STI). This index serves as an overall measure of criticality for each scheduling alternative in terms of total float and duration. The STI equation is as follows:

$$Schedule\ Time\ Index\ (STI) = [\ \frac{\sum ACI}{N}\ ] \qquad [3.7]$$

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Where:

$0 <= STI < 100\%$

ACI = Activity Criticality Index

N = number of activities

An STI value of 0.0% means that all activities in the scheduling alternative under study are critical and have 0 TF, and an STI value of 50% indicates a reasonable degree of criticality. An STI value should be always less than 100% since CPM assumes that some activities in the network should be critical, i.e. have zero TF. Exhibit (3.13) shows the STI values of the four scheduling alternatives (CBS, TBS, RBS, and FS) for project XYZ. These values indicate that the TBS alternative is the most critical schedule in terms of total float and duration. This means that most of TBS activities have 0 TF. The planner can use this index as a control criterion for the selection of a target schedule.

EXHIBIT (3.13).--SCHEDULE CRITICALITY INDEX

Alternative	CBS	TBS	RBS	FS
STI	60%	87%	55%	63%

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

2. Schedule Cost Index (SCOI). During project execution, contractors want to control the cost of their projects. This index serves as an overall measure of criticality for each scheduling alternative in terms of cost. It provides the planner with the average cost per scheduling alternative. The SCOI equation is as follows:

$$\text{Schedule Cost Index (SCI)} = \left[\frac{\sum \text{Activity Cost}}{N} \right] \quad [3.8]$$

Where:

N = number of activities

If the planner is not satisfied with the SCOI value, he/she can change the network logic and replan some of the activities that have high ACOI values. Exhibit (3.14) shows the SCOI values of the four scheduling alternatives (CBS, TBS, RBS, and FS) for project XYZ. These values indicate that the RBS alternative is the most expensive and therefore is the most critical in terms of cost. The planner can use this index as a control criterion for the selection of a target schedule.

EXHIBIT (3.14).--SCHEDULE COST INDEX

Alternative	CBS	TBS	RBS	FS
SCOI	$32,250	$31,190	$38,280	$35,900

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

3. Schedule Resource Index (SRI). This index serves as another overall measure of criticality for each scheduling alternative in terms of labor resource requirements. It provides the planner with the average man-hours per scheduling alternative. The SRI equation is as follows:

$$\textit{Schedule Resource Index (SRI)} = \left[\frac{\sum \textit{Man-Hours}}{N} \right] \qquad [3.9]$$

Where:

N = number of activities

If the planner is not satisfied with the SRI value, he/she can change the network logic and replan some of the activities that have high ARI values. This index can be also used as a control criterion for the selection of a target schedule. Exhibit (3.15) shows the SRI values of the four scheduling alternatives (CBS, TBS, RBS, and FS) for project XYZ. These values indicate that the FS alternative with 1020 Man-Hours, is the most critical schedule in terms of labor resource requirements.

EXHIBIT (3.15).--SCHEDULE COST INDEX

Alternative	CBS	TBS	RBS	FS
SRI (Man-Hours)	896	876	910	1020

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Network Criticality Index

1. Network Logic Index (NLI). This index measures the criticality of a network based on the comparison of its duration to the duration of the project if its activities were to be performed in series--see Exhibit (3.16). This is explained by the following equation:

$$\text{Network Logic Index (NLI)} = [\, 1 - \frac{\text{CPM Duration}}{\sum \text{Activity Durations}} \,] \qquad [3.10]$$

where: 0 <= NLI <= 100%

This equation indicates that the shorter the CPM duration, the more critical the network. An NLI value of 0.0% means that the CPM duration is equal to the duration in series and, accordingly, the logic of this network is not critical. As soon as, the logic is established and a parallel network is designed, the project will have a duration that is based on the complexity (density) of the logic. During the planning phase, a project manager may elect to use this index to change the logic of the network to reduce its complexity. This index also may be used as an indicator of the required frequency for reviewing this network. A high NLI (for instance 75-85%) means that more frequent updates are required to control the project, while a low (40-50%) NLI indicates that the project is more simple and might be monitored on a less frequent

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

basis. This index can also be used to indicate how careful the planner has been in preparing the project network. For instance, a low NLI (less than 50%) means that the planner has established a simple network logic.

EXHIBIT (3.16).--NETWORK LOGIC INDEX (NLI)

Activities in parallel for project XYZ

```
   #1        #2          #3           #4        #5
|----->|-------->|---------->|----->|------------->|
|<------------------------------------------------>|
          Total Activity Duration
```

Activities in series for project XYZ

```
  #1
|----->
   #2
 |------->
         #3
      |---------->
      #4         #5
   |----->|------------->|
|----------------------->|
     CPM Duration
```

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

The planner or the project manager may elect to use the Schedule Criticality Index and the Network Logic Index (and/or any other index) as control criteria to help in the selection of a target schedule.

Exhibit (3.17) summarizes the network analysis algorithm. In this algorithm, the planner starts by calculating the ACI, ACOI, ARI, AINTI, AINDI, AGI, STI, SCOI, SRI, and NLI. Next, the planner determines an acceptable upper limit value of NLI, for the project under study, based on past experience. If the NLI value previously calculated is less than that limit, this means that the network logic is acceptable, i.e., it is not complex. On the other hand, if the NLI value previously calculated is higher than that limit, this may indicate that the network logic is complex. Accordingly, the planner should replan the project by establishing a new technological logic and/or modifying the original database established in Step (1.a).

EXHIBIT (3.17).--NETWORK ANALYSIS ALGORITHM

Step 0 : For each activity, determine the ACI value.

Step 1 : For each activity, determine the ACOI value.

Step 2 : For each activity, determine the ARI value.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Step 3 : For each activity, determine the AINTI value.

Step 4 : For each activity, determine the AINDI value.

Step 5 : For each activity, determine the AGI value.

Step 6 : For each schedule, determine the STI value.

Step 7 : For each schedule, determine the SCOI value.

Step 8 : For each schedule, determine the SRI value.

Step 9 : For each network, determine the NLI value.

Step 10: Determine an acceptable upper limit value of NLI for the project under study. If the NLI value calculated in Step 9 is less than that limit, then go to step 14. Otherwise, go to Step 11.

Step 11: Replan the project by establishing a new technological logic and/or modifying the original database.

Step 12: Use the activity indices (i.e., ACI, ACOI, ARI, AINTI, AINDI, and AGI) to concentrate on the activities that have high AGI values, and replan them.

Step 13: Repeat all the previous algorithms for the new logic.

Step 14: STOP.

---

b- Determine control criteria. After analyzing each schedule (based on the final logic), it is important to determine the control criteria that will be used in selecting a target

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

schedule. CM-DSS, for example, suggests the following control criteria: a) TOTAL COST of each schedule, b) DURATION of each schedule, c) RESOURCE UTILIZATION of each schedule, d) SCHEDULE CRITICALITY INDEX, and e) NETWORK LOGIC INDEX. Criteria (a), (b), and (c) are incorporated in the CM-DSS algorithm based on the results obtained from the questionnaire.

c- Select a target schedule. In this step, CM-DSS uses a decision-making system along with the schedule criticality indices to help the planner in evaluating the four scheduling alternatives, and selecting a target schedule. In order to select a target schedule, the control criteria can be used in a form of a value matrix as can be seen in figure (3.2). Of course, it is up to the project manager to add any realistic control criterion to this matrix. In using this matrix, CM-DSS compares each control criterion to a preset upper limit. It measures : a) TOTAL COST of each schedule against the project allotted budget, b) DURATION of each schedule against the deadline imposed on the project by contractual obligation, c) RESOURCE UTILIZATION of each schedule against the labor resource requirements estimated by the planner, d) the SCI value as a direct measure of SCHEDULE CRITICALITY INDEX, and e) the NLI value as a direct measure of NETWORK CRITICALITY INDEX. Exhibit (3.18) demonstrates the steps required by the value matrix.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Control Criteria	RI	CBS		TBS		RBS		FS	
		RV 0 to 10	RV x RI	RV 0 to 10	RV x RI	RV 0 to 10	RV x RI	RV 0 to 10	RV x RI
Total Cost	25%	6	150	7.5	187.50	8.5	212.50	8	200
Duration	25%	4	100	8	200	4.5	112.50	7	175
Resource Utilization	20%	6.5	130	4.5	90	4	80.	7	140
Schedule Criticality Index	15%	8	120	6	90	7	105	5	75
Network Criticality Index	15%	4	60	5	75	6	90	7	105
Total Value	100%	560.00		642.50		600.00		695.00	

Figure(3.2).--VALUE MATRIX

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

EXHIBIT (3.18).--SCHEDULING VALUE MATRIX ALGORITHM

Step 0 : List a series of control criteria.

Step 1 : Weigh the relative importance (RI) of each criterion to the overall criteria.

Step 2 : Rate the extent to which each scheduling alternative meets the first criterion on a scale from 0 to 10 (call this the rating value RV). This is done by comparing the actual performance to a preset upper limit.

CM-DSS uses the following equations to scale each criterion:

a. RV = [contractual duration/project duration] x 10

b. RV = [project budget/project total cost] x 10

c. RV = [estimated man-hours/schedule resource utilization] x 10

d. RV = [1 - (NLI/100)] x 10

e. RV = [1 - (SCI/100)] x 10

Step 3 : For the first criterion, calculate the sub-value for each scheduling alternative by multiplying its RV by RI.

Step 4 : Repeat Step 3 for each control criteria.

Step 5 : Determine the total value for each scheduling alternative by adding up its sub-values.

Step 6 : Select the scheduling alternative that has the highest value as a target schedule.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

The use of this matrix allows the planner to evaluate the four scheduling alternatives based on the control criteria. He/she can conduct a What-If analysis by changing the relative importance of each control criterion. This helps him/her to examine different scenarios to choose the target schedule.

<u>d- Forecast potential deviation.</u> To effectively execute and control a project, it is important to predict any possible deviation from the target schedule. CM-DSS forecasts a possible deviation by analyzing the target schedule. For instance, in addition to the critical activities, the project manager can monitor the noncritical activities whose criticality index value ACI is at least 75% to prevent them from becoming critical or deviate from the target schedule. If the project manager is concerned with the allocated budget and wants to control the cost of the project, he/she can use the cost index to monitor the execution of the activities that have a high cost index value. Another way of employing the activity indices is to prepare a performance matrix similar to the one shown in table (3.1). In this matrix, only activities that have ACI, AINTI, and AINDI values at least 80%, ACOI and ARI values greater that 6% and an AGI value greater than 50% are considered. According to the performance matrix, anytime

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

an activity appears under one of the indices, it will get a score of (1) point. For instance, activity #12 appears 9 times, and thus it has a score of (9). During the control phase, any activity (critical or noncritical) may deviate from the target schedule and, accordingly, cause project delay or slippage. This matrix provides the planner with a forecasting tool that can determine the activities that have high tendency to become critical, regardless of the scheduling alternative that the planner may choose. To use this matrix, the planner should determine the minimum score above which an activity is considered to be candidate for future slippage. Of course, if the planner feels confident about the control system, he/she may increase the minimum score and concentrate on a smaller number of activities. This algorithm is summarized in exhibit (3.13).

During the planning phase, the final scheduling alternative to be executed is not known to the planner. Therefore, the four scheduling alternatives should be used in the forecasting process. The idea here is to evaluate the overall potential deviation for each activity no matter which scheduling alternative is to be executed later.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table (3.1).--ACTIVITY PERFORMANCE MATRIX

Index / ACTIVITY	COST-BASED SCHEDULE						TIME-BASED SCHEDULE						RESOURCE-BASED SCHEDULE						FEASIBLE SCHEDULE						Total Score
	ACI	AINT	AIND	IACO	ARI	AGI	ACI	AINT	AIND	IACO	ARI	AGI	ACI	AINT	AIND	IACO	ARI	AGI	ACI	AINT	AIND	IACO	ARI	AGI	
1			●						●						●						●				4
2			●	●	●				●		●				●						●	●			8
3	●	●	●				●	●	●			●	●	●	●			●	●	●	●			●	14
4			●						●						●						●				5
5			●	●	●				●		●				●						●	●	●		9
6	●	●	●			●	●	●	●			●	●	●	●			●	●	●	●			●	16
7			●						●						●						●				4
8			●					●	●						●						●				5
9	●	●	●	●		●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●		●	●	22
10	●		●	●					●						●	●					●	●	●		9
11	●		●					●	●	●					●						●				7
12		●	●		●			●	●					●	●					●	●				9
13			●						●						●						●				4
14		●	●					●	●					●	●	●				●	●				9
15	●	●	●			●	●	●	●			●	●	●	●			●	●	●	●			●	16
16		●		●		●		●		●				●		●				●		●			9
17	●	●	●			●	●	●	●			●	●	●	●			●	●	●	●			●	16
18	●	●	●			●	●	●	●			●	●	●	●			●	●	●	●			●	16
19	●	●	●			●	●	●	●			●	●	●	●			●	●	●	●			●	16
20	●	●	●			●	●	●	●			●	●	●	●			●	●	●	●			●	16
21	●	●	●			●	●	●	●			●	●	●	●			●	●	●	●	●		●	17
22	●	●	●		●	●	●	●	●		●	●	●	●	●		●	●	●	●	●	●	●	●	20

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

EXHIBIT (3.13).--ALGORITHM FOR FORECASTING DEVIATION

Step 0 : After selecting the target schedule, establish a matrix and list all activities and the four scheduling alternatives.

Step 1: Under each schedule list ACI, AINTI, AINDI, ACOI, ARI, and AGI. Determine an acceptable minimum value for each index. Only those activities that exceed this condition are considered for evaluation.

Step 2: Each activity that is considered for evaluation is given a score of 1 point.

Step 3: Repeat steps 1 and 2 for the remaining schedule.

Step 4: Calculate the total score for each activity.

Step 5: During the construction phase concentrate on the activities that have high scores. As a rule of thumb choose the activity that has a score higher than (8).

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

3.4 CPM ALGORITHM

Based on the presentation of chapter one, the following CPM algorithm is developed.

EXHIBIT (3.14).--CPM ALGORITHM

Step 0 : Determine the number of activities (ACTIVITIES) and events (EVENTS) for the network at hand.

Step 1 : Start at event (P:= 1). Label this event 1.

Step 2 : Move to the next unlabeled event. All activities merging into this event should emanate from a labeled event. 1).

Step 3: If all events are labeled, STOP. Otherwise, go to step 2.

Step 4 : For each activity (N), determine its start and finish events (i.e, nodes i and j), and its duration (Dur).

Step 5 : (Start the forward pass)

For each activity N, calculate

$ES_j := \max [ES_j, (ES_i + Dur)]$

Put N := N + 1

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Step 6 : If N <= ACTIVITIES, then go to step 7. Otherwise, put N := ACTIVITIES

LS (final event) := ES(N)

Now, all events have their earliest start time.

Step 7 : (Start the backward pass)

For each activity N, calculate

$LS_i := \min [LS_i, (LS_j - Dur)]$

Put N := N - 1.

Step 8 : If N > 0, go to step 9. Otherwise, go to step 11.

Step 9 : Initialize N := 0

Step 10 : Put activity (N := N + 1), and calculate the following:

$ES(N) = ES_i$

$LF(N) = LS_j$

$EF(N) = ES(N) + Dur$

$LS(N) = LF(N) - Dur$

$TF(N) = LF(N) - EF(N)$

$FF(N) = ES(N) - EF(N)$

$INTF(N) = ES_j - EF(N)$

$INDF(N) = ES_j - LS_i - Dur$

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Step 11 : If N := ACTIVITIES, then STOP. Otherwise, go to step 2.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

CHAPTER FOUR

CM-DSS COMPUTER PROGRAM

4.1 INTRODUCTION

The CM-DSS algorithm is comprehensive and multi-dimensional. This makes it difficult to manually exercise CM-DSS even on a very small network, especially if the planner elects to run what-if analyses, to examine different schedules, to select a target schedule, or to change the technological logic. Therefore, it becomes essential to computerize the algorithm. Computerizing the CM-DSS algorithm comprises the third major part of this research.

In this chapter, the design and the development of the computer program will be discussed and presented in detail.

4.2 COMPUTER LANGUAGE

The programming language Pascal was developed in 1971 by Professor Nicklaus Wirth of Zurich, Switzerland. It is a high-level, general-purpose language. It is designed to facilitate structured programs that are relatively easy to read and to understand[16]. Turbo Pascal, which is a dialect of Pascal, is used to develop the CM-DSS computer program.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

4.3 PROGRAM STRUCTURE

4.3.1 MODE

Typically, two principal modes of computer programming are available for programmers: batch and interactive. When using an interactive mode, the user can interact with the program and enter data during its execution. In a batch mode, the user should prepare all data beforehand, since the program does not allow any interaction while it is executing. As a good programming practice, it is recommended to use the extension (.DAT) to identify a data file.

4.3.2 MODULARITY

When dealing with a complicated programming problem, it is usual to break up the program into smaller subprograms. Pascal provides two types of subprograms: procedures and functions. A procedure is a group of program statements in a single program unit[16]. A function is another Pascal feature that is helpful in specifying numeric computations. Each function performs a different mathematical operation and returns a single value[16]. When these subprograms are included within a program, they carry out specific operations as part of the entire solution. They are usually structured sequentially in the main body of the program. In this case, the main body acts as a coordinator that assigns tasks to each subprogram.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Using modularity makes it easier to write and debug a program, since each subprogram can be written and debugged separately. This helps programmers to easily upgrade and revise an already written program by allowing him/her to quickly locate those parts that need modifying[16].

4.3.4.TOP-DOWN AND BOTTOM-UP PROGRAMMING

Typically, there are two approaches to Computer programming: top-down, and bottom-up. In top-down programming, the design process starts by, first, writing an outline of the main body in pseudocode. If the original problem is quite complicated, it may become necessary to refine the outline and to add more details as necessary. This approach is known as a stepwise refinement[16]. When the refinement process is completed, a final version of the main body can be obtained.

In bottom-up programming, the programmer develops the procedures first and the main body last. It is sometimes advantageous to employ a mixed approach, specially if the programmer is dealing with a complicated problem. In this case, he/she designs the main body and decides on the structure of the procedures and functions before determining exactly how all the parts of the main body will fit together[16].

The approach of using modularity, stepwise refinements, and some form of top-down programming in developing a computer program

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

is called structured programming. Its objective is to produce programs that are easier to (1) write, (2) debug, (3) read, and (4) modify[16].

4.4 CM-DSS COMPUTER MODEL

CM-DSS is a structured computer program. It employs a mix of top-down and bottom-up programming, stepwise refinements and the concept of modularity. Its mode is designed to be a mixed mode in order to keep the interaction between the user and the computer system at minimum and thus save both time and effort.

The CM-DSS computer model is shown in figure (4.1). In the planning phase, a planner starts the computer session by first loading the productivity-cost database on the computer's hard disk. If the planner elects to take the learning curve effect into account, the program starts by reading the learning rate for each repetitive activity and its execution order and adjusting the original man-hours available in the database, accordingly. If the planner decides to skip the learning curve feature, the program, then, will read the original man-hours, productivity rate, crew size and unit cost values and, accordingly, calculate the adjusted-hours and total cost. It performs this operation for each activity. At this stage, the user can specify one modular of constraints (i.e., resource, time, and budget) that is imposed on each activity. Consequently, the system develops four scheduling alternatives: (1) a cost-based schedule (assuming optimal

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

productivity), (2) a time-based schedule, (3) a resource-based schedule, and (4) a feasible schedule under several constraint modulars. The user may accept the feasible schedule, as a target schedule, or use the worth matrix in selecting the most feasible schedule among the scheduling alternatives. In using the worth matrix, the program selects the most feasible schedule after evaluating the four schedules based on the built-in control criteria that have been presented in section (3.3). If the user is still not satisfied with the target schedule, he/she may decide to change the network logic and replan the project. After conducting What-If analyses and selecting a target schedule, the user can obtain a hard copy of all the schedules' analyses.

4.4.1 CONSTANTS AND VARIABLES

Exhibit (4.1) shows three constants used by CM-DSS. For example, maxevents and maxactivities are used for the maximum number of events and activities that CM-DSS can accept. The use of these constants makes the task of future upgrading and revising the program much easier. In this case, the programmer can, for instance, change only the value of maxevents from 50 to 100 in the declaration part of the program, to account for an increase in the number of events in the network.

Exhibit (4.1) displays also the variables used by CM-DSS along with their meanings. These variables vary in their type and structure. In order to facilitate the use of the productivity-cost

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

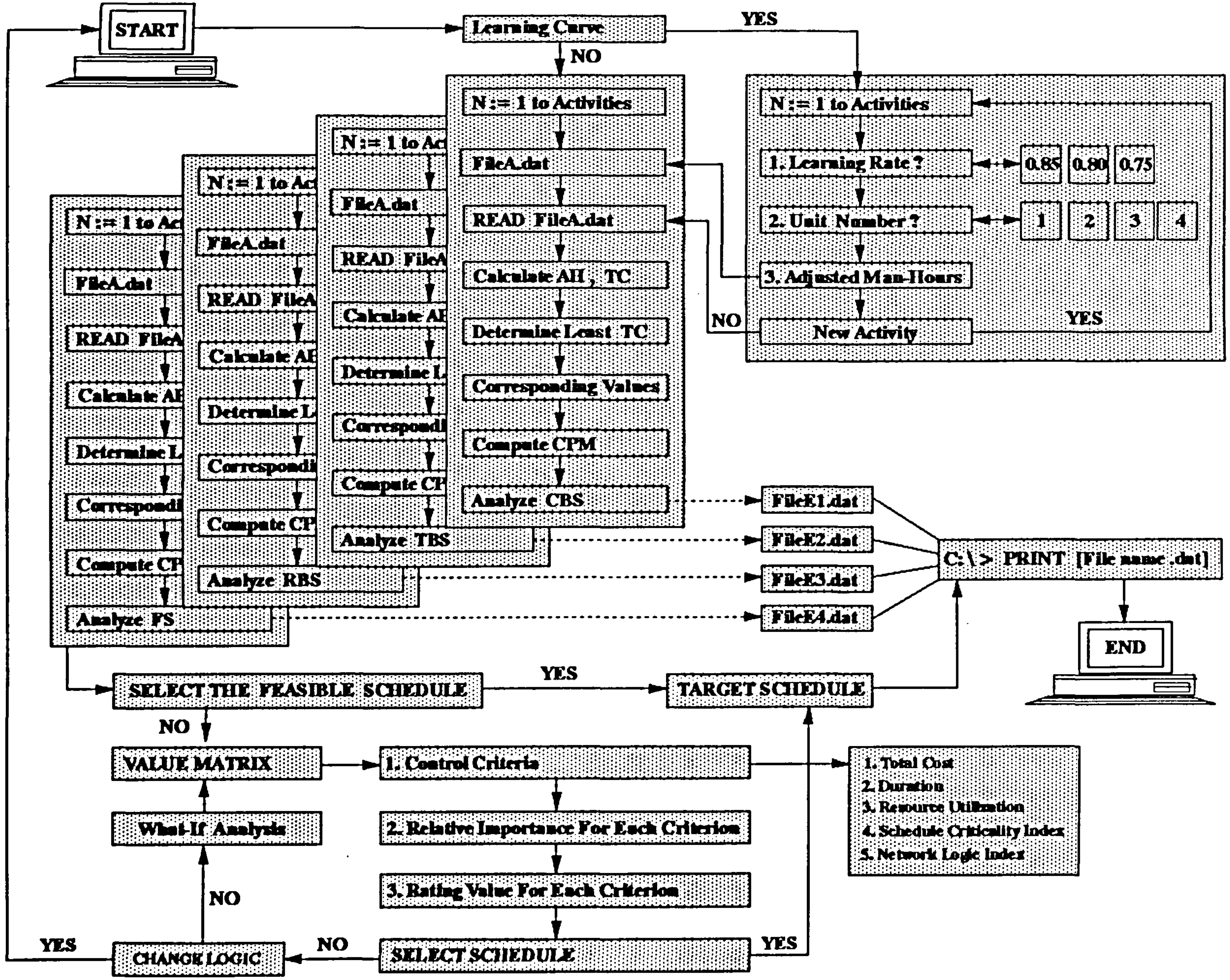


Figure (4.1).--CM-DSS Computer Model

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

database, CM-DSS employs the "record" structure that Pascal offers. A record structure instructs the computer program to move its pointer to an activity, to read its four modulars (rows), and to store in memory the exact position of each piece of information. To be used as a variable, each record should be defined first under "Type." Accordingly, its fields become known to the computer program. Exhibit (4.1) shows that the record used by CM-DSS is called ProductivityRec. This record is needed because the position of each value in the database should be specified by a row and a column. It contains 6 fields (MH, PR, CS, UC, AH, and TC). Each field consists of a one dimensional four-row array and each row represents one activity set. CM-DSS utilizes 148 variables of different types: real, integer, character, array, and user-defined records). As an essntial part of good programming practice, these variables are assigned, when ever possible, meaningful names. These variables are presented in the following sections.

EXHIBIT (4.1).--CM-DSS CONSTANTS AND VARIABLES

```
const {.................CONSTANTS........................}

      maxevents       = 50;
      maxactivities   = 50;
      Largenumber     = 10000;

type
      {.................TYPES...........................}

        ActivityArr       = array[1..maxactivities] of integer;
      EventArr          = array[1..maxnodes] of integer;
      ActivityIndex     = 1..50;


      ProductivityRec = record
      MH: array [1..4] of real;   {MH = Man-Hours          }
      PR: array [1..4] of real;   {PR = Productivity Rate }
      CS: array [1..4] of real;   {CS = Crew Size          }
      UC: array [1..4] of real;   {UC = Unit Cost ($)      }
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
      AH: array [1..4] of real;   {AH = Adjusted Hour      }
      TC: array [1..4] of real;   {TC = Total Cost         }
      end; {ProductivityRec}
      ActivityArray = array[ActivityIndex] of ProductivityRec;

var   {..................VARIABLES.........................}

      answer1, answer2, YES, NO                                   : char;
      RI, sub1, sub2, sub3, sub4                                  : real;
      worth1, worth2, worth3, worth4                              : real;
      RV1, RV2, RV3, RV4                                          : real;
      RES1,RES2,RES3,RES4                                         : real;
      BUDGET,Cost1,Cost2,Cost3,Cost4                              : real;
      factor                                                      : real;
      events,activities                                           : integer;
      M, N, K                                                     : integer;
      number,ProjectDur1,ProjectDur2,ProjectDur3,ProjectDur4      : integer;
      CD,MaxResource                                              : integer;
      Act                                                         : ActivityArray;

      i             {i    = Node i                    } ,
      j             {j    = Node j                    } ,
      Dur           {Dur  = Duration                  } ,
      ES            {ES   = Early Start               } ,
      EF            {EF   = Early Finish              } ,
      LS            {LS   = Late Start                } ,
      LF            {LF   = Late Finish               } ,
      FF            {FF   = Free Float                } ,
      TF            {TF   = Total Float               } ,
      INTF          {INTF = Interfering Float         } ,
      INDF          {INDF = Independent Float         }   : Activityarr;
      EST           {EST  = Early Start Time          } ,
      LST           {LST  = Late  Start Time          }   : Eventarr;


      Pro                                                         : ProductivityRec;
      Least_Time1,Least_Time2,Least_Time3,Least_Time4             : real;
      Least_Cost1,Least_Cost2,Least_Cost3,Least_Cost4             : real;
      duration1, duration2, duration3, duration4                  : real;
      resource1, resource2, resource3, resource4                  : real;
      Total_Cost1,Total_Cost2,Total_Cost3,Total_Cost4             : real;
      TOTAL1,TOTAL2,TOTAL3,TOTAL4                                 : real;

      NLI1  {NLI1  = Network Logic Index  for        CBS} ,
      NLI2  {NLI2  = Network Logic Index  for        TBS} ,
      NLI3  {NLI3  = Network Logic Index  for        RBS} ,
      NLI4  {NLI4  = Network Logic Index  for        FS } ,

      SCI1  {SCI1  = Schedule Criticality Index for  CBS} ,
      SCI2  {SCI2  = Schedule Criticality Index for  TBS} ,
      SCI3  {SCI3  = Schedule Criticality Index for  RBS} ,
      SCI4  {SCI4  = Schedule Criticality Index for  FS } ,

      SCOI1 {SCOI1 = Schedule Cost Index for         CBS} ,
      SCOI2 {SCOI2 = Schedule Cost Index for         TBS} ,
      SCOI3 {SCOI3 = Schedule Cost Index for         RBS} ,
      SCOI4 {SCOI4 = Schedule Cost Index for         FS } ,

      SRI1  {SRI11 = Schedule Resource Index for     CBS} ,
      SRI2  {SRI12 = Schedule Resource Index for     TBS} ,
      SRI3  {SRI13 = Schedule Resource Index for     RBS} ,
      SRI4  {SRI14 = Schedule Resource Index for     FS } ,

      TACI1 {TACI1 = Total Activity Criticality Index for  CBS} ,
      TACI2 {TACI2 = Total Activity Criticality Index for  TBS} ,
      TACI3 {TACI3 = Total Activity Criticality Index for  RBS} ,
      TACI4 {TACI4 = Total Activity Criticality Index for  FS } ,

      TACOI1{TACOI1= Total Activity Cost Index for   CBS} ,
      TACOI2{TACOI2= Total Activity Cost Index for   TBS} ,
      TACOI3{TACOI3= Total Activity Cost Index for   RBS} ,
      TACOI4{TACOI4= Total Activity Cost Index for   FS } ,
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
TAD1  (TAD1  = Total Activity Durations for        CBS) ,
TAD2  (TAD2  = Total Activity Durations for        TBS) ,
TAD3  (TAD3  = Total Activity Durations for        RBS) ,
TAD4  (TAD4  = Total Activity Durations for        FS ) ,

TARI1 (TARI1 = Total Activity Resource Index for   CBS) ,
TARI2 (TARI2 = Total Activity Resource Index for   TBS) ,
TARI3 (TARI3 = Total Activity Resource Index for   RBS) ,
TARI4 (TARI4 = Total Activity Resource Index for   FS ) ,

TRU1  (TRU1  = Total Resource Utilization for      CBS) ,
TRU2  (TRU2  = Total Resource Utilization for      TBS) ,
TRU3  (TRU3  = Total Resource Utilization for      RBS) ,
TRU4  (TRU4  = Total Resource Utilization for      FS )    : real;

ACI1          : array [1..50] of real;
ACI2          : array [1..50] of real;    { ACTIVITY CRITICALITY INDICES }
ACI3          : array [1..50] of real;
ACI4          : array [1..50] of real;

ACOI1         : array [1..50] of real;
ACOI2         : array [1..50] of real;    { ACTIVITY COST INDICES }
ACOI3         : array [1..50] of real;
ACOI4         : array [1..50] of real;

AGI1          : array [1..50] of real;
AGI2          : array [1..50] of real;    { ACTIVITY GLOBAL INDICES }
AGI3          : array [1..50] of real;
AGI4          : array [1..50] of real;

ARI1          : array [1..50] of real;
ARI2          : array [1..50] of real;    { ACTIVITY RESOURCE INDEX }
ARI3          : array [1..50] of real;
ARI4          : array [1..50] of real;

AINTI1        : array [1..50] of real;
AINTI2        : array [1..50] of real;    { ACTIVITY INTERFERING INDEX }
AINTI3        : array [1..50] of real;
AINTI4        : array [1..50] of real;

AINDI1        : array [1..50] of real;
AINDI2        : array [1..50] of real;    { ACTIVITY INDEPENDENT INDEX }
AINDI3        : array [1..50] of real;
AINDI4        : array [1..50] of real;

Optimal_Cost1: array [1..4]  of real;
Optimal_Cost2: array [1..4]  of real;
Optimal_Cost3: array [1..4]  of real;
Optimal_Cost4: array [1..4]  of real;

COS           : array [1..50] of real;
RES           : array [1..50] of real;
duration      : array [1..50] of real;
resource      : array [1..50] of real;
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

4.4.2 FILES

The CM-DSS computer program utilizes 19 files while executing. Exhibit (4.3) shows these files along with their contents. The design of this program is based on manipulating a given data file, formatting and storing the results in specific files, using them to perform the required analyses and, finally, saving them in output files. Figure (4.2) shows the flow of information through these files.

EXHIBIT (4.2).--CM-DSS INPUT AND OUTPUT FILES

```
     fileA1   { Contains the original Database prepared by the user         },
     fileA  { Contains the modified Database after considering the Learning Effect },

{*** FILES FROM B1 TO E1 BELONG TO THE COST-BASED SCHEDULE (CBS)      ***}

     fileB1  { Contains the MH, PR, CS, UC, AH, and TC values              },
     fileC1  { Contains the CS, AH, and TC in an organized format          },
     fileD1  { Contains the durations used to produce CSB                  },
     fileE1  { Contains the final CPM analyses for CSB.                    },

{*** FILES FROM B2 TO E2 BELONG TO THE TIME-BASED SCHEDULE (TBS).     ***}

     fileB2  { Contains the MH, PR, CS, UC, AH, and TC values              },
     fileC2  { Contains the CS, AH, and TC in an organized format          },
     fileD2  { Contains the durations used to produce TBS                  },
     fileE2  { Contains the final CPM analyses for TBS                     },

{*** FILES FROM B3 TO E3 BELONG TO THE RESOURCE-BASED SCHEDULE (RBS).***}

     fileB3  { Contains the MH, PR, CS, UC, AH, and TC values              },
     fileC3  { Contains the CS, AH, and TC in an organized format          },
     fileD3  { Contains the durations used to produce RBS                  },
     fileE3  { Contains the final CPM analyses for RBS                     },

{*** FILES FROM B4 TO E4 BELONG TO THE FEASIBLE SCHEDULE (FS).         ***}

     fileB4  { Contains the MH, PR, CS, UC, AH, and TC values              },
     fileC4  { Contains the CS, AH, and TC                                 },
     fileD4  { Contains the durations used to produce FS                   },
     fileE4  { Contains the final CPM analyses for FS                      },
     fileB5  { This file is used as a temporary file                       } :text;
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

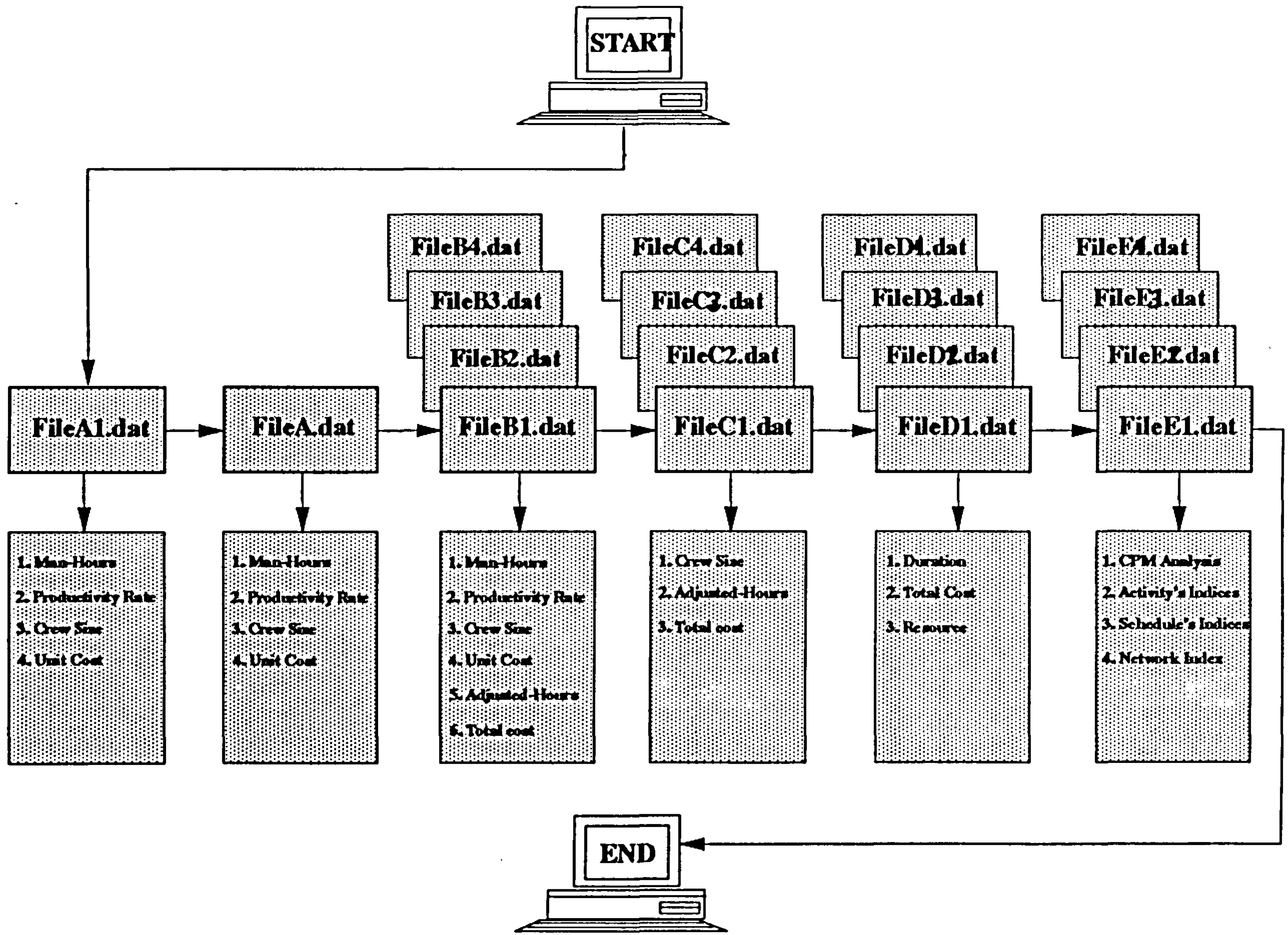


Figure (4.2).--CM-DSS Data Files

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

4.4.3 PROCEDURES

CM-DSS Incorporates 16 procedures. Exhibit (4.3) shows these procedures as they appear in the main body of the program. They are organized, in a specific sequential order. The first procedure acts as an independent small program, executes specific tasks, and stores the calculated values into the computer's memory. At a later stage, the main body instructs the procedure that follows to use the stored information in order to perform another part of the whole solution. This sequential execution leads to the final analysis for the project at hand.

In the following sections, seven major procedures used by CM-DSS will be presented in the same order of their appearance in the main body.

EXHIBIT (4.3). CM-DSS PROCEDURES

```
L_Cost;      {******** First Schedule *******}
readinfo1;
CPM1;
path1;
L_Time;      {******** Second Schedule ******}
readinfo2;
CPM2;
path2;
L_Resource;  {******* Third Schedule ********}
readinfo3;
CPM3;
path3;
Feasible;    {******* Fourth Schedule *******}
readinfo4;
CPM4;
path4;
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

4.4.3.1 Description Of L_Cost Procedure

This procedure is designed to produce the cost-based schedule. Figure (4.3) presents the computer model of procedure L_Cost. The process starts by reading the information of the first activity as it appears in the database, calculating the adjusted-hours and total cost values, and storing these values in a file [fileC1.dat]. The procedure, then, selects the least total cost value and the corresponding adjusted hours and crew size values. The procedure repeats this iteration for each activity. Finally, its stores all the results in a file [fileD1.dat].

Exhibit (4.4) shows the source code of procedure L_Cost. It consists of two [While] loops and one [For] loop. The first while loop reads the database [fileA1.dat], manipulates its data, and calculates the adjusted-hours (Pro.AH[M]) and the total cost (Pro.TC[M]). It performs this operation for each activity four times (from M := 1 to 4) and stores these values in a file [fileB1.dat]. Then, the second [While] loop reads file [fileB1.dat], reorganizes it, and rewrites its contents in file [fileC1.dat]. The third loop is a [For] loop. Its task is to search the file (fileB1.dat), and to select the least-cost value of each activity. Finally, it stores the least-cost value along with the corresponding adjusted-hours and crew size values in a file [fileD1.dat].

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

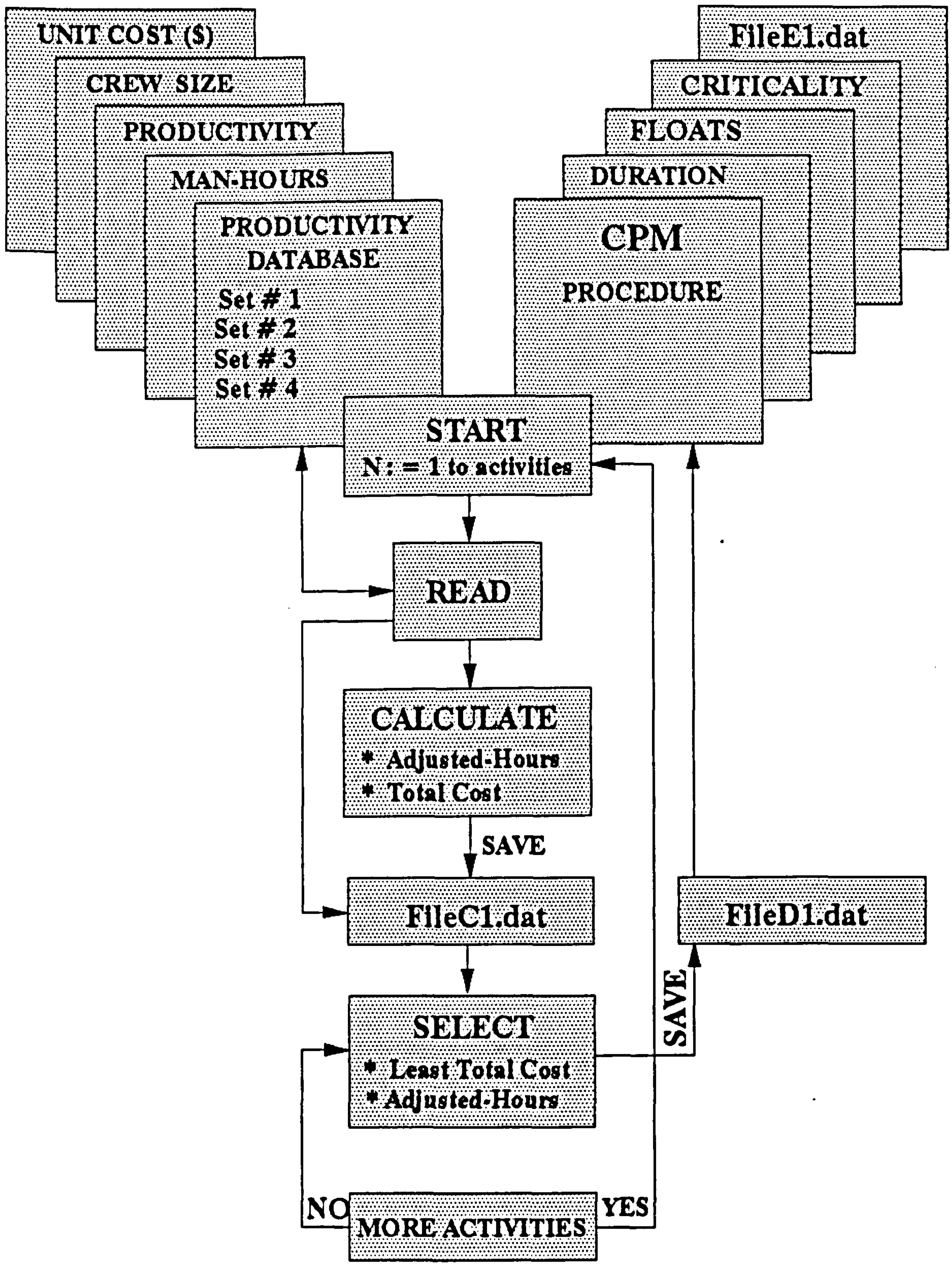


Figure (4.3).--Computer Model For Procedure L-Cost

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

EXBIBIT (4.4).--SOURCE CODE OF PROCEDURE L_COST

```
procedure L_cost;

begin  (Read the Database and Develop the Cost_Based Schedule)
   clrscr;
   writeln;
   writeln ('      *******************************');
   writeln ('      PLEASE RESPOND TO EACH PROMPT');
   writeln ('      *******************************');
   writeln;
   write ('      How many activities do you have ? >  ' );
   readln (activities);
   assign (fileA, 'c:prodA.dat');
   reset (fileA);
   assign (fileB1, 'c:prodB1.dat');
   rewrite(fileB1);
   assign (fileC1,'c:prodC1.dat');
   rewrite(fileC1);
   while not eof (fileA) do
     begin
       for M:= 1 to 4 do
         begin
           readln (fileA,Pro.MH[M],Pro.PR[M],Pro.CS[M],Pro.UC[M]);
           Pro.AH[M]:= Pro.MH[M]/(Pro.PR[M] * Pro.CS[M]);
           Pro.TC[M]:= Pro.AH[M] * Pro.UC[M] * Pro.CS[M];
           writeln(fileB1,Pro.MH[M],Pro.PR[M],Pro.CS[M],Pro.UC[M],Pro.AH[M],Pro.TC[M]);
         end;  (end for loop)
     end;     (end while loop)
   close (fileA);
   reset (fileB1);
   while not eof (fileB1) do
     begin
       writeln('IF YOU HAVE SOME CONSTRAINED ACTIVITIES');
       writeln('PLEASE READ THE FOUR MODULARS FOR EACH ACTIVITY');
       writeln('AS THEY APPEAR ON THE SCREEN AND SELECT ONE MODULAR ONLY.');
       writeln('IF YOU DO NOT HAVE CONSTRAINED ACTIVITIES, PLEASE ENTER (0, 0, 0);
       writeln('CM-DSS WILL GIVE YOU ENOUGH TIME TO READ THIS INFORMATION.');
       writeln;
       writeln('*******************************************************');
       writeln('Act.','     ','MH','      ','PR','     ','CS','     ','UC','     ',' AH','    ','  TC');
       writeln('*******************************************************');
       writeln;
       writeln(fileC1,'*******************************************************');
       writeln(fileC1,'Act.','     ','MH','      ','PR','     ','CS','     ','UC','     ',' AH','    ','  TC');
       writeln(fileC1,'*******************************************************');
       for N:= 1 to activities do
         begin
          writeln(fileC1,N);
          for M := 1 to 4 do
             begin
               readln(fileB1,Act[N].MH[M],Act[N].PR[M],Act[N].CS[M],Act[N].UC[M],Act[N].AH[M],Act[N].TC[M]);
               write(fileC1,'        ',Act[N].MH[M]:5:3,'  ',Act[N].PR[M]:5:3,'  ',Act[N].CS[M]:5:3,'  ');
               writeln(fileC1,Act[N].UC[M]:5:3,'  ',Act[N].AH[M]:5:3,'  ',Act[N].TC[M]:5:3);
             end;
          writeln(fileC1,'==================================================');
          writeln('==================================================');
          Delay(3000);
         end;
     end;
  Close (fileB1);
  Close (fileC1);

  assign (fileD1, 'c:prodD1.dat');
  rewrite(fileD1);
  resource1   := 0;
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
  duration1    := 0;
  Least_Cost1 := 0;
  Total_Cost1 := 0;
  TAD1        := 0;
  TRU1        := 0;

  for N := 1 to activities do
    begin
      M:= 1;
      resource1 := Act[N].CS[M];
      duration1 := Act[N].AH[M];
      Least_Cost1 := Act[N].TC[M];
      for M := 2 to 4 do
         if Act[N].TC[M] < Least_Cost1
            then
              begin
                 resource1   := Act[N].CS[M];
                 duration1   := Act[N].AH[M];
                 Least_Cost1 := Act[N].TC[M];
              end
             else
               begin
                 resource1   := resource1;
                 duration1   := duration1;
                 Least_Cost1 := Least_Cost1;
               end;
      if resource1 = 1 then
         RES[N] := 0
      else
         RES[N] := resource1 * duration1;
      duration1 := duration1;
      Optimal_Cost1 [N]:= Least_Cost1;
      TRU1 := TRU1 + (resource1 * duration1);
      TAD1 := TAD1 + duration1;
      Total_Cost1  :=  Total_Cost1 + Optimal_Cost1 [N];
      writeln(fileD1,duration1:5:0,Optimal_Cost1[N]:10:2,RES[N]:6:0);
    end;
    Cost1 := Total_Cost1;
    TAD1  := TAD1;
    TRU1  := TRU1;
    close (fileD1);

    writeln('       ***********************************************
    writeln('       CM-DSS IS DEVELOPING THE COST BASED SCHEDULE
    writeln('       ***********************************************
    writeln;
    writeln('       PLEASE WAIT.........');

    Delay(15000);

end;
```

4.4.3.2 Description Of Procedure L_Time

This procedure is almost identical to procedure L_Cost except that it searches for the least-time of each activity and saves it along with the corresponding total cost and crew size values in a file [fileD2.dat]. Figure (4.4) shows the computer model of procedure L_Time while exhibit (4.5) displays its source code.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

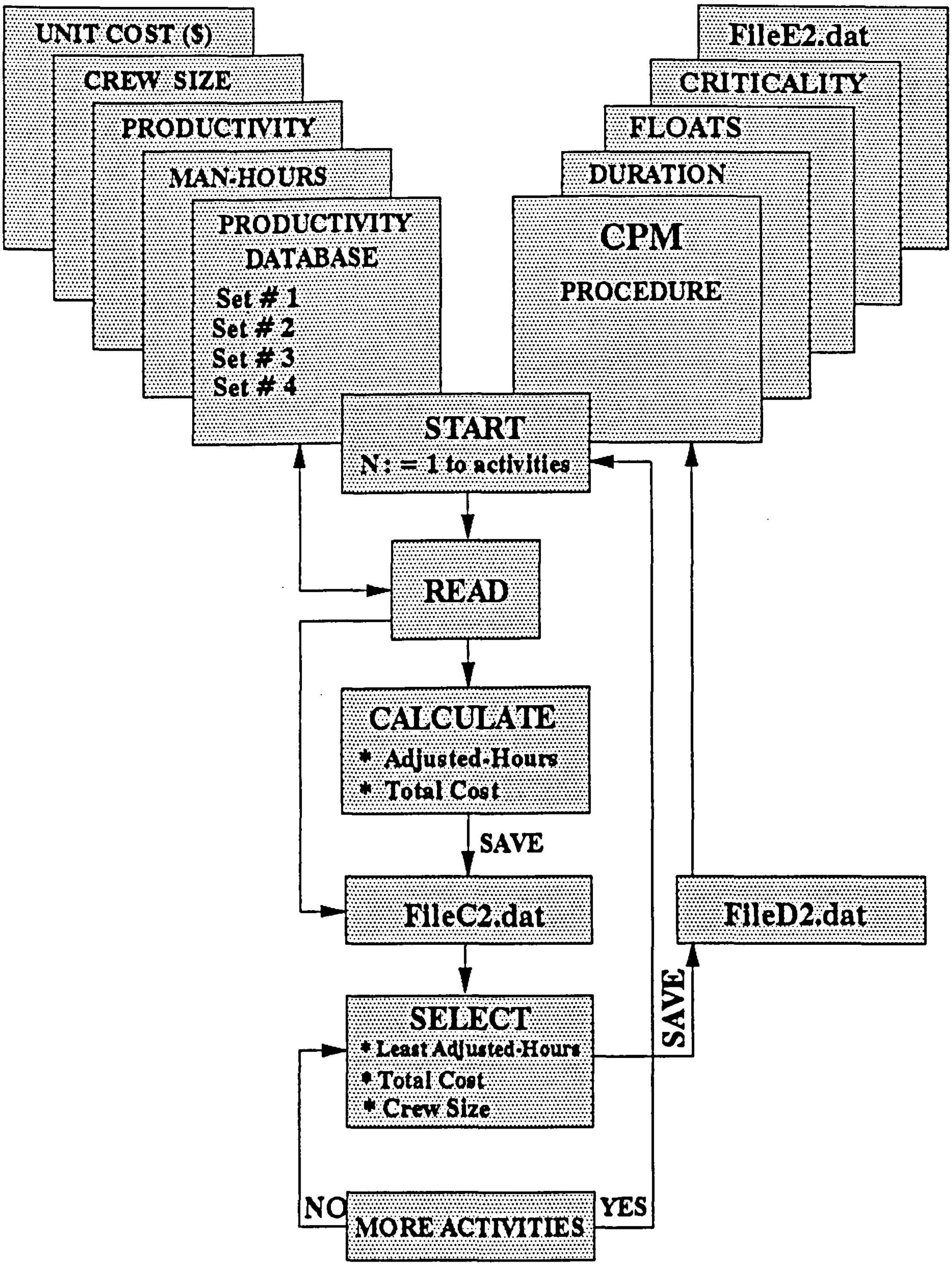


Figure (4.4).--Computer Model For Procedure L-Time

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

EXHIBIT (4.5).--SCOURCE CODE OF PROCEDURE L_TIME

```
procedure L_Time;

begin {Read the Database and Develop the Time_Based Schedule}
   clrscr;
   assign (fileA, 'c:prodA.dat');
   reset (fileA);
   assign (fileB2, 'c:prodB2.dat');
   rewrite(fileB2);
   assign (fileC2,'c:prodC2.dat');
   rewrite(fileC2);

   while not eof (fileA) do
     begin
       for M:= 1 to 4 do
          begin
            readln (fileA,Pro.MH[M],Pro.PR[M],Pro.CS[M],Pro.UC[M]);
            Pro.AH[M]:= Pro.MH[M]/(Pro.PR[M] * Pro.CS[M]);
            Pro.TC[M]:= Pro.AH[M] * Pro.UC[M] * Pro.CS[M];
            writeln(fileB2,Pro.CS[M],Pro.AH[M],Pro.TC[M]);
          end;  {end for loop}
     end;      {end while loop}

   close (fileA);
   reset (fileB2);

   while not eof (fileB2) do
     begin
       writeln(fileC2,'*******************************************************');
       writeln(fileC2,'Act.','   ' ,'CS','    ',' AH','    ',' TC');
       writeln(fileC2,'*******************************************************');
       for N:= 1 to activities do
         begin
          writeln(fileC2,N);
          for M := 1 to 4 do
             begin
               readln (fileB2,Act[N].CS[M],Act[N].AH[M],Act[N].TC[M]);
               writeln(fileC2,'        ',Act[N].CS[M]:5:3,'  ',Act[N].AH[M]:5:3,'  ',Act[N].TC[M]:5:3);
             end;
          writeln(fileC2,'================================================');
         end;
      end;

  Close (fileB2);
  Close (fileC2);

  assign (fileD2, 'c:prodD2.dat');
  rewrite(fileD2);
  resource2   := 0;
  duration2   := 0;
  Least_Cost2 := 0;
  Total_Cost1 := 0;
  Total_Cost2 := 0;
  TAD2        := 0;
  TRU2        := 0;

  for N := 1 to activities do
    begin
      M:= 1;
      resource2   := Act[N].CS[M];
      duration2   := Act[N].AH[M];
      Least_Cost2 := Act[N].TC[M];
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
      for M := 2 to 4 do
         if Act[N].AH[M] < duration2
            then
               begin
                 resource2   := Act[N].CS[M];
                 duration2   := Act[N].AH[M];
                 Least_Cost2 := Act[N].TC[M];
               end
             else
               begin
                 resource2   := resource2;
                 duration2   := duration2;
                 Least_Cost2 := Least_Cost2;
               end;
      if resource2 = 1 then
         RES[N] := 0
      else
         RES [N] := resource2 * duration2;
      duration2 := duration2;
      Optimal_Cost2 [N]:= Least_Cost2;
      TRU2 := TRU2 + (resource2 * duration2);
      TAD2 := TAD2 + duration2;
      Total_Cost2  :=  Total_Cost2 + Optimal_Cost2 [N];
      writeln(fileD2,duration2:5:0,Optimal_Cost2[N]:10:2,RES[N]:6:0);
    end;

    Cost2 := Total_Cost2;
    TAD2 := TAD2;
    TRU2 := TRU2;

    close (fileD2);
end; {end of L_Time}
```

4.4.3.3 Description Of Procedure L_Resource

This procedure is almost identical to procedure L_Cost. The only exception is that it searches for the least-resource of each activity and saves it along with the corresponding adjusted-hours and total cost values in a file [fileD3.dat]. Figure (4.5) shows its computer model and exhibit (4.6) displays the source code of this procedure.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

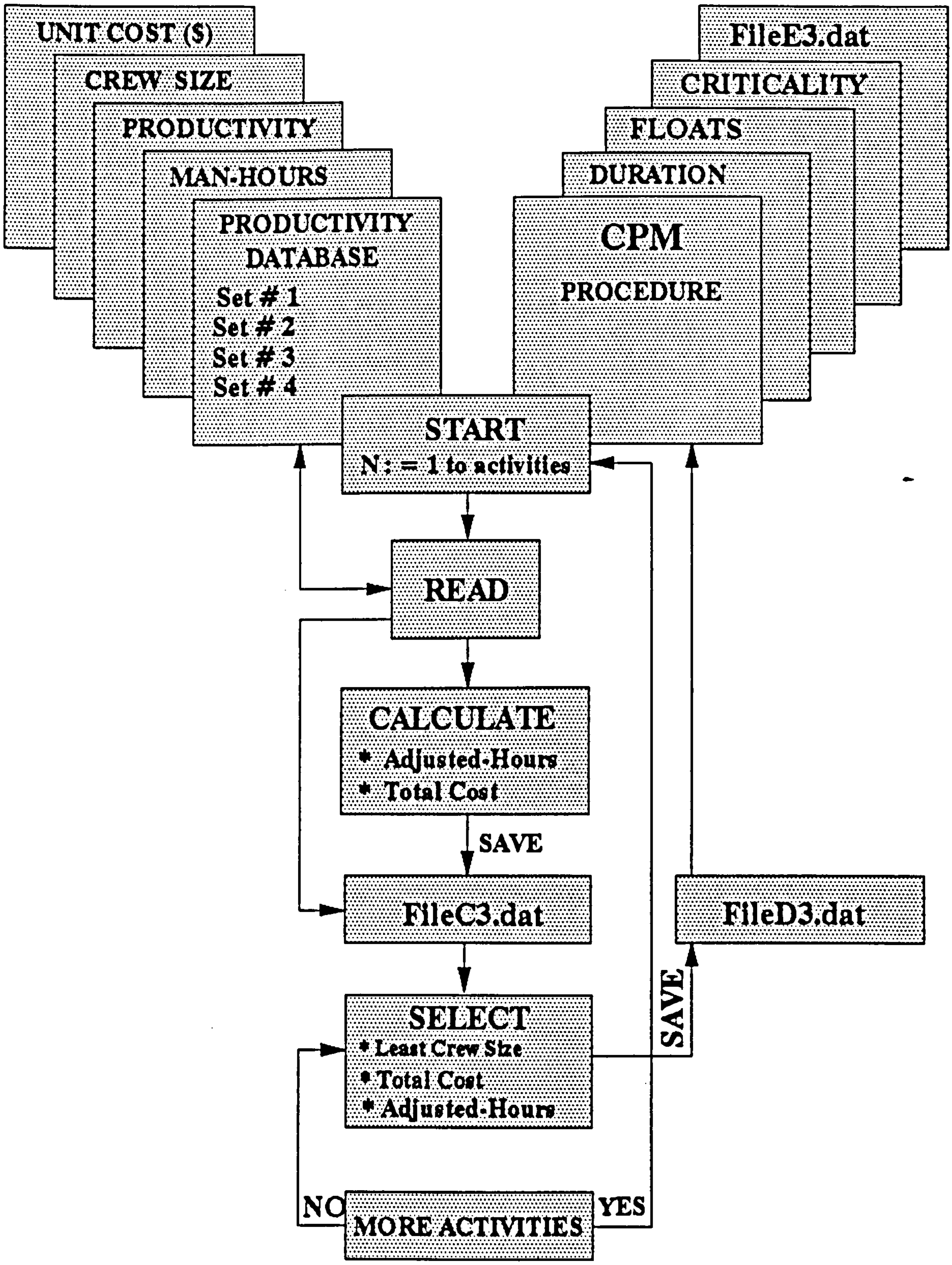


Figure (4.5).--Computer Model For Procedure L-Resource

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

EXHIBIT (4.6).--SOURCE CODE OF PROCEDURE L_RESOURCE

```
procedure L_Resource;

 begin    {Read Database and Develop the Resource_Based Schedule}
   assign (fileA, 'c:prodA.dat');
   reset (fileA);
   assign (fileB3, 'c:prodB3.dat');
   rewrite(fileB3);
   assign (fileC3,'c:prodC3.dat');
   rewrite(fileC3);
    while not eof (fileA) do
     begin
       for M:= 1 to 4 do
          begin
            readln (fileA,Pro.MH[M],Pro.PR[M],Pro.CS[M],Pro.UC[M]);
            Pro.AH[M]:= Pro.MH[M]/(Pro.PR[M] * Pro.CS[M]);
            Pro.TC[M]:= Pro.AH[M] * Pro.UC[M] * Pro.CS[M];
            writeln(fileB3,Pro.CS[M],Pro.AH[M],Pro.TC[M]);
          end;  {end for loop}
     end;     {end while loop}
   close (fileA);
   reset (fileB3);
   while not eof (fileB3) do
     begin
       writeln(fileC3,'********************************************************');
       writeln(fileC3,'Act.','    ' ,'CS','     ',' AH',' ',' TC');
       writeln(fileC3,'********************************************************');
       for N:= 1 to activities do
         begin
          writeln(fileC3,N);
          for M := 1 to 4 do
             begin
               readln (fileB3,Act[N].CS[M],Act[N].AH[M],Act[N].TC[M]);
               writeln(fileC3,'        ',Act[N].CS[M]:5:3,'  ',Act[N].AH[M]:5:3,'  ',Act[N].TC[M]:5:3);
             end;
          writeln(fileC3,'==================================================');
         end;
     end;
  Close (fileB3);
  Close (fileC3);
  assign (fileD3, 'c:prodD3.dat');
  rewrite(fileD3);

  resource3    := 0;
  duration3    := 0;
  Least_Cost3 := 0;
  Total_Cost3 := 0;
  TAD3        := 0;
  TRU3        := 0;
  for N := 1 to activities do
    begin
      M:= 1;
      resource3    := Act[N].CS[M];
      duration3    := Act[N].AH[M];
      Least_Cost3 := Act[N].TC[M];
      for M := 2 to 4 do
         if Act[N].CS[M] < resource3
            then
               begin
                 resource3    := Act[N].CS[M];
                 duration3    := Act[N].AH[M];
                 Least_Cost3 := Act[N].TC[M];
               end
             else
               begin
                 resource3    := resource3;

                 duration3    := duration3;
                 Least_Cost3 := Least_Cost3;
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
            end;
      if resource3 = 1 then
         RES[N] := 0
      else
         RES [N]    := resource3 * duration3;
      duration3     := duration3;
      Optimal_Cost3 [N] := Least_Cost3;
      TRU3          := TRU3 + (resource3 * duration3);
      TAD3          := TAD3 + duration3;
      Total_Cost3   := Total_Cost3 + Optimal_Cost3 [N];
     writeln(fileD3,duration3:5:0,Optimal_Cost3[N]:10:2,RES[N]:6:0);
    end;
    Cost3 := Total_Cost3;
    TAD3 := TAD3;
    TRU3 := TRU3;
    close (fileD3);

end; {end of L_Time}
```

4.4.3.4 Description Of Procedure Feasible

Figure (4.6) presents the computer model of procedure Feasible. This model is similar to the previous ones except that it, first, allows the user to input activity constraints through the computer keyboard, reads the database, and then performs the operations shown in figure (4.6).

This procedure differs from the previous three procedures in that its structure is based on two [For] loops and one [While] loop. The first [For] loop asks the user to input one modular of constraints for each activity, if any. If the user responds by entering (0,0,0), the procedure will understand that the activity in question has no constraints. Consequently, the procedure reads the original database and saves it without any modification in a file [fileB4.dat]. If some activities have constraints, the procedure will store the values of the constraints directly in a file [fileB4.dat]. The remaining [While] and [For] loops are identical to the ones used in the last three procedures.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

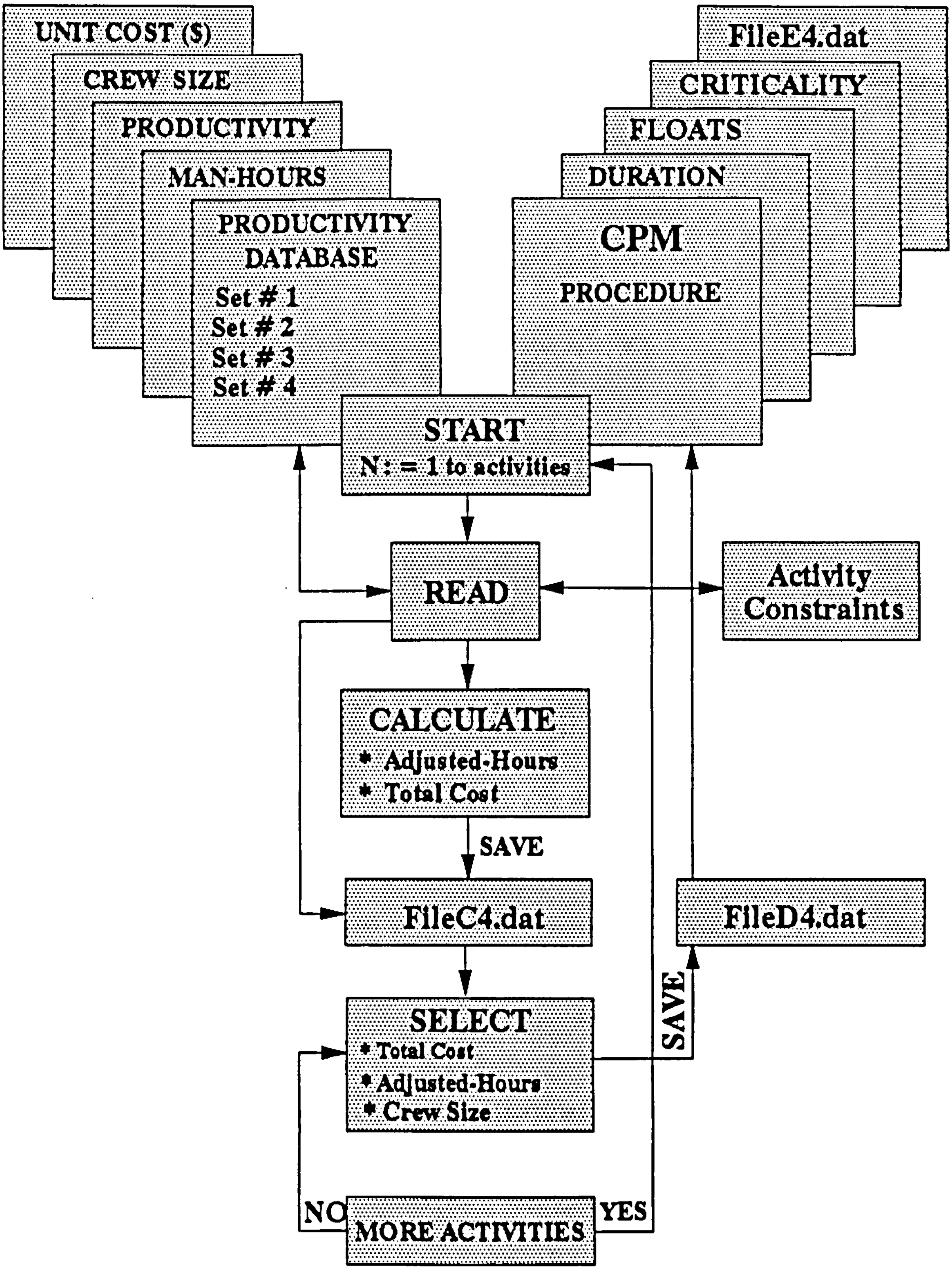


Figure (4.6).--Computer Model For Procedure Feasible

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

EXHIBIT (4.7).--SOURCE CODE OF PROCEDURE FEASIBLE

```
procedure Feasible;

begin  {Read the Constraints and the Database and Develop the Feasible Schedule}
   assign (fileB4, 'c:prodB4.dat');
   rewrite (fileB4);
   assign (fileB2, 'c:prodB2.dat');
   reset (fileB2);
   assign (fileB5, 'c:prodB5.dat');
   rewrite (fileB5);

   for N := 1 to activities do
    begin
      M := 1;
      writeln ('      Please enter one constraint for activity no.', N );
      write ('     Resource ', ' Time ', ' &  Budget >  ');
      readln (Act[N].CS[M],Act[N].AH[M],Act[N].TC[M]);
      writeln;
      if Act[N].CS[M] > 0 then
        begin
          writeln (fileB4,Act[N].CS[M],Act[N].AH[M],Act[N].TC[M]);
          writeln (fileB4,Act[N].CS[M],Act[N].AH[M],Act[N].TC[M]);
          writeln (fileB4,Act[N].CS[M],Act[N].AH[M],Act[N].TC[M]);
          writeln (fileB4,Act[N].CS[M],Act[N].AH[M],Act[N].TC[M]);
          for M := 1 to 4 do
            begin
              readln (fileB2,Act[N].CS[M],Act[N].AH[M],Act[N].TC[M]);
              writeln(fileB5,Act[N].CS[M],Act[N].AH[M],Act[N].TC[M]);
            end;
        end
      else if Act[N].CS[M] = 0 then
        begin
          for M := 1 to 4 do
            begin
              readln (fileB2,Act[N].CS[M],Act[N].AH[M],Act[N].TC[M]);
              writeln(fileB4,Act[N].CS[M],Act[N].AH[M],Act[N].TC[M]);
            end;
        end; { end if }
    end;  { end N for loop }

   assign (fileC4,'c:prodC4.dat');
   rewrite(fileC4);
   reset (fileB4);

   while not eof (fileB4) do
     begin
       writeln(fileC4,'*************************************************');
       writeln(fileC4,'Act.',' CS',' ',' AH',' ',' TC');
       writeln(fileC4,'*************************************************');

       for N:= 1 to activities do
         begin
          writeln(fileC4,N);

          for M := 1 to 4 do
             begin
               readln (fileB4,Act[N].CS[M],Act[N].AH[M],Act[N].TC[M]);
               write(fileC4,'        ',Act[N].CS[M]:5:3,'  ',Act[N].AH[M]:5:3,'  ',Act[N].TC[M]:5:3);
             end;
          writeln(fileC4,'=====================================================');
         end;
     end;

   Close (fileB4);
   Close (fileC4);
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
assign (fileD4, 'c:prodD4.dat');
rewrite(fileD4);


resource4   := 0;
duration4   := 0;
Least_Cost4 := 0;
Total_Cost4 := 0;
TAD4        := 0;
TRU4        := 0;


for N := 1 to activities do
  begin
    M:= 1;
    resource4   := Act[N].CS[M];
    duration4   := Act[N].AH[M];
    Least_Cost4 := Act[N].TC[M];

    for M := 2 to 4 do
       if Act[N].TC[M] < Least_Cost4
          then
             begin
               resource4   := Act[N].CS[M];
               duration4   := Act[N].AH[M];
               Least_Cost4 := Act[N].TC[M];
             end
           else
             begin
               resource4   := resource4  ;
               duration4   := duration4  ;
               Least_Cost4 := Least_Cost4;
              end;


     if resource4 = 1 then
       RES[N] := 0
    else
       RES[N] := resource4 * duration4 ;
    Optimal_Cost4 [N]:= Least_Cost4;
    TRU4 := TRU4 + (resource4 * duration4);
    TAD4 := TAD4 + duration4;
    Total_Cost4  :=  Total_Cost4 + Optimal_Cost4 [N];
    writeln(fileD4,duration4:5:0,Optimal_Cost4[N]:10:2,RES[N]:6:0);
  end;

  Total_Cost4 := Total_Cost4;
  TAD4 := TAD4;
  TRU4 := TRU4;

  close (fileD4);

end; (end of Feasible)
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

4.4.3.5 Description Of Procedure ReadInfo

This procedure asks the user to input the number of events and the activities that form the project network. Then, it starts reading the information available in the file [fileD1.dat]. This information includes the duration (Dur [K]), total cost (COS [K]), and resource (RES [K]) of each activity. Next, the procedure checks if the user violates the i-j rule (i.e., the number of the start event (i) greater than the number of the finish events (j)). It also checks that all durations have positive or zero values.

EXHIBIT (4.8).--SOURCE CODE OF PROCEDURE READINFO

```
procedure ReadInfo1;
  var K: integer;
  begin
     repeat
       write('      How many events do you have ? > ');
       readln(events);
     until ((events in [2..maxnodes]) and (activities in [1..maxactivities]) and (activities >= events -1));
     writeln;
     write('      Please enter activities with their start event');
     writeln(' and finish event');
     assign (fileD1, 'c:prodD1.dat');
     reset (fileD1);
     while not eof (fileD1) do
      begin
       for K := 1 to activities do
        begin
         readln (fileD1, Dur[K], COS[K], RES[K]);
        end;
      end;

     for K := 1 to activities do
     repeat
        read(i[K], j[K]);
         if (i[K] >= j[K]) then
            writeln('      Start event must be less than finish event');
         if (Dur[K] < 0) then
            writeln('      Duration must be positive or zero');
     until (Dur[K] >= 0) and (i[K] < j[K]);
     close (fileD1);
   end (ReadInfo);
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

4.4.3.6 Description Of Procedure CPM

This procedure performs a comprehensive network analysis. It consists of two [For] loops and two [Repeat] loops. The purpose of the first [For] loop is to assign a zero value to the early start time and a larger number value to the late start time of each activity. This is required to initialize the EST and LST for each event. The first [Repeat] loop is designed to perform the forward pass calculations by determining the early start time of each activity. The second [Repeat] loop is designed to carry out the backward pass calculations. the second [For] loop performs the conventional CPM time analysis. Then, it uses this analysis to develop all activity indices (i.e. ACI, AINTI, AINDI, ACOI, ARI, and AGI). CM-DSS uses four CPM procedures, one for each scheduling alternative.

EXHIBIT (4.9).--SOURCE CODE OF PROCEDURE CPM

```
procedure CPM1;

 var K,P : integer;
 begin
    for P := 1 to events do
     begin
       EST[P] := 0;
       LST[P] := Largenumber
     end;
     K := 1;

    repeat
       EST[j[K]] := max(EST[j[K]], EST[i[K]] + Dur[K]);     {Fortward Pass}
       K:= K + 1
    until (K > activities);

    K := activities;
    LST[events] := EST[events];

    repeat
       LST[i[K]] := min(LST[i[K]], LST[j[K]] - Dur[K]);     {Backward Pass}
       K:= K - 1
    until (K < 1);
   TARI1    := 0;
   TACOI1   := 0;
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
    TOTAL1   := 0;
    TACI1    := 0;
    ACI1[1]  := 0;
    AGI1[1]  := 0;
    ACOI1[1] := 0;
    ARI1 [1] := 0;

    for K := 1 to activities do
      begin
        ES[K]    := EST[i[K]];
        EF[K]    := ES[K] + Dur[K];
        LF[K]    := LST[j[K]];
        LS[K]    := LF[K] - Dur[K];
        TF[K]    := LF[K] - ES[K] - Dur[K];
        INTF[K] := max(0, EST[j[K]] - LST[i[K]] - Dur[K]);
        INDF[K] := LST[j[K]] - LST[i[K]] - Dur[K];
        FF[K]    := EST[j[K]] - EST[i[K]] - Dur[K];
        if TF[K] = 0 then ACI1[K] := 100
        else ACI1[K]    := (1 - (TF[K])/(Dur[K] + TF[K])) * 100;
        if INTF[K] = 0 then AINTI1[K] := 100
        else AINTI1[K] := ( 1 - (INTF[K])/(Dur[K] + INTF[K])) * 100;
        if INDF[K] = 0 then AINDI1[K] := 100
        else AINDI1[K] := ( 1 - (INDF[K])/(Dur[K] + INDF[K])) * 100;
        TACI1    := TACI1 + ACI1[K];
        ACOI1[K] := ((COS[K])/(Total_Cost1)) * 100;
        TACOI1   := TACOI1 + ACOI1[K];
        ARI1[K]  := ((RES[K])/(TRU1)) * 100;
        TARI1    := TARI1 + ARI1[K];
        TOTAL1   := (ACI1[K] + ACOI1[K] + ARI1[K] + AINTI1[K] + AINDI1[K]);
        AGI1[K]  := ((TOTAL1)/(5));
      end;
        ProjectDur1 := EF[activities];
end {CPM1};
```

4.4.3.7 Description Of Procedure Path

The objective of this procedure is to save the analysis produced by each CPM procedure and write it in the E's files (i.e., fileE1.dat, fileE2.dat, fileE3.dat, and fileE4.dat). Procedures Path1, Path2, Path3, and Path4 are identical in their structure. Exhibit (4.10) shows procedure path2. It consists of one [For] loop that is designed to save the CPM analysis in file (fileE2.dat).

EXHIBIT (4.10).--SOURCE CODE OF PROCEDURE PATH

```
procedure path2;  {Save CPM Analysis, for the Time_Based Schedule, into file fileE2.dat}
  var K : integer;
  begin
    assign (fileE2,'c:prodE2.dat');
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
    rewrite(fileE2);
    writeln(fileE2,' ');
    writeln(fileE2,'***** CASE # 2 : THE TIME_BASED SCHEDULE *******');
    writeln(fileE2,' ');
    writeln(fileE2,'==========================================================================');
    write(fileE2,'Act.No',' i ', ' j ', ' Dur ', ' ES ', ' EF ',' LS ',' LF ',' TF ');
    writeln(fileE2,' FF ', ' INTF ', 'INDF ','ACI ',' AINTI ',' AINDI ','  ACOI',' ARI ', ' AGI');
    writeln(fileE2,'==========================================================================');
    writeln;
    for K := 1 to activities do
      begin
        write(fileE2,K:3,i[K]:4,j[K]:4,Dur[K]:4,ES[K]:4,EF[K]:4,LS[K]:4,LF[K]:4,TF[K]:4,FF[K]:4,INTF[K]:4);
       writeln(fileE2,INDF[K]:4,ACI2[K]:8:2,AINTI2[K]:8:2,AINDI2[K]:8:2,ACOI2[K]:6:2,ARI2[K]:6:2,AGI2[K]:8:2);
      end;
    writeln;
    writeln (fileE2,'==========================================================================');
    writeln (fileE2);
    writeln (fileE2,'1. The critical activities have Criticality Index of 100%');
    writeln (fileE2,'2. Total Cost                                      = $', Total_Cost2 :8:3);
    writeln (fileE2,'3. Schedule Duration "activities in parallel"      =  ', EF[activities],' hours');
    writeln (fileE2,'4. Schedule Duration "activities in series"        =  ', TAD2:5:0,     ' hours');
                    RES2 := TRU2;
    writeln (fileE2,'5. Total Resource Utilization                      =  ', RES2 :6:0,' man-hours');
                    NLI2 := ( 1 - (EF[activities])/(TAD2)) * 100;
    writeln (fileE2,'6. Network Logic Index                             =  ', NLI2 :4:2,' %');
                    SCI2 := ((TACI2)/(activities));
    writeln (fileE2,'7. Schedule Criticality Index                      =  ', SCI2 :4:2,' %');
                    SCOI2:= ((TACOI2)/(activities));
    writeln (fileE2,'8. Schedule Cost Index                             =  ', SCOI2:6:2,' %');
                    SRI2  := ((TARI2)/(activities));
    writeln (fileE2,'9. Schedule Resource Index                         =  ', SRI2 :6:2,' %');
                    Cost2 := Total_Cost2;
    close (fileE2);
end {path2};
```

4.4.4 THE LEARNING CURVE FEATURE

The learning curve feature is designed to perform specific tasks if the planner elects to take into account the learning curve effect. This segment of the program prompts the user to the question *"What is the execution order of activity ( ) ?"* The segment expects a number from 1 to 4. If the input, for instance, is 2, the program will understand that the execution order of this activity is 2. The program, then, requests the user to enter the learning rate value; *"Please enter the learning rate value (0.75, 0.80, or 0.85) >."* Based on the user's input, the program adjusts the original man-hours of this activity. This segment consists of

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

four [M-For] loops, and one [N-For] loop. Each [M-For] loop reads the contents of file (fileA.dat) and adjusts the man-hours values for each repetitive activity based on its order of execution. The purpose of the [N-For] loop is to repeat the [M-For] loops for each activity when it is necessary.

EXHIBIT (4.11).--SOURCE CODE OF THE LEARNING CURVE FEATURE

```
assign (fileA, 'c:proda.dat');
reset  (fileA);
assign (fileA1,'c:proda1.dat');
rewrite(fileA1);
writeln;
write('Do you want to consider the Learning Curve Effect : YES or NO ');
readln(answer1);
IF answer1 = 'YES' THEN  {... Start of if answer1...}
  begin
    writeln(' What is the learning rate (0.75, 0.80, or 0.85) ? > ');
    readln (rate);
    if rate = 0.75 then
    begin
     write('      How many activities do you have ? >');
     readln(activities);
     for N := 1 to activities do
      begin
        write(' What is the execution order of activity, 'N ' ? ');
        readln(number);
         if number = 1 then
         begin
             Factor := 1;
             for M := 1  to 4 do
             begin
                 readln(fileA, Act[N].MH[M], Act[N].PR[M], Act[N].CS[M], Act[N].UC[M]);
                 Act[N].MH[M] := Act[N].MH[M] * Factor;
                 writeln(fileA1,Act[N].MH[M], Act[N].PR[M], Act[N].CS[M], Act[N].UC[M]);
             end; {end of M loop}
           end
         else if number = 2 then
           begin
             Factor := 0.80;
             for M := 1  to 4 do
             begin
                 readln(fileA, Act[N].MH[M], Act[N].PR[M],
                 Act[N].CS[M],Act[N].UC[M]);
                 Act[N].MH[M] := Act[N].MH[M] * Factor;
                 writeln(fileA1,Act[N].MH[M], Act[N].PR[M], Act[N].CS[M], Act[N].UC[M]);
             end; {end of M loop}
           end
         else if number = 3 then
           begin
             Factor := 0.702;
             for M := 1  to 4 do
              begin
                 readln(fileA, Act[N].MH[M], Act[N].PR[M], Act[N].CS[M], Act[N].UC[M]);
                 Act[N].MH[M] := Act[N].MH[M] * Factor;
                 writeln(fileA1,Act[N].MH[M], Act[N].PR[M], Act[N].CS[M], Act[N].UC[M]);
              end; {end of M loop}
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
      end
    else if number = 4 then
      begin
        Factor := 0.640;
        for M := 1  to 4 do
        begin
            readln(fileA, Act[N].MH[M], Act[N].PR[M], Act[N].CS[M], Act[N].UC[M]);
            Act[N].MH[M] := Act[N].MH[M] * Factor;
            writeln(fileA1,Act[N].MH[M], Act[N].PR[M], Act[N].CS[M], Act[N].UC[M]);
        end; {end of M loop}
    end
end; {end of N loop}
```

4.4.5 THE VALUE MATRIX FEATURE

This part of the program is designed to facilitate What-If analysis for the planner. It helps him/her in evaluating the four scheduling alternatives by considering five control criteria. First the program prompts the user to the question *"Do you want to run a What-If analysis?"* If the answer is *"YES"*, the program will lead the user through eight prompts as shown in Exhibit (4.11). At each prompt, the user is asked to enter either a relative importance value or the upper limit against which each criterion is measured. This information is provided by the user based on experience. After manipulating the inputs, the program provides the user with an on-screen worth values for the four scheduling alternatives.

EXHIBIT (4.11).--SOURCE CODE OF THE VALUE MATRIX FEATURE

```
            {******* The Value Matrix Feature *******}

    clrscr;
    writeln('      Do you want to run a What-If analysis?  >  ')
    write('      Please respond by YES or NO ');
     readln(answer2);
     writeln(answer2);
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
if answer2 = 'YES' then
  begin
    value1 := 0;
    value2 := 0;
    value3 := 0;
    value4 := 0;

    write('      Please enter relative importance RI for DURATION >  ');
    readln (RI);
    writeln;
    write('      Please enter the contractual duration >  ');
    readln (CD);
    RV1 := (CD/ProjectDur1) * 10;
    RV2 := (CD/ProjectDur2) * 10;
    RV3 := (CD/ProjectDur3) * 10;
    RV4 := (CD/ProjectDur4) * 10;
    sub1 := RI * RV1;
    Value1 := sub1 + Value1;
    sub2 := RI * RV2;
    Value2 := sub2 + Value2;
    sub3 := RI * RV3;
    Value3 := sub3 + Value3;
    sub4 := RI * RV4;
    Value4 := sub4 + Value4;

    write('      Please enter relative importance RI for TOTAL COST >  ');
    readln (RI);
    writeln;
    write('      Please enter the project budget >  ');
    readln (BUDGET);
    RV1 := (BUDGET/Total_Cost1) * 10;
    RV2 := (BUDGET/Total_Cost2) * 10;
    RV3 := (BUDGET/Total_Cost3) * 10;
    RV4 := (BUDGET/Total_Cost4) * 10;
    sub1 := RI * RV1;
    Value1 := sub1 + Value1;
    sub2 := RI * RV2;
    Value2 := sub2 + Value2;
    sub3 := RI * RV3;
    Value3 := sub3 + Value3;
    sub4 := RI * RV4;
    Value4 := sub4 + Value4;

    write('      Please enter relative importance RI for RESOURCE UTILIZATION >  ');
    readln (RI);
    writeln;
    write('      Please enter the project MaxResource >  ');
    readln (MAXRESOURCE);
    RV1 := (MAXRESOURCE/TRU1) * 10;
    RV2 := (MAXRESOURCE/TRU2) * 10;
    RV3 := (MAXRESOURCE/TRU3) * 10;
    RV4 := (MAXRESOURCE/TRU4) * 10;
    sub1 := RI * RV1;
    Value1 := sub1 + Value1;
    sub2 := RI * RV2;
    Value2 := sub2 + Value2;
    sub3 := RI * RV3;
    Value3 := sub3 + Value3;
    sub4 := RI * RV4;
    Value4 := sub4 + Value4;

    write('      Please enter relative importance RI for NETWORK CRITICALITY INDEX >  ');
    readln (RI);
    writeln;
    RV1 := (1 - (NLI1)/(100)) * 10;
    RV2 := (1 - (NLI2)/(100)) * 10;
    RV3 := (1 - (NLI3)/(100)) * 10;
    RV4 := (1 - (NLI4)/(100)) * 10;
    sub1 := RI * RV1;
    Value1 := sub1 + Value1;
    sub2 := RI * RV2;
    Value2 := sub2 + Value2;
    sub3 := RI * RV3;
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
Value3 := sub3 + Value3;
sub4 := RI * RV4;
Value4 := sub4 + Value4;

write('      Please enter relative importance RI for SCHEDULE CRITICALITY INDEX >  ');
readln (RI);
writeln;
RV1 := (1 - (SCI1)/(100)) * 10;
RV2 := (1 - (SCI2)/(100)) * 10;
RV3 := (1 - (SCI3)/(100)) * 10;
RV4 := (1 - (SCI4)/(100)) * 10;
sub1 := RI * RV1;
Value1 := sub1 + Value1;
sub2 := RI * RV2;
Value2 := sub2 + Value2;
sub3 := RI * RV3;
Value3 := sub3 + Value3;
sub4 := RI * RV4;
Value4 := sub4 + Value4;

writeln;
    writeln ('      Value Of The Feasible Schedule      = ', Value1:7:3);
    writeln ('      Value Of The Cost_Based Schedule    = ', Value2:7:3);
    writeln ('      Value Of The Time_Based Schedule    = ', Value3:7:3);
    writeln ('      Value Of The Resource_Based Schedule = ', Value4:7:3);
    writeln;
    writeln ('      The schedule that has the highest value is the best.');
    writeln;
    writeln ('      **************************************************');
    writeln ('                THANK YOU FOR USING CM-DSS           ');
    writeln ('      **************************************************');
end
else
  begin
    writeln  ('      **************************************************');
    writeln  ('                THANK YOU FOR USING CM-DSS           ');
    writeln  ('      **************************************************');
  end;
end. ( end of if answer2)
```

4.5 DATA REQUIREMENTS

4.5.1 SYSTEM INPUTS

A typical computer session with CM-DSS requires several inputs. The user is usually requested to input data related to:

a. Productivity Database File [fileA1.dat]. This file--see Appendix (III)--is essential to run the CM-DSS program. It is used by the program to determine the time, resource, and cost values for each activity. It includes four columns and four rows for each activity.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

The first column contains optimal the man-hours required to perform each activity. The second column contains four productivity rates. The third column contains the crew size needed to perform the job. The fourth column contains the unit cost per unit of resource. In preparing the database, any dummy is treated as an activity with zero resource, duration, and cost values.

b. The Learning Curve Effect. If the user decides to take the effect of the learning curve into account, he/she should respond by entering the following information for each activity:

1. The learning rate value (75%, 80%, or 85%); and
2. The execution order of this activity (i.e., unit number 1, 2, 3, or 4).

c. Network Logic in (i-j) format. The user should enter the network logic by using the i-j format. He/she should enter these activities, including all dummies, in the same order available in file [fileA1.dat].

d. Activity Constraints (resource/time/budget). CM-DSS allows for one set of constraints per activity. The user should enter the modular in the following order: 1) Crew Size (Resource), 2) Adjusted-Hours (Time) , and 3) Total Cost (Budget). During the computer session, the program will provide the user with on-the-screen copy of file [fileC1.dat] to provide him/her with the possible constraints based on the original database. Accordingly,

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

he/she can select a realistic set of constraint for each activity to develop the feasible schedule. It is important to mention that the constraint to be used by the user should be one of the four sets available in the database.

f. Value Matrix. If the user does not accept the feasible schedule as a target schedule and decides to evaluate the four schedules, he/she should respond by *"YES"* to the program's question *"Do you want to run a What-If analysis?"* In this case, the user is requested to enter the following:

1. A relative importance RI value for each control criterion based on the planner's judgement.
2. The upper limit for each criterion--see Step (3.c). These limits are:
   a) the project alloted budget;
   b) the project contractual duration;
   c) the labor resource requirements estimated by the planner;
   d) for the SCHEDULE CRITICALITY INDEX and the NETWORK CRITICALITY INDEX, the planner is not required to enter upper limit values, since CM-DSS uses the percentage of each index as a direct measure.

4.5.2 SYSTEM OUTPUTS

a. Adjusted-Hours and Total Cost for each activity's modular. CM-DSS manipulates the available data in [fileA1.dat] and produces the Adjusted hours and Total Cost values for each activity. These

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

values can be found in files [fileC1.dat], [fileC2.dat], [fileC3.dat], and [fileC4.dat]. The user can get a hard copy from each file by typing the DOS command [C:\>PRINT filename] or importing each file into any word processing software and printing it as a text file.

b. Scheduling Alternatives' Analysis. The CM-DSS computer program develops four scheduling alternatives, namely: 1) the Cost-Based schedule; 2) the Time-Based schedule; 3) the Resource-Based schedule; and 4) the Feasible schedule. The program produces a comprehensive network analysis for each schedule. These analyses can always be found in files [fileE1.dat], [fileE2.dat], [fileE3.dat], and [fileE4.dat]. The user can obtain a hard copy from each file by using the same procedures previously explained to obtain the C files.

d. The Value Matrix Analysis. CM-DSS provides the user with four worth values, every time he/she evaluates the four scheduling alternatives. The schedule that has the highest worth value is considered to be the best or the most feasible schedule.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

CHAPTER FIVE

COMPUTER SESSION WITH CM-DSS

5.1 COMPUTER SESSION

In this chapter, the reader will be exposed to the CM-DSS features through a step-by-step computer session. This session is designed to demonstrate how a planner can use the CM-DSS program to study a project during either the planning phase or the control phase.

Supposing that the planner intends to study project XYZ under the following scenario:

1. The project consists of 21 activities.
2. Contractual Duration = 110 hours
3. The project allotted Budget = $50,000
4. Allowable Resource Utilization = 1450 man-hours
5. There is one group of repetitive activities that follows an 80% learning rate. This group contains activities #4, #11, and #15 which to be executed in the following order.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

6. The planner wants to study the schedule when activities #3, #10 and #20 have constraints. The planner will select the value of these constraints, during the computer session, when CM-DSS displays the modified values of time, resource, and cost on the screen.

5.2 INPUTS

Contractors usually establish historical records concerning their projects and classify the data in a preset manner. Their objective is to establish a database in which all activity-related information, such as productivity rate, cost rate, normal and optimal level of resource usage, environment effects, durations, etc., are organized in a suitable format. These are established mainly on the basis of two sources, past experience and records, and manuals or catalogues such as MEANS , SWEET CATALOGUES, etc. For the purpose of this illustration, a format similar to the one shown in exhibit (5.1) is used to prepare the project productivity-cost database.

Usually, the technological logic of a project is established by the planner after examining possible technologies and construction methods. The final network for project XYZ is shown in figure (5.1). Next, the planner determines the man-hours or man-days and the fundamental information needed for each activity. All the required information is shown in exhibit (5.1).

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

EXHIBIT (5.1).--PRODUCTIVITY-COST DATABASE

ACT	MH	PR	CS	UC
1	40	0.79	3	30
	40	0.88	4	30
	40	1.00	5	30
	40	0.92	6	30
2	90	0.86	2	28
	90	1.00	3	28
	90	0.84	4	28
	90	0.80	6	28
3	70	0.79	3	30
	70	0.87	4	30
	70	1.00	5	30
	70	0.85	7	30
4	80	0.60	4	30
	80	0.89	6	30
	80	1.00	8	30
	80	0.58	9	30
5	95	0.82	2	30
	95	0.90	3	30
	95	1.00	6	30
	95	0.88	4	30
6	72	0.78	2	35
	72	0.88	4	35
	72	1.00	6	35
	72	0.82	8	35
7	0	0	0	0
	0	0	0	0
	0	0	0	0
	0	0	0	0
8	56	0.85	4	45
	56	1.00	7	45
	56	0.90	6	45
	56	0.75	3	45
9	96	0.80	4	40
	96	1.00	8	40
	96	0.92	6	40
	96	0.72	2	40
10	84	0.79	2	40
	84	0.84	4	40
	84	1.00	6	40
	84	0.82	3	40
11	80	0.85	2	40
	80	1.00	5	40
	80	0.90	4	40
	80	0.75	10	40
12	64	0.89	3	35
	64	0.92	4	35
	64	1.00	8	35
	64	0.75	6	35
13	56	0.77	2	40
	56	1.00	4	40
	56	0.82	7	40
	56	0.79	6	40
14	72	0.62	4	40
	72	0.74	5	40
	72	1.00	6	40
	72	0.69	3	40
15	75	0.83	6	45
	75	1.00	9	45
	75	0.88	7	45
	75	0.79	4	45
16	72	0.79	5	45
	72	0.84	4	45
	72	1.00	6	45
	72	0.72	3	45
17	64	0.90	4	35
	64	1.00	8	35
	64	0.88	5	35
	64	0.75	3	35
18	80	0.76	6	35
	80	0.90	7	35
	80	1.00	5	35
	80	0.70	3	35
19	72	0.80	3	30
	72	0.84	4	30
	72	1.00	6	30
	72	0.82	5	30
20	56	0.76	3	25
	56	1.00	7	25
	56	0.86	4	25
	56	0.82	5	25
21	90	0.88	2	35
	90	0.92	3	35
	90	1.00	5	35
	90	0.90	4	35
22	96	0.74	3	25
	96	1.00	8	25
	96	0.88	4	25
	96	0.79	2	25

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

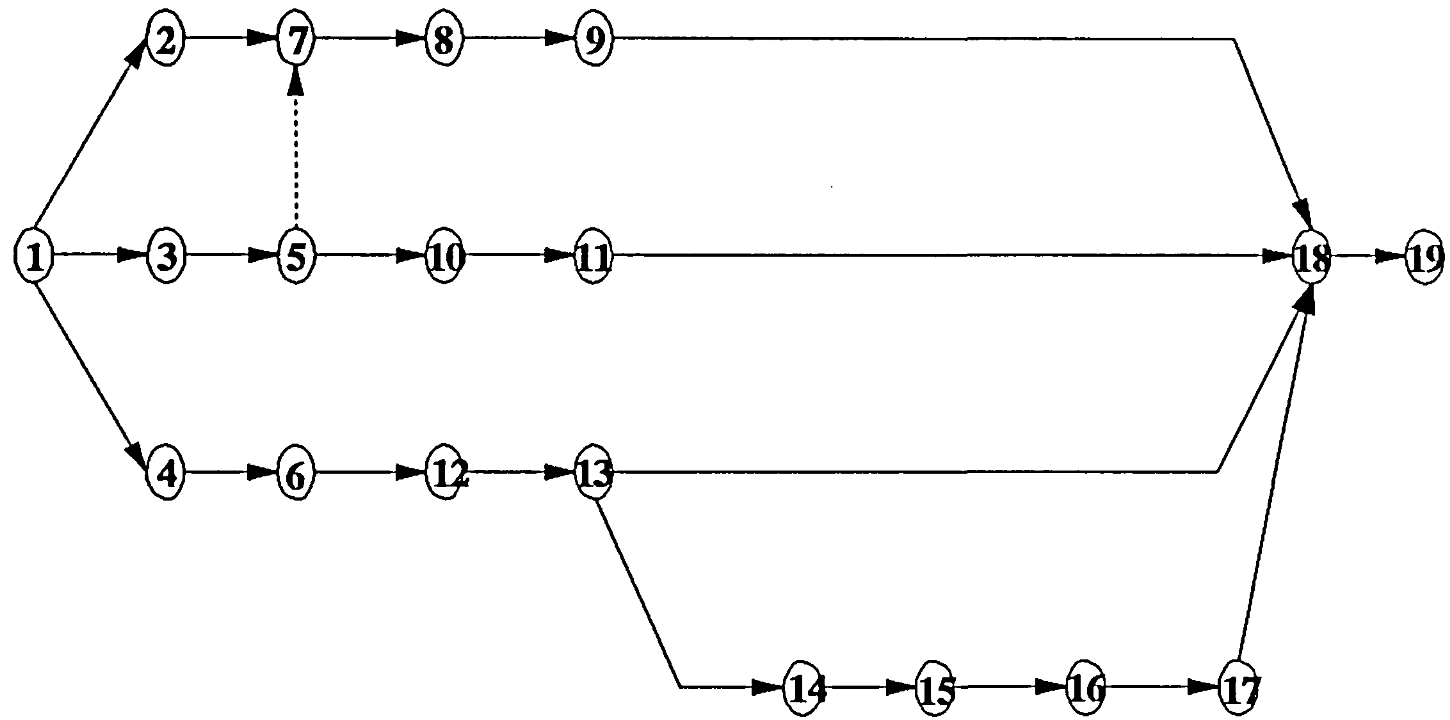


Figure (5.1).--XYZ Network

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

CM-DSS starts the session by introducing the user to its features. It prompts the user to an introductory screen as shown in exhibit (5.2).

EXHIBIT (5.2)

```
******************************************************************
*                                                                *
*                        WELCOME TO                              *
*                                                                *
*                      CM-DSS PLANNER                            *
*                                                                *
*                                                                *
* CM_DSS is a CPM-based program. It is designed to help          *
* the project manager and/or planner in planning an activity     *
* under variable productivity, time, resource, and budget        *
* constraints, and to develop four CPM-based scheduling          *
* alternatives.                                                  *
*                                                                *
* CM-DSS is capable of:                                          *
*                                                                *
*       1. considering one set of constraints per activity;      *
*       2. providing the project manager/planner with the        *
*          flexibility of selecting a target schedule among four *
*          alternatives by using a value matrix;                 *
*       3. allowing the project manager to run a WHAT-IF analysis*
*          to examine different scheduling scenarios;            *
*       4. generating CRITICALITY indices that help the project  *
*          manager in examining the criticality of each activity *
*          as well as the criticality of the network logic.      *
*                                                                *
*  PLEASE read the following before continuing with CM-DSS:      *
*                                                                *
*  1. Prepare and load the database file on the [C:\] drive.     *
*  2. Prepare the logic of your project in an (i-j) format.      *
*  3. Read carefully the set values for each activity, as        *
*     they appear on the screen, and choose a possible set       *
*     of constraints. CM-DSS will give you enough time to make   *
*     your selection.                                            *
*  4. If you want to select a target schedule other than the     *
*     default schedule provided by CM-DSS, PLEASE respond to the *
*     program prompts as they appear.                            *
*                                                                *
******************************************************************
*                                                                *
*       WE HOPE YOU HAVE A GOOD TIME WITH CM-DSS PLANNER         *
*                                                                *
******************************************************************
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Then, CM-DSS presents the user with the question *"DO YOU WANT TO CONSIDER THE LEARNING CURVE EFFECT?"* If the user's input is *"YES,"* CM-DSS will ask the user to respond two consecutive questions *"WHAT IS THE LEARNING RATE?"* AND *"HOW MANY ACTIVITIES DO YOU HAVE?."* If an activity is repetitive, then the user can enter one execution order from 1 to 4. In case of a non repetitive activity, the user should input a value of (1). This instructs the CM-DSS program to use the original man-hours available in the database without any modification. Exhibit (5.3) shows the user's inputs for project XYZ.

EXHIBIT (5.3)

```
DO YOU WANT TO CONSIDER THE LEARNING EFFECT? YES or NO >  YES

WHAT IS THE LEARNING RATE (0.75, 0.80, 0.85)          >  0.80

HOW MANY ACTIVITIES DO YOU HAVE ?                      >  22

What is the execution order of activity (1)  ?         >  1
What is the execution order of activity (2)  ?         >  1
What is the execution order of activity (3)  ?         >  1
What is the execution order of activity (4)  ?         >  1
What is the execution order of activity (5)  ?         >  1
What is the execution order of activity (6)  ?         >  1
What is the execution order of activity (7)  ?         >  1
What is the execution order of activity (8)  ?         >  1
What is the execution order of activity (9)  ?         >  1
What is the execution order of activity (10) ?         >  1
What is the execution order of activity (11) ?         >  2
What is the execution order of activity (12) ?         >  1
What is the execution order of activity (13) ?         >  1
What is the execution order of activity (14) ?         >  1
What is the execution order of activity (15) ?         >  3
What is the execution order of activity (16) ?         >  1
What is the execution order of activity (17) ?         >  1
What is the execution order of activity (18) ?         >  1
What is the execution order of activity (19) ?         >  1
What is the execution order of activity (20) ?         >  1
What is the execution order of activity (21) ?         >  1
What is the execution order of activity (22) ?         >  1
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Next, the user is prompted for information about the technological logic of the project under study. After answering the questions *"HOW MANY ACTIVITIES DO YOU HAVE?"*, and *"HOW MANY EVENTS DO YOU HAVE?,"* the planner inputs the network logic in an i-j format. This is done by entering the number of the start event followed by the number of the finish event for each activity. During this interaction, the program manipulates the information available in the database and assigns part of this data to each activity. Exhibit (5.4) shows the user's input for project XYZ.

EXHIBIT (5.4)

```
******************************
PLEASE RESPOND TO EACH PROMPT
******************************

HOW MANY ACTIVITIES DO YOU HAVE?  >   22

HOW MANY EVENTS DO YOU HAVE? >  19

Please enter activities with their start event and finish event.

Activity (1)          1      2
Activity (2)          1      3
Activity (3)          1      4
Activity (4)          2      7
Activity (5)          3      5
Activity (6)          4      6
Activity (7)          5      7
Activity (8)          7      8
Activity (9)          8      9
Activity (10)         9      18
Activity (11)         5      10
Activity (12)         10     11
Activity (13)         11     18
Activity (14)         6      12
Activity (15)         12     13
Activity (16)         13     18
Activity (17)         13     14
Activity (18)         14     15
Activity (19)         15     16
Activity (20)         16     17
Activity (21)         17     18
Activity (22)         18     19
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

To develop a feasible schedule, CM-DSS presents the user with an on-screen listing of possible modulars of constraints for each activity--see file (fileC1.dat) Appendix (III). In this case, the planner can select one modular of constraint per activity, if any, and enter the modular in the following order (resource, time , and budget). In case of project XYZ, the user's input is shown in exhibit (5.5).

EXHIBIT (5.5)

```
Please enter one set of constraints for activity no. 1
Resource, Time, & Budget                             > 0 0 0
Please enter one set of constraints for activity no. 2
Resource, Time, & Budget                             > 0 0 0
Please enter one set of constraints for activity no. 3
Resource, Time, & Budget                             > 3 16.88 1518.99
Please enter one set of constraints for activity no. 4
Resource, Time, & Budget                             > 0 0 0
Please enter one set of constraints for activity no. 5
Resource, Time, & Budget                             > 0 0 0
Please enter one set of constraints for activity no. 6
Resource, Time, & Budget                             > 0 0 0
Please enter one set of constraints for activity no. 7
Resource, Time, & Budget                             > 0 0 0
Please enter one setr of constraints for activity no.8
Resource, Time, & Budget                             > 0 0 0
Please enter one set of constraints for activity no. 9
Resource, Time, & Budget                             > 0 0 0
Please enter one set of constraints for activity no. 10
Resource, Time, & Budget                             > 4 25.00 4000.00
Please enter one set of constraints for activity no. 11
Resource, Time, & Budget                             > 0 0 0
Please enter one set of constraints for activity no. 12
Resource, Time, & Budget                             > 0 0 0
Please enter one set of constraints for activity no. 13
Resource, Time, & Budget                             > 0 0 0
Please enter one set of constraints for activity no. 14
Resource, Time, & Budget                             > 0 0 0
Please enter one set of constraints for activity no. 15
Resource, Time, & Budget                             > 0 0 0
Please enter one set of constraints for activity no. 16
Resource, Time, & Budget                             > 0 0 0
Please enter one set of constraints for activity no. 17
Resource, Time, & Budget                             > 0 0 0
Please enter one set of constraints for activity no. 18
Resource, Time, & Budget                             > 0 0 0
Please enter one set of constraints for activity no. 19
Resource, Time, & Budget                             > 0 0 0
Please enter one set of constraints for activity no. 20
Resource, Time, & Budget                             > 3 24.56 1842.11
Please enter one set of constraints for activity no. 21
Resource, Time, & Budget                             > 0 0 0
Please enter one set of constraints for activity no. 22
Resource, Time, & Budget                             > 0 0 0
```

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Typically, conventional CPM as well as available scheduling software provide the planner with a time-oriented scheduling alternative. This forces the planner to use one type of scheduling alternatives. For example, if the planner wants to examine another alternative, he/she may elect to change the durations of some activities and to recalculate the CPM schedule. This approach of selecting a target schedule is neither realistic nor practical. It also consumes time and effort. Therefore, the CM-DSS system was designed to help the planner in studying four scheduling alternatives.

To evaluate the four scheduling alternatives, the user can use the What-If feature provided by CM-DSS. The program prompts the user to the question *"DO YOU WANT TO RUN A WHAT-IF ANALYSIS?"* If the user's answer is *"NO"*, CM-DSS will stop and terminate the session. If the user's answer is *"YES,"* CM-DSS will request him/her to input information about the relative importance of each control criterion and the standard against which each criterion is measured. If the user is not satisfied with the results and wants to examine another scenario by changing the importance of the control criteria, he/she can respond to the question *"DO YOU WANT TO RUN A WHAT-IF ANALYSIS?"* by *"YES."* For project XYZ, exhibit (5.6) shows the user's input.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

EXHIBIT (5.6)

```
*********************************
THIS IS THE VALUE MATRIX FEATURE
*********************************

DO YOU WANT TO RUN WHAT-IF ANALYSIS?  YES or NO                      > YES
Please enter the relative importance RI of DURATION (%)              > 25
Please enter the contractual duration in hours                       > 110
Please enter the relative importance RI of TOTAL COST (%)            > 25
Please enter the estimated budget in dollars                         > 50000
Please enter the relative importance RI of RESOURCE UTILIZATION (%)  > 20
Please enter the maximum allowable man-hours                         > 1450
Please enter the relative importance RI of NETWORK LOGIC INDEX (%)   > 15
Please enter the relative importance RI of SCHEDULE CRITICALITY INDEX (%) > 15
DO YOU WANT TO RUN WHAT-IF ANALYSIS?  YES or NO                      > NO
```

5.3 OUTPUTS

After manipulating all the input data, CM-DSS generates a comprehensive analysis for the project under study. It develops one report for each scheduling alternative. A typical report consists of two sections. The first section contains information related to the project's activities. This information includes the conventional CPM analysis as well as all the activity indices. The second section contains information related to the scheduling alternative and the original network logic. In case of project XYZ, the four reports are shown in exhibits (5.7), (5.8), (5.9), and (5.10).

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Having established four scheduling alternatives for a project, a planner can immediately compare the results of these schedules. By comparing the four reports, it can be seen that, as expected, the Cost_Based schedule is the least expensive among the four alternatives ($7,092). This schedule along with the Time_Based schedule require the least duration (51 hours). The Cost_Based schedule is also the best in terms of man-hours consumption (685 man-hours). However, this schedule has the highest Network Logic Index, and Schedule Criticality Index, with 57.95% and 60.66%., respectively. To incorporate the effect of several control criteria into the selection process, CM-DSS offers a practical approach for selecting the best schedule among these alternatives. It uses the worth matrix to help the planner in selecting a target schedule.

After considering the user's inputs and evaluating the importance of the scheduling alternatives, CM-DSS displays the results of the worth matrix analysis on the screen. For project XYZ, the results are shown in exhibit (5.11). The Feasible schedule is considered to be the best schedule under the preset control criteria, with a worth value of 872.733. At this stage, the CM-DSS program terminates the session.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

EXHIBIT (5.8).--NETWORK ANALYSIS FOR THE COST-BASED SCHEDULE

***** CASE # 1 : THE COST_BASED SCHEDULE *******

Act.No	i	j	Dur	ES	EF	LS	LF	TF	FF	INTF	INDF	ACI	AINTI	AINDI	ACOI	ARI	AGI
1	1	2	8	0	8	54	62	54	0	54	0	12.90	87.10	100.00	2.26	2.63	40.98
2	1	3	30	0	30	25	55	25	0	25	0	54.55	45.45	100.00	4.75	5.91	42.13
3	1	4	14	0	14	0	14	0	0	0	0	100.00	0.00	100.00	3.96	4.60	41.71
4	2	7	10	8	18	62	72	54	28	26	0	15.63	72.22	100.00	4.53	5.26	39.53
5	3	5	16	30	46	55	71	25	0	25	0	39.02	60.98	100.00	5.38	6.31	42.34
6	4	6	12	14	26	14	26	0	0	0	0	100.00	0.00	100.00	4.75	4.73	41.90
7	5	7	0	46	46	72	72	26	0	26	0	0.00	100.00	100.00	0.00	0.00	40.00
8	5	10	13	46	59	71	84	25	0	25	0	34.21	65.79	100.00	4.75	5.98	42.15
9	6	12	12	26	38	26	38	0	0	0	0	100.00	0.00	100.00	7.24	6.31	42.71
10	7	8	8	46	54	72	80	26	0	26	0	23.53	76.47	100.00	6.34	3.15	41.90
11	8	9	12	54	66	80	92	26	0	26	0	31.58	68.42	100.00	4.83	3.94	41.75
12	9	18	14	66	80	92	106	26	26	0	0	35.00	0.00	100.00	4.23	7.36	29.32
13	10	11	8	59	67	84	92	25	0	25	0	24.24	75.76	100.00	4.23	2.10	41.27
14	11	18	14	67	81	92	106	25	25	0	0	35.90	0.00	100.00	5.43	5.52	29.37
15	12	13	6	38	44	38	44	0	0	0	0	100.00	0.00	100.00	4.50	3.55	41.61
16	13	18	12	44	56	94	106	50	50	0	50	19.35	0.00	19.35	6.11	4.73	9.91
17	13	14	8	44	52	44	52	0	0	0	0	100.00	0.00	100.00	4.23	4.21	41.69
18	14	15	16	52	68	52	68	0	0	0	0	100.00	0.00	100.00	5.28	5.26	42.11
19	15	16	12	68	80	68	80	0	0	0	0	100.00	0.00	100.00	4.08	4.73	41.76
20	16	17	8	80	88	80	88	0	0	0	0	100.00	0.00	100.00	2.64	3.68	41.26
21	17	18	18	88	106	88	106	0	0	0	0	100.00	0.00	100.00	5.94	5.91	42.37
22	18	19	12	106	118	106	118	0	0	0	0	100.00	0.00	100.00	4.53	6.31	42.17

1. The critical activities have Criticality Index of 100%
2. Total Cost = $53005.000
3. Schedule Duration "activities in parallel" = 118 hours
4. Schedule Duration "activities in series" = 263 hours
5. Total Resource Utilization = 1522 man-hours
6. Network Logic Index = 55.05 %
7. Schedule Time Index = 60.27 %
8. Schedule Cost Index = $2409.32 /activity
9. Schedule Resource Index = 70.68units of resource/activity

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

EXHIBIT (5.9).--NETWORK ANALYSIS FOR THE TIME-BASED SCHEDULE

***** CASE # 2 : THE TIME_BASED SCHEDULE *******

Act.No	i	j	Dur	ES	EF	LS	LF	TF	FF	INTF	INDF	ACI	AINTI	AINDI	ACOI	ARI	AGI
1	1	2	7	0	7	52	59	52	0	52	0	11.86	88.14	100.00	2.32	2.59	40.98
2	1	3	19	0	19	34	53	34	0	34	0	35.85	64.15	100.00	5.59	7.04	42.53
3	1	4	12	0	12	0	12	0	0	0	0	100.00	0.00	100.00	4.39	5.19	41.92
4	2	7	10	7	17	59	69	52	18	34	0	16.13	77.27	100.00	4.26	4.94	40.52
5	3	5	16	19	35	53	69	34	0	34	0	32.00	68.00	100.00	5.06	5.93	42.20
6	4	6	11	12	23	12	23	0	0	0	0	100.00	0.00	100.00	5.46	5.44	42.18
7	5	7	0	35	35	69	69	34	0	34	0	0.00	100.00	100.00	0.00	0.00	40.00
8	5	10	8	35	43	70	78	35	0	35	0	18.60	81.40	100.00	4.47	3.46	41.59
9	6	12	12	23	35	23	35	0	0	0	0	100.00	0.00	100.00	6.82	5.93	42.55
10	7	8	14	35	49	69	83	34	0	34	0	29.17	70.83	100.00	5.97	5.19	42.23
11	8	9	9	49	58	83	92	34	0	34	0	20.93	79.07	100.00	6.06	5.56	42.32
12	9	18	8	58	66	92	100	34	34	0	0	19.05	0.00	100.00	3.98	3.95	25.40
13	10	11	10	43	53	78	88	35	0	35	0	22.22	77.78	100.00	4.85	4.32	41.83
14	11	18	12	53	65	88	100	35	35	0	0	25.53	0.00	100.00	5.11	4.45	27.02
15	12	13	6	35	41	35	41	0	0	0	0	100.00	0.00	100.00	4.23	3.34	41.51
16	13	18	12	41	53	88	100	47	47	0	47	20.34	0.00	20.34	5.75	4.45	10.18
17	13	14	8	41	49	41	49	0	0	0	0	100.00	0.00	100.00	3.98	3.95	41.59
18	14	15	13	49	62	49	62	0	0	0	0	100.00	0.00	100.00	5.52	5.62	42.23
19	15	16	12	62	74	62	74	0	0	0	0	100.00	0.00	100.00	3.84	4.45	41.66
20	16	17	8	74	82	74	82	0	0	0	0	100.00	0.00	100.00	2.49	3.46	41.19
21	17	18	18	82	100	82	100	0	0	0	0	100.00	0.00	100.00	5.59	5.56	42.23
22	18	19	12	100	112	100	112	0	0	0	0	100.00	0.00	100.00	4.26	5.93	42.04

1. The critical activities have Criticality Index of 100%
2. Total Cost = $56319.260
3. Schedule Duration "activities in parallel" = 112 hours
4. Schedule Duration "activities in series" = 235 hours
5. Total Resource Utilization = 1619 man-hours
6. Network Logic Index = 52.43 %
7. Schedule Time Index = 56.89 %
8. Schedule Cost Index = $2559.97/activity
9. Schedule Resource Index = 74.14units of resource/activity

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

EXHIBIT (5.10).--NETWORK ANALYSIS FOR THE RESOURCE-BASED SCHEDULE

***** CASE # 3 : THE RESOURCE_BASED SCHEDULE *******

Act.No	i	j	Dur	ES	EF	LS	LF	TF	FF	INTF	INDF	ACI	AINTI	AINDI	ACOI	ARI	AGI
1	1	2	17	0	17	167	184	167	0	167	0	9.24	90.76	100.00	2.20	2.58	40.96
2	1	3	52	0	52	107	159	107	0	107	0	32.70	67.30	100.00	4.24	5.26	41.90
3	1	4	30	0	30	0	30	0	0	0	0	100.00	0.00	100.00	3.85	4.55	41.68
4	2	7	33	17	50	184	217	167	60	107	0	16.50	76.43	100.00	5.79	6.67	41.08
5	3	5	58	52	110	159	217	107	0	107	0	35.15	64.85	100.00	5.03	5.86	42.18
6	4	6	46	30	76	30	76	0	0	0	0	100.00	0.00	100.00	4.68	4.65	41.87
7	5	7	0	110	110	217	217	107	0	107	0	0.00	100.00	100.00	0.00	0.00	40.00
8	5	10	25	110	135	236	261	126	0	126	0	16.56	83.44	100.00	4.87	3.79	41.73
9	6	12	67	76	143	76	143	0	0	0	0	100.00	0.00	100.00	7.73	6.77	42.90
10	7	8	53	110	163	217	270	107	0	107	0	33.13	66.88	100.00	6.16	5.36	42.30
11	8	9	38	163	201	270	308	107	0	107	0	26.21	73.79	100.00	4.36	3.84	41.64
12	9	18	24	201	225	308	332	107	107	0	0	18.32	0.00	100.00	3.65	3.64	25.12
13	10	11	36	135	171	261	297	126	0	126	0	22.22	77.78	100.00	4.21	3.64	41.57
14	11	18	35	171	206	297	332	126	126	0	0	21.74	0.00	100.00	6.05	5.31	26.62
15	12	13	17	143	160	143	160	0	0	0	0	100.00	0.00	100.00	4.37	3.44	41.56
16	13	18	33	160	193	299	332	139	139	0	139	19.19	0.00	19.19	6.52	5.00	9.98
17	13	14	28	160	188	160	188	0	0	0	0	100.00	0.00	100.00	4.33	4.25	41.71
18	14	15	38	188	226	188	226	0	0	0	0	100.00	0.00	100.00	5.79	5.76	42.31
19	15	16	30	226	256	226	256	0	0	0	0	100.00	0.00	100.00	3.91	4.55	41.69
20	16	17	25	256	281	256	281	0	0	0	0	100.00	0.00	100.00	2.67	3.79	41.29
21	17	18	51	281	332	281	332	0	0	0	0	100.00	0.00	100.00	5.18	5.16	42.07
22	18	19	61	332	393	332	393	0	0	0	0	100.00	0.00	100.00	4.40	6.17	42.11

1. The critical activities have Criticality Index of 100%
2. Total Cost = $69037.220
3. Schedule Duration "activities in parallel" = 393 hours
4. Schedule Duration "activities in series" = 797 hours
5. Total Resource Utilization = 1978 man-hours
6. Network Logic Index = 50.67 %
7. Schedule Time Index = 56.86 %
8. Schedule Cost Index = $3138.06 /activity
9. Schedule Resource Index = 89.95 units of resource/activity

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

EXHIBIT (5.11).--NETWORK ANALYSIS FOR THE FEASIBLE SCHEDULE

***** CASE # 4 : THE FEASIBLE SCHEDULE *******

Act.No	i	j	Dur	ES	EF	LS	LF	TF	FF	INTF	INDF	ACI	AINTI	AINDI	ACOI	ARI	AGI
1	1	2	8	0	8	62	70	62	0	62	0	11.43	88.57	100.00	2.24	2.60	40.97
2	1	3	30	0	30	34	64	34	0	34	0	46.88	53.13	100.00	4.71	5.86	42.11
3	1	4	17	0	17	0	17	0	0	0	0	100.00	100.00	100.00	2.84	3.32	61.23
4	2	7	10	8	18	70	80	62	28	34	0	13.89	77.27	100.00	4.49	5.21	40.17
5	3	5	16	30	46	64	80	34	0	34	0	32.00	68.00	100.00	5.33	6.25	42.32
6	4	6	12	17	29	17	29	0	0	0	0	100.00	100.00	100.00	4.71	4.69	61.88
7	5	7	0	46	46	80	80	34	0	34	0	0.00	100.00	100.00	0.00	0.00	40.00
8	5	10	8	46	54	92	100	46	0	46	0	14.81	85.19	100.00	4.71	3.65	41.67
9	6	12	12	29	41	29	41	0	0	0	0	100.00	100.00	100.00	7.18	6.25	62.69
10	7	8	25	46	71	80	105	34	0	34	0	42.37	57.63	100.00	7.48	6.51	42.80
11	8	9	13	71	84	105	118	34	0	34	0	27.66	72.34	100.00	4.78	4.23	41.80
12	9	18	8	84	92	118	126	34	34	0	0	19.05	100.00	100.00	4.19	4.17	45.48
13	10	11	14	54	68	100	114	46	0	46	0	23.33	76.67	100.00	4.19	3.65	41.57
14	11	18	12	68	80	114	126	46	46	0	0	20.69	100.00	100.00	5.38	4.69	46.15
15	12	13	6	41	47	41	47	0	0	0	0	100.00	100.00	100.00	4.46	3.51	61.59
16	13	18	12	47	59	114	126	67	67	0	67	15.19	100.00	15.19	6.06	4.69	28.22
17	13	14	8	47	55	47	55	0	0	0	0	100.00	100.00	100.00	4.19	4.17	61.67
18	14	15	16	55	71	55	71	0	0	0	0	100.00	100.00	100.00	5.23	5.21	62.09
19	15	16	12	71	83	71	83	0	0	0	0	100.00	100.00	100.00	4.04	4.69	61.74
20	16	17	25	83	108	83	108	0	0	0	0	100.00	100.00	100.00	3.44	4.88	61.66
21	17	18	18	108	126	108	126	0	0	0	0	100.00	100.00	100.00	5.89	5.86	62.35
22	18	19	12	126	138	126	138	0	0	0	0	100.00	100.00	100.00	4.49	6.25	62.15

1. The critical activities have Criticality Index of 100%
2. Total Cost = $53506.100
3. Schedule Duration "activities in parallel" = 138 hours
4. Schedule Duration "activities in series" = 293 hours
5. Total Resource Utilization = 1536 man-hours
6. Network Logic Index = 52.89 %
7. Schedule Time Index = 57.60 %
8. Schedule Cost Index = $2432.10 /activity
9. Schedule Resource Index = 70.05 units of resource/activity

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

EXHIBIT (5.11).--THE ON-SCREEN WHAT-IF ANALYSIS

```
********************************
RESULTS OF THE WHAT-IF ANALYSIS
********************************

VALUE OF THE FEASIBLE SCHEDULE         = 786.438

VALUE OF THE COST_BASED SCHEDULE       = 782.655

VALUE OF THE TIME_BASED SCHEDULE       = 536.333

VALUE OF THE RESOURCE_BASED SCHEDULE   = 755.909

THE SCHEDULE THAT HAS THE HIGHEST VALUE IS THE BEST

***************************
THANK YOU FOR USING CM-DSS
***************************
```

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

5.4 ANALYSIS

After conducting a What-If analysis the planner will choose a target schedule. Based on that choice, CM-DSS calculates the Network Logic Index to reflect the complexity of the project network logic. If the planner is not satisfied with the Network Logic Index, and decides to replan the technological logic of the project at hand. In this case, CM-DSS provides the planner with valuable information that can be used in replanning the project. The planner should concentrate on the activities that have, for instance, the highest AGI values (say greater than 70%) and replan them. Replanning allows the user to study different construction methods, to establish new activity modulars. In case of project XYZ, the planner is satisfied with the Network Logic Index (52.48%). as a result, the Feasible schedule is selected as a target schedule.

To forecast possible deviation from the target schedule, the planner can use the four reports generated by CM-DSS. He/she can prepare a matrix similar to the one shown in table (5.1). In this matrix, only those activities that have ACI, AINTI, and AINDI values of at least 80% are considered. While those activities that have ACOI and ARI values of greater than 6% are considered. Also, each activity that has an AGI value of greater than 50% is considered. According to the performance matrix, any time an

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

activity appears under one of the indices, it will get a score of (1) point. For instance, activity #12 appears 9 times, and thus it has a score of (9). During the control phase, any activity (critical or noncritical) may deviate from the target schedule and, accordingly, cause project delay or slippage. This matrix provides the planner with a forecasting tool that can determine the activities that have a high tendency to become critical, regardless of the scheduling alternative that the planner may choose. Assume that the activities with a score greater than (6) are considered to be candidates for future slippage. Of course, if the planner feels confident about the control system, he/she may increase this score and concentrate on a smaller number of activities.

According to conventional CPM, the project manager is usually concerned with the critical activities during the construction phase. For project XYZ, the project manager will be concerned with activities, #3, #6, #9, #15, #17, #18, #19, #20, #21, and #22. However, by using the performance matrix, the planner should also monitor the noncritical activities that have the potential to become critical and may cause project slippage and financial losses. In case of project XYZ, the planner should also focus on activities #2, #5, #10, #11, #14, and #16.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table(5.1).--ACTIVITY PERFORMANCE MATRIX FOR PROJECT XYZ

Index / ACTIVITY	COST-BASED SCHEDULE						TIME-BASED SCHEDULE						RESOURCE-BASED SCHEDULE						FEASIBLE SCHEDULE						Total Score
	ACI	AINT	AIND	ACO	ARI	AGI	ACI	AINT	AIND	ACO	ARI	AGI	ACI	AINT	AIND	ACO	ARI	AGI	ACI	AINT	AIND	ACO	ARI	AGI	
1			●						●						●						●				4
2			●	●	●				●		●				●						●	●			8
3	●	●	●				●	●	●			●	●	●	●			●	●	●	●			●	14
4			●						●						●						●				5
5			●	●	●				●		●				●						●	●	●		9
6	●	●	●			●	●	●	●			●	●	●	●			●	●	●	●			●	16
7			●						●						●						●				4
8			●					●	●						●						●				5
9	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	24
10	●		●	●					●						●	●					●	●	●		9
11	●		●					●	●	●					●						●				7
12		●	●		●			●	●					●	●					●	●				9
13			●						●						●						●				4
14		●	●					●	●					●	●	●				●	●				9
15	●	●	●			●	●	●	●			●	●	●	●			●	●	●	●			●	16
16		●		●		●		●		●				●		●				●		●			9
17	●	●	●			●	●	●	●			●	●	●	●			●	●	●	●			●	16
18	●	●	●			●	●	●	●			●	●	●	●			●	●	●	●			●	16
19	●	●	●			●	●	●	●			●	●	●	●			●	●	●	●			●	16
20	●	●	●			●	●	●	●			●	●	●	●			●	●	●	●			●	16
21	●	●	●			●	●	●	●			●	●	●	●			●	●	●	●	●		●	17
22	●	●	●		●	●	●	●	●		●	●	●	●	●		●	●	●	●	●	●	●	●	21

CHAPTER SIX

SUMMARY AND CONCLUSIONS

6.1 SUMMARY

In recent years, CPM has been criticized for its inability to cope with the dynamic nature and variability of construction projects. Basically, CPM's capabilities are constrained by assuming the availability of unlimited resources, a deterministic network structure, and a fixed duration for each activity. A recent survey has shown that construction companies are not comfortable with CPM because it is laborious and time-consuming, expensive to update, impractical for field application, and not flexible.

CPM-based scheduling programs have been developed to facilitate scheduling practices for construction managers. They perform CPM computations, and provide these managers with graphical and tabular representations of project scheduling information. These capabilities enhance and facilitate the effectiveness of scheduling practices. In addition, these programs are capable of presenting the network schedule in a time-scaled diagram, and/or a bar chart. This helps project managers to report job progress and other related information to top management as well as owners. Moreover, many programs allow the user to plan the utilization of

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

resources in a project by generating a crew schedule, and material and equipment reports.

A sound scheduling technique cannot be only involved with determining a critical path from deterministic activity durations and following the execution of the critical activities. This technique should employ productivity rates, crew size and cost as basic production data to plan each network activity before applying CPM to the network. This will provide the planner with the flexibility to study different schedules in order to choose the most feasible one. In addition, this technique should allow the planner, to consider different types of constraints such as time, resources, and budget. CPM as well as CPM-based scheduling software lack this integrity.

This dissertation presents a computerized multi-dimensional support system (CM-DSS) for planning, scheduling, and controlling construction projects. CM-DSS is a CPM-based system. It is developed to help the project manager and/or planner in planning and activity under variable productivity, time, resource, and budget constraints. It generates four CPM-based scheduling alternatives. It also provides the project manager with the flexibility to select a target schedule among these alternatives by using the value matrix, and to examine different scheduling scenarios by conducting What-If analyses. CM-DSS provides the planner/construction manager, with three groups of criticality

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

indices to forecast the future performance and criticality of the activities as well as project under study.

As part of this research, a questionnaire was conducted to acquire information on the planning and control practices of U.S. construction companies. The focus of this study was to verify the methods and techniques that are used by the construction industry to plan and control projects. The results and findings were used to validate the criticism that CPM has received in recent years and to address the basic needs of the construction industry by developing a more comprehensive tool.

Construction companies have demonstrated dissatisfaction with the use of statistical methods, resource allocation techniques, and Learning Curves in controlling their projects. Consequently, they suggested that planning and control methods should be practical, flexible, integral and simple enough to understood by the field staff. While planning projects, construction experts regard several factors as a prerequisite element for project success. They believe that successful planning practices can be guaranteed if resources are carefully planned. When dealing with the execution and control of projects, experts prefer to establish control criteria to guide the project management team. They also suggested that these criteria should be reviewed and adjusted to accommodate any changes during execution of the project. These findings were used to design

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

and develop the CM-DSS system which was computerized by using Pascal as a programming language.

6.2 CONTRIBUTIONS OF CM-DSS

The major contribution of this research is the development of the CM-DSS algorithm and computer program whic is a comprehensive and realistic mechanism for planning, scheduling and controlling construction projects. CM-DSS is a CPM-based system. It integrates a productivity-based cost database for different size crews, three groups of criticality indices, and a decision support system. It can be used during either the planning phase or the control phase. In the planning phase, a planner interacts with CM-DSS by defining the learning rate for some repetitive activities and entering the project activities, their interdependencies, and the estimated man-hours for each activity. The program then adjusts these values based on the optimal productivity rate available in the database. At this stage, the user can specify one set of constraints for each activity. Consequently, CM-DSS develops four scheduling alternatives. These alternatives are: (1) a feasible schedule, (2) a cost-based schedule, (3) a time-based schedule, and (4) a resource-based schedule. Then, it analyzes the project technological logic and provides the planner with a logic index that reflects the criticality of the project network. If the planner is not satisfied with the logic index value and wishes to change the logic of the project, CM-DSS provides him/her with a

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

comprehensive analysis for the project's activities and recommends which activities should be considered for replanning. After modifying the logic, CM-DSS uses a decision support system along with the criticality indices to help the planner in evaluating the four schedules and selecting a target schedule. When the planner selects the target schedule, CM-DSS generates a comprehensive schedule analysis. This analysis can be used to forecast the potential of each activity to deviate from the target schedule.

Unlike CPM, CM-DSS treats all project activities as critical but with different degrees of criticality. By using the degree of criticality, it is capable of distinguishing between the noncritical activities as defined by conventional CPM.

CM-DSS introduces the concept of network logic criticality. It provides the planner with a simple measure that reflects the criticality or density of the network logic. This allows the planner to evaluate the criticality of the network logic and replan the logic if necessary. This index can also be used as a measure for the frequency of network review and updating.

Several major advantages are realized from this system:

1. CM-DSS can be used by planners and construction managers to plan an activity under variable productivity rates, and sets of constraints.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

2. CM-DSS is capable of developing four different scheduling alternatives;

3. The user can use CM-DSS to examine different scheduling scenarios and to select a target schedule.

4. CM-DSS can be used to tune up the logic of the target schedule by running What-If analyses. This allows the planner to select a network logic with an acceptable degree of criticality in order to reduce the possibility of future project slippage;

5. CM-DSS also allows the planner to incorporate control criteria into the project schedule during the planning process;

6. It provides a project manager with a comprehensive analysis that allows him/her to effectively monitor the execution of project activities and to prevent some of them from becoming critical by taking corrective actions ahead of time;

7. The Network Logic Index can be used to determine the frequency of network schedule review and updating;and

8. CM-DSS can be used during both the planning and control phases.

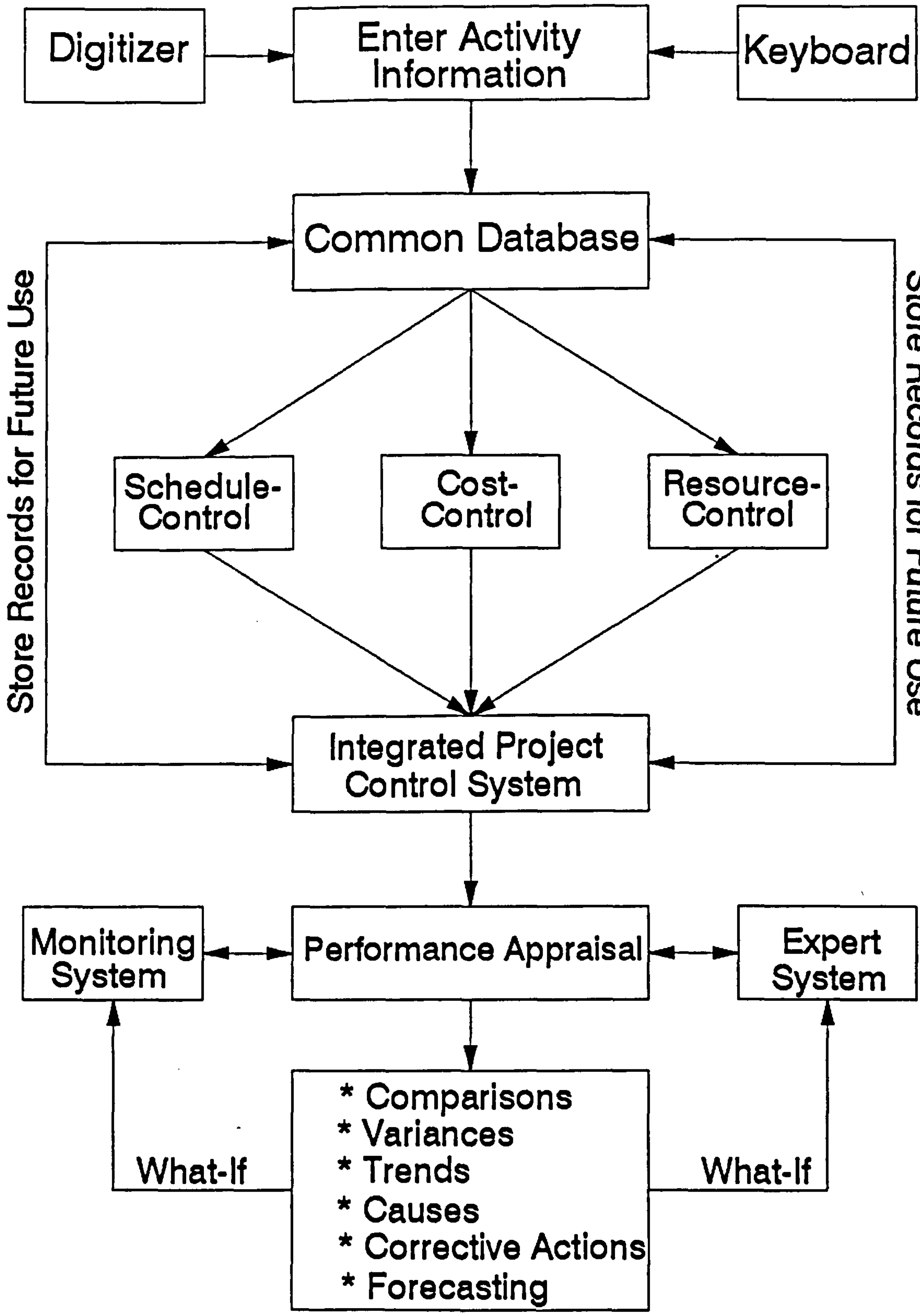
6.3 DIRECTIONS FOR FUTURE RESEARCH

The development of CM-DSS presents an essential first step toward developing more effective methods for planning and controlling practices. It addresses the basic needs of the

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

construction industry in the United States. To further advance these practices, it is recommended that future research should be geared toward incorporating CM-DSS into a broader project control system. This system should be capable of analyzing the performance of all activities, calculating the productivity rates and measuring trends to forecast possible deviations. It should also be capable of making comparisons, determining variances, and identifying the causes of these variances. When causes are determined, the system should provide the user with alternative solutions and corrective action(s). In addition, the system should be able to identify and analyze subjective data (strikes, weather, rework, etc.), and to consider their impact on performance. Forecasting of future performance is another feature that should be incorporated in such a system. To achieve this, an expert system that has a broad knowledge base can be interfaced with it. Such a system should also integrate major control functions, such as schedule control, cost control, and resource control. This integration requires sharing of information, and building a broad industry database. In addition, full integration requires the determination and compatibility of the objectives of the control functions. Figure (6.1) shows a schematic design for such a system.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.



Figure(6.1).-- Flow Chart of Future Project Control System

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

REFERENCES

(1) Adrian, James, J., Quantitative Methods in Construction Management, American Elsevier Publishing Company, Inc., 1973.

(2) Albert Battersby, Network Analysis for Planning and Scheduling, ST MARTIN'S PRESS, New York, N.Y., 1967.

(3) Alkayyali, O., Minkarah, I., and Mansour, W., "How Effective Is CPM In Planning and Controlling Construction Projects?," accepted for publication in *the 7TH CIB W-65 International Symposium on Organization and Management of Construction -the Way Forward-*, September, Trinidad, 1993.

(4) Alkayyali, O., and Minkarah, I.,"An Automated Cost Planning System (ACPS) for Construction Projects," proceeding of the Fifth International Conference on Computing in Civil and Building Engineering, V-ICCCBE, Anahiem, CA, June, 1993.

(5) Alkayyali, O., "Resource Allocation: A Comparative Analysis Toward A Practical Solution," Master Thesis, University of Cincinnati, Cincinnati, 1990.

(6) Arditi, D., "Diffusion of Network Planning in Construction," *Journal of Construction Engineering and Management*, ASCE, Vol. 109, No. CO1, Mar., 1983.

(7) Birrel, G. S., "Construction Planning-Beyond the Critical Path," *Journal of the Construction Division*, ASCE, Vol. 106, No. CO3, Sept., 1980.

(8) Branes, N.M.L., "Cost Modelling - An integrated Approach to Planning and Cost Control," *Engineering and Process Economics 2*, Elsvier Scientific Application Company, New York, N.Y., 1977.

(9) Carr,Robert, and Maloney, William, "Basic Research Needs in Construction Engineering," *Journal of Construction Engineering and Management*, ASCE, Vol. 109, No. 2, June, 1983.

(10) Chris Hendrickson, and Tung Au, Project Management for Construction, Prentice-Hall, Inc., New Jersey, 1989.

(11) David K. Smith, Network Optimization Practice a Computational Guide, ELLIS HORWOOD LIMITED Publishers, Chichester, U.K., 1982.

(12) Davis, E.W., "CPM Use in Top 400 Construction Firms," *Journal of the Construction Division*, ASCE, Vol. 100, No. CO1, March, 1974.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

(13) D. S. Kezsbom, D. L. Schilling, and K. A. Edward, Dynamic Project Management, John Wiley & Sons, Inc., New York, 1989.

(14) Edward Minieka, Optimization Algorithms for Networks and Graphs, Marcel Dekker, Inc., New York, 1978.

(15) Edwards W. Davis, "Project Network Summary Measures: Constrained-Resource Scheduling," *AIIE Transactions*, Volume 7, No. 2, June, 1975.

(16) Elliot B. Koffman, TURBO PASCAL, 3rd edition, Assison Wesley, Mew York, 1991.

(17) Elvers D.A., "Planning Monitoring Frequencies For CPM Projects," *Journal of Construction Division*, Proceedings of the American Society of Civil Engineers, Vol. 97, No. CO2, November, 1971.

(18) Farid, F., and Karshenas S., "Cost/Schedule Control System Criteria Under Inflation," *Project Management Journal*, Vol. XIX, No. 5, November, 1988.

(19) Gail Thornley, Critical Path Analysis collected papers on project control, Tavistock Publications, London, U.K., 1968.

(20) Galloway, P., and Nielsen, K., "Schedule Control For PCM Projects," *Journal of Construction Division*, Proceedings of the American Society of Civil Engineers, Vol. 107, No. CO2, June, 1981.

(21) Gough-Palmer, D. H., "Integration of Time nd Cost in the Planning and Monitoring of the Hong Kong Mass Transit Railway Project," *International Journal of Project Management*, Vol. 1, No. 2, May, 1983, pp. 95-100.

(22) Hamburger, D.H., " "On Time" Project Completion--Managing the Critical Path," *Project Management Journal*, Vol. XVIII, No. 4, September, 1987.

(23) Handa, V.K., and Barcia, R.M., "Construction Production Planning," *Journal of Construction Engineering and Management*, ASCE, Vol. 112, No. 2, June, 1986.

(24) Hinkley, P., "Planning and Control of Repetitive Construction," *Project Management Journal*, Vol. XX, No. 1, March, 1989.

(25) H.S. Woodgate, Planning by Network, Business Publications Limited, London, U.K., 1967.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

(26) Jaafari, A., "Criticism of CPM for Project Planning Analysis," *Journal of Construction Engineering and Management*, ASCE, Vol. 110, No. 2, June, 1984.

(27) Jack R. Meredith, and Samuel J. Mantel, Jr., Project Management a Managerial Approach, Second Edition, john Wiley & Sons, Inc., New York, 1989.

(28) James M. Antill, and Ronald W. Woodhead, Critical Path Methods in Construction Practice, Second Edition, John Wiley & Sons, Inc., New York, 1970.

(29) James D. Stevenes, Techniques for Construction Network Scheduling, McGraw-Hill, Inc., New York, 1990.

(30) James J. O'Brien, CPM in Construction Management, Third Edition, McGraw-Hill, Inc., New York, 1984.

(31) John W. Hackney, Control & Management of Capital Projects, Second Edition, McGraw-Hill, Inc., New York, 1992.

(32) Joseph J. Moder, Cecil R. Phillips, and Edward W. Davis, Project Management with CPM, PERT and Precedence Diagramming, Third Edition, Van Nostrand Reinhold Company, New York, 1983.

(33) Kawal, D.E., "Information Utilization in Project Planning," *Journal of Construction Division*, Proceedings of the American Society of Civil Engineers, Vol. 97, No. CO2, November, 1971.

(34) LeVitt, R.R., "New Methods in Construction Systems," *Journal of Construction Engineering and Management*, ASCE, Vol.100, No. CO3, September, 1974.

(35) Milton D. Rosenau, Jr., Successful Project Management A Step-by-Step Approach with Practical Examples, Second Edition, Van Nostrand Reinhold, New York, 1992.

(36) M. T. Callahan, D. G. Quackenbush, and J. E. Rowings, Construction Project Scheduling, McGraw-Hill, Inc., New York, 1992.

(37) Naaman, A.E., "Networking Methods for Project Planning and Control," *Journal of Construction Division*, Proceedings of the American Society of Civil Engineers, Vol. 100, No. CO3, September, 1974.

(38) O'Brien, J.J., Kreitzberg, F.C., and Mikes W.F., "Network Scheduling Variations for Repetitive Work," *Journal of Construction Engineering and Management*, ASCE, Vol. 111, No. 2, June, 1985.

(39) Ostwald, Phillip, Cost Estimating For Engineering and Management, Prentice-Hall, Inc., Englewood, New Jersy, 1974.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

(40) Panagiotakopoulos, D., "Cost-Time Model for Large CPM Project Networks," *Journal of Construction Division*, Proceedings of the American Society of Civil Engineers, Vol. 103, No. CO2, June, 1977.

(41) Peer, S., "Network Analysis and Construction Planning," *Journal of the construction Division*, ASCE, Vol. 100, No. CO3, Sept., 1974.

(42) Popescu, C., and Borcherding, J.D., "Developments in CPM, PERT, and Network Analysis," *Journal of Construction Division*, Proceedings of the American Society of Civil Engineers, Vol. 101, No. CO4, December, 1975.

(43) Prentis, E.L., "Master Project Plasnning: Scope, Time and Cost," *Project Management Journal*, Vol. XX, No. 1, March, 1989.

(44) Perera, S., "Linear Programming Solution to Network Compression," *Journal of Construction Division*, Proceedings of the American Society of Civil Engineers, Vol. 106, No. CO3, September, 1980.

(45) Richard H. Clough, and Glenn A. Sears, Construction Project Management, Third Edition, John Wiley & Sons, Inc., New York, 1991.

(46) Robert B. Harris, Precedence and Arrow Networking Techniques for Construction, John Wiley & Sons, Inc., New York, 1978.

(47) Royer, King, "The Federal Government and the Critical Path," *Journal of Construction Engineering and Management*, ASCE, Vol. 112, No. 2, June, 1986.

(48) Schlick, H., "Project Integrated Management System (PRIM)," *Journal of Construction Division*, Proceedings of the American Society of Civil Engineers, Vol. 107, No. CO2, June, 1981.

(49) Sears, G.A., "CPM/COST: An Integrated Approach," *Journal of Construction Division*, Proceedings of the American Society of Civil Engineers, Vol. 107, No. CO2, June, 1981.

(50) Tammimi, S., and Diekmann, J., "Soft Logic in Network Analysis," *Journal of Construction Engineering and Management*, ASCE, Vol. 2, No. 3, July, 1988.

(51) Tavakoli, A., "Effective Progress Scheduling and Control for Construction Projects," *Journal of Construction Engineering and Management*, ASCE, Vol. 6, No. 1, January, 1990.

(52) Tiong, R.L.K., "Effective Controls for Large Scale Construction Projects," *Project Management Journal*, Vol. XXI, No. 1, March, 1990.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

(53) William Press, Brian Flannery, Saul Teukolsky, and William Vetterling, NUMERICAL RECIPES IN PASCAL, The Art of Scientific Computing, Cambridge University Press, New York, 1990.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

PLANNING QUESTIONNAIRE

All responses will be kept confidential. Please respond by putting a check next to the appropriate number.

Title or position of the Respondent:______________________________

(I) COMPANY INFORMATION

[A] TYPE OF CONTRACTOR

__(1) COMMERCIAL
__(2) INDUSTRIAL
__(3) HIGHWAY AND HEAVY
__(4) RESIDENTIAL

[B] ANNUAL BILLINGS (CONSTRUCTION IN PLACE) (MILLIONS OF DOLLARS)

__(1) UNDER 5
__(2) 5 - 10
__(3) 10 - 50
__(4) 50 - 100
__(5) OVER 500

[C] NUMBER OF PERMANENT ENGINEERS

__(1) UNDER 10
__(2) 10 - 20
__(3) 20 - 50
__(4) 50 - 100
__(5) OVER 100

[D] NUMBER OF PROJECT MANAGERS

__(1) UNDER 5
__(2) 5 - 10
__(3) 10 - 20
__(4) 20 - 50
__(5) OVER 50

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

[E] AVERAGE JOB SIZE (MILLIONS OF DOLLARS)

__(1) UNDER 1
__(2) 1 - 5
__(3) 5 - 10
__(4) 10 - 25
__(5) 25 - 100
__(6) OVER 100

[F] AVERAGE PROJECT DURATION (YEARS)

__(1) UNDER 1/2
__(2) 1/2 - 1
__(3) 1 - 2
__(4) 2 - 3
__(5) OVER 3

[G] REGION OF OPERATION (CHECK ALL THAT APPLY)

__(1) EASTERN U.S.
__(2) MIDWEST U.S.
__(3) SOUTHERN U.S.
__(4) WESTERN U.S.
__(5) INTERNATIONAL
__(6) OTHER________________________________(PLEASE SPECIFY)

(II)PERSONAL INFORMATION

[A] EDUCATIONAL BACKGROUND

__(1) HIGH SCHOOL
__(2) BACHELOR
__(3) MASTER
__(4) OTHER________________________________(PLEASE SPECIFY)

[B] NUMBER OF YEARS YOU HAVE BEEN INVOLVED IN CONSTRUCTION

__(1) UNDER 5
__(2) 5 - 10
__(3) 10 - 15
__(4) 15 - 25
__(5) OVER 25

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

[C] NUMBER OF YEARS YOU HAVE BEEN WORKING AS A PROJECT MANAGER

__(1) UNDER 5
__(2) 5 - 10
__(3) 10 - 15
__(4) 15 - 25
__(5) OVER 25

[D] HOW DID YOU GAIN YOUR MANAGERIAL SKILLS?

__(1) EXPERIENCE
__(2) TECHNICAL LITERATURE
__(3) ATTENDING CONFERENCES AND SEMINARS
__(4) OTHER________________________________(PLEASE SPECIFY)

[E] MAXIMUM NUMBER OF PROJECTS YOU HAVE BEEN IN CHARGE OF AT A GIVEN TIME

__(1) 1
__(2) 1 - 5
__(3) 5 - 10
__(4) OVER 10

[F] AVERAGE PROJECT SIZE YOU HAVE BEEN IN CHARGE OF AT A GIVEN TIME (MILLIONS OF DOLLARS)

__(1) UNDER 1
__(2) 1 - 5
__(3) 5 - 10
__(4) 10 - 25
__(5) 25 - 100
__(6) OVER 100

(III) PLANNING A PROJECT

[A] HOW FREQUENTLY ARE YOU INVOLVED IN A PROJECT PLANNING PROCESS?

__(1) NEVER __(2) SOMETIMES __(3) ALWAYS

[B] IF (2) OR (3), WHEN? (PLEASE CHECK ALL THAT APPLY)

__(1) PREBIDDING PHASE
__(2) AFTER BIDDING AND BEFORE PROJECT EXECUTION PHASE
__(3) DURING PROJECT EXECUTION PHASE
__(4) OTHER________________________________(PLEASE SPECIFY)

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

[C] WHAT IS YOUR ROLE IN THE ABOVE STAGE(S)?

__(1) BREAKDOWN THE PROJECT INTO VARIOUS ACTIVITIES
__(2) PREPARE THE TECHNOLOGICAL LOGIC FOR THE PROJECT
__(3) PREPARE THE LIST OF RESOURCES
__(4) PREPARE A COMPREHENSIVE PLAN FOR THE PROJECT
__(5) OTHER________________________________(PLEASE SPECIFY)

[D] WHAT METHOD(S) DO YOU USUALLY USE IN PLANNING A PROJECT? (PLEASE CHECK ALL THAT APPLY)

__(1) BAR CHART (GANTT)
__(2) CRITICAL PATH METHOD (CPM)
__(3) PROGRAM EVALUATION AND REVIEW TECHNIQUE (PERT)
__(4) COMPUTER SCHEDULING SOFTWARE
__(5) OTHER______________________________(PLEASE SPECIFY)

[E] IF YOU ARE NOT COMFORTABLE WITH CPM, WOULD YOU PLEASE INDICATE THE REASON(S)?

__(1) TOO SOPHISTICATED TO BE USED
__(2) IMPRACTICAL FOR FIELD APPLICATION
__(3) RESULTS ARE DIFFICULT TO ANALYZE
__(4) UNFAMILIAR WITH FORMAT
__(5) LABORIOUS AND TIME-CONSUMING
__(6) EXPENSIVE TO UPDATE
__(7) UNREALISTIC
__(8) NOT FLEXIBLE
__(9) DOES NOT GUARANTEE AN OPTIMAL PLAN
__(10) OTHER_____________________________(PLEASE SPECIFY)

[F] IF YOU ARE NOT COMFORTABLE WITH PERT,WOULD YOU PLEASE INDICATE THE REASON(S)?

__(1) TOO SOPHISTICATED TO BE USED
__(2) IMPRACTICAL FOR FIELD APPLICATION
__(3) RESULTS ARE DIFFICULT TO ANALYZE
__(4) UNFAMILIAR WITH FORMAT
__(5) LABORIOUS AND TIME-CONSUMING
__(6) EXPENSIVE TO UPDATE
__(7) UNREALISTIC
__(8) NOT FLEXIBLE
__(9) DOES NOT GUARANTEE AN OPTIMAL PLAN
__(10) OTHER_____________________________(PLEASE SPECIFY)

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

[G] PLEASE INDICATE THE IMPACT OF EACH OF THE FOLLOWING FACTORS ON THE PLANNING PROCESS
[HIGHLY CRITICAL = 5; NOT CRITICAL = 1]

1. SIZE OF PROJECT	5	4	3	2	1
2. TYPE OF PROJECT	5	4	3	2	1
3. PROJECT DURATION	5	4	3	2	1
4. PROJECT LOCATION	5	4	3	2	1
5. PERSONAL EXPERIENCE OF PLANNER	5	4	3	2	1
6. AVAILABILITY OF:					
a. MANPOWER	5	4	3	2	1
b. MATERIAL	5	4	3	2	1
c. EQUIPMENT	5	4	3	2	1
d. FINANCE	5	4	3	2	1
7. COORDINATION WITH OTHER DEPARTMENTS	5	4	3	2	1
8. PLANNING METHODS AND TECHNIQUES	5	4	3	2	1

(IV) PROJECT CONTROL

[A] WHAT IS THE IMPORTANCE OF EACH OF THE FOLLOWING OBJECTIVES?
[HIGHLY IMPORTANT = 5; NOT IMPORTANT = 1]

1. SECURE PROFIT	5	4	3	2	1
2. MAINTAIN HIGH QUALITY	5	4	3	2	1
3. FINISH WITHIN THE PROJECT DURATION	5	4	3	2	1
4. IMPROVE PRODUCTIVITY	5	4	3	2	1
5. OWNER'S SATISFACTION	5	4	3	2	1
6. SAFETY	5	4	3	2	1
7. AVOID CLAIMS, DISPUTES, LITIGATION	5	4	3	2	1

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

[B] HOW FREQUENTLY DO YOU VISIT YOUR PROJECT(S)?

__(1) EVERY DAY
__(2) EVERY WEEK
__(3) BI-WEEKLY
__(4) EVERY MONTH
__(5) OTHER________________________________(PLEASE SPECIFY)

[C] WHAT METHOD(S) DO YOU ACTUALLY USE TO CONTROL YOUR PROJECT(S)? (PLEASE CHECK ALL THAT APPLY)

__(1) BAR CHART (GANTT)
__(2) CPM
__(3) PERT
__(4) COMPUTER SCHEDULING SOFTWARE
__(5) OTHER________________________________(PLEASE SPECIFY)

[D] DO YOU UTILIZE A "LEARNING CURVE" TO CONTROL YOUR PROJECT?

__(1) YES __(2) SOMETIMES __(3) NO

[E] DO YOU USE "BENEFIT TO COST RATIO" TO CONTROL YOUR PROJECT?

__(1) YES __(2) SOMETIMES __(3) NO

[F] DO YOU USUALLY USE PROJECT "COST CURVE" TO CONTROL YOUR PROJECT?

__(1) YES __(2) SOMETIMES __(3) NO

[G] DO YOU USE STATISTICAL OR MATHEMATICAL METHODS TO CONTROL YOUR PROJECT?

__(1) YES __(2) SOMETIMES __(3) NO

IF (1) OR (2), (PLEASE SPECIFY)

____

____

____(OPTIONAL)

[H] OTHER METHODS________________________________(PLEASE LIST)

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

(V) HUMAN RESOURCE PLANNING AND CONTROL

[A] AT WHAT STAGE SHOULD RESOURCES BE PLANNED?

__(1) PREBIDDING
__(2) AFTER BIDDING AND BEFORE PROJECT EXECUTION
__(3) DURING PROJECT EXECUTION
__(4) OTHER________________________________(PLEASE SPECIFY)

[B] HOW OFTEN DO YOU ENCOUNTER SITUATIONS WHERE THE FOLLOWING CATEGORIES OF HUMAN RESOURCES ARE LIMITED?

RESOURCE	RARELY	SOMETIMES	ALWAYS
1. UNSKILLED LABOR	______	______	______
2. SKILLED LABOR	______	______	______
3. FOREMEN	______	______	______
4. ENGINEERS	______	______	______

[C] HOW LONG DOES THIS SITUATION USUALLY LAST?

__(1) LESS THAN 1 WEEK
__(2) 1 - 2 WEEKS
__(3) 2 WEEKS - 1 MONTH
__(4) MORE THAN 1 MONTH

[D] HOW DO YOU USUALLY SOLVE THIS PROBLEM?

__(1) SPECIAL REALLOCATION TECHNIQUE(S)________(PLEASE SPECIFY)

__(2) REALLOCATION BASED ON EXPERIENCE
__(3) WORK OVERTIME
__(4) MORE SHIFTS
__(5) OTHER________________________________(PLEASE SPECIFY)

[E] IF YOU RELY ON YOUR EXPERIENCE ONLY, DO YOU CONSIDER IT ENOUGH TO HANDLE THE LIMITED-RESOURCE SITUATION?

__(1) YES __(2) SOMETIMES __(3) NO

[F] WOULD YOU PREFER TO USE "COMPUTER SCHEDULING SOFTWARE" IN PLANNING AND CONTROLLING YOUR HUMAN RESOURCES?

__(1) YES __(2) SOMETIMES __(3) NO

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

[G] DO YOU CONSIDER CHANGING THE ORIGINAL TECHNOLOGICAL LOGIC OF THE PROJECT, WHEN SOLVING THE LIMITED-RESOURCE PROBLEM?

__(1) YES __(2) NO

[H] IF YES, WHAT IS THE RESULT?

__(1) EXCELLENT
__(2) GOOD
__(3) NORMAL
__(4) POOR

(VI) COMPUTER FACILITIES

[A] DO YOU FEEL THAT THE USE OF COMPUTER FACILITIES IS ESSENTIAL IN PLANNING A PROJECT?

__(1) YES __(2) NO

[B] DO YOU FEEL THAT THE USE OF COMPUTER FACILITIES IS ESSENTIAL FOR SITE CONTROL OF A PROJECT?

__(1) YES __(2) NO

[C] DO YOU HAVE COMPUTER FACILITIES ON SITE?

__(1) YES __(2) NO

[D] DO YOU FEEL THAT THE "WINDOWING" FEATURE, WHICH IS OFFERED BY SOME COMPUTER SCHEDULING SOFTWARE, IS HELPFUL IN PLANNING AND CONTROLLING YOUR PROJECT?

__(1) YES __(2) NO

[E] DO YOU THINK THAT A 3-D PRESENTATION OF YOUR PROJECT NETWORK WOULD BE HELPFUL IN PLANNING AND CONTROLLING YOUR PROJECT?

__(1) YES __(2) NO

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

COMMENTS

[F] IF YOU HAVE ANY ADDITIONAL COMMENTS THAT WOULD HELP US UNDERSTAND YOUR APPROACH TO HANDLING PROJECT PLANNING AND CONTROL, PLEASE LIST BELOW.

____

____

____

____

____

THANK YOU VERY MUCH FOR YOUR COOPERATION

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

APPENDIX (II) CM-DSS SOURCE CODE

```
program CM_DSS (input,output);

          {************************************}
          {*                                  *}
          {*  PART # 1 : DECLARATION PART     *}
          {*                                  *}
          {************************************}

uses  crt;

const maxnodes    = 50;
      maxarcs     = 50;
      lomgnumber = 10000;

type  ActivityArr   = array[1..maxarcs] of integer;
      ActivityArr1  = array[1..maxarcs] of real;
      EventArr    = array[1..maxnodes] of integer;
      ActivityIndex = 1..MaxAct;
      ProductivityRec = record
      MH: array [1..4] of real;       {MH = Man-Hours           }
      PR: array [1..4] of real;       {PR = Productivity Rate }
      CS: array [1..4] of integer;    {CS = Crew Size          }
      UC: array [1..4] of real;       {UC = Unit Cost ($)      }
      AH: array [1..4] of real;       {AH = Adjusted Hour      }
      TC: array [1..4] of real;       {TC = Total Cost         }
      end; {ProductivityRec}
      ActivityArray = array[ActivityIndex] of ProductivityRec;
      numarraytype = array[1..maxsize, 1..maxisize] of integer;

var
      numarray: numarraytype;
      alternatives : 1..maxsize;
      scale, alternative, RI                    : real;
      sub1, sub2, sub3, sub4                    : real;
      worth1, worth2, worth3, worth4 : real;
      RV1, RV2, RV3, RV4 : real;
      answer1, answer2, answer3, YES, NO : string;
      number,ProjectDur1,ProjectDur2,ProjectDur3,ProjectDur4: integer;
      RES1,RES2,RES3,RES4 : real;
      CD,MaxResource : real;
      Cost1,Cost2,Cost3,Cost4 : real;
      factor : real;
      BUDGET : real;
      TC1,TC2,TC3,TC4 : integer;
      events,activities: integer;
      Act: ActivityArray;
      M, N, K, size: integer;
      i              {i    = Node i              } ,
      j              {j    = Node j              }    :Activityarr;
      Dur            {Dur  = Duration            }    :Activityarr;
      ES             {ES   = Early Start         } ,
      EF             {EF   = Early Finish        } ,
      LS             {LS   = Late Start          } ,
      LF             {LF   = Late Finish         } ,
      FF             {FF   = Free Float          } ,
      TF             {TF   = Total Float         } ,
      INTF           {INTF = Interfering Float   } ,
      INDF           {INDF = Independent Float   }    :Activityarr;
      EST            {EST  = Early Start Time    } ,
      LST            {LST  = Late  Start Time    }    : Eventarr;

      fileA   { Contains the originl Database prepared by the user              },
      fileA1  { Contains the modified Database after considering the Learning Effect },
      fileA3  { Contains the database in an ascending order of Crew Size         },
         {*** Files from B1 to E1 belong to the Cost-Based Schedule (CBS).      ***}
      fileB1  { Contains the MH, PR, CS, UC, AH, and TC values                   },
      fileC1  { Contains the CS, AH, and TC                                      },
      fileD1  { Contains the durations used to produce CBS                       },
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
fileE1  { Contains the final CPM analyses for CBS.                              },
      {*** Files from B2 to E2 belong to the Least Time Schedule (TBS).      ***}
fileB2  { Contains the MH, PR, CS, UC, AH, and TC values                        },
fileC2  { Contains the CS, AH, and TC                                           },
fileD2  { Contains the durations used to produce TBS                            },
fileE2  { Contains the final CPM analyses for TBS                               },
      {*** Files from B3 to E3 belong to the Limited Resource Schedule (RBS).***}
fileB3  { Contains the MH, PR, CS, UC, AH, and TC values                        },
fileC3  { Contains the CS, AH, and TC                                           },
fileD3  { Contains the durations used to produce RBS                            },
fileE3  { Contains the final CPM analyses for RBS                               },
      {*** Files from B4 to E4 belong to the Feasible Schedule (FS).         ***}
fileB4  { Contains the MH, PR, CS, UC, AH, and TC values                        },
fileC4  { Contains the CS, AH, and TC                                           },
fileD4  { Contains the durations used to produce FS                             },
fileE4  { Contains the final CPM analyses for FS                                },
fileB5  { This file is used as a temporary file                                 } : text;

Pro: ProductivityRec;
Least_Time1,Least_Time2,Least_Time3,Least_Time4          :real;
Least_Cost1,Least_Cost2,Least_Cost3,Least_Cost4          :real;
duration1, duration2, duration3, duration4               :real;
resource1, resource2, resource3, resource4               :integer;
LR, LimitedR, LResource                                  :real;
Total_Cost1,Total_Cost2,Total_Cost3,Total_Cost4          :real;
TOTAL1,TOTAL2,TOTAL3,TOTAL4                              :real;
NLI1  {NLI1  = Network Logic Index  for               CBS} ,
NLI2  {NLI2  = Network Logic Index  for               TBS} ,
NLI3  {NLI3  = Network Logic Index  for               RBS} ,
NLI4  {NLI4  = Network Logic Index  for               FS } ,
SCI1  {SCI1  = Schedule Criticality Index for         CBS} ,
SCI2  {SCI2  = Schedule Criticality Index for         TBS} ,
SCI3  {SCI3  = Schedule Criticality Index for         RBS} ,
SCI4  {SCI4  = Schedule Criticality Index for         FS } ,
SCOI1 {SCOI1 = Schedule Cost Index for                CBS} ,
SCOI2 {SCOI2 = Schedule Cost Index for                TBS} ,
SCOI3 {SCOI3 = Schedule Cost Index for                RBS} ,
SCOI4 {SCOI4 = Schedule Cost Index for                FS } ,
SRI1  {SRI11 = Schedule Resource Index for            CBS} ,
SRI2  {SRI12 = Schedule Resource Index for            TBS} ,
SRI3  {SRI13 = Schedule Resource Index for            RBS} ,
SRI4  {SRI14 = Schedule Resource Index for            FS } ,
TACI1 {TACI1 = Total Activity Criticality Index for   CBS} ,
TACI2 {TACI2 = Total Activity Criticality Index for   TBS} ,
TACI3 {TACI3 = Total Activity Criticality Index for   RBS} ,
TACI4 {TACI4 = Total Activity Criticality Index for   FS } ,
TACOI1{TACOI1= Total Activity Cost Index        for   CBS} ,
TACOI2{TACOI2= Total Activity Cost Index        for   TBS} ,
TACOI3{TACOI3= Total Activity Cost Index        for   RBS} ,
TACOI4{TACOI4= Total Activity Cost Index        for   FS } ,
TAD1  {TAD1  = Total Activity Durations for           CBS} ,
TAD2  {TAD2  = Total Activity Durations for           TBS} ,
TAD3  {TAD3  = Total Activity Durations for           RBS} ,
TAD4  {TAD4  = Total Activity Durations for           FS } ,
TARI1 {TARI1 = Total Activity Resource Index for      CBS} ,
TARI2 {TARI2 = Total Activity Resource Index for      TBS} ,
TARI3 {TARI3 = Total Activity Resource Index for      RBS} ,
TARI4 {TARI4 = Total Activity Resource Index for      FS } : real;
TRU1  {TRU1  = Total Resource Utilization for         CBS} ,
TRU2  {TRU2  = Total Resource Utilization for         TBS} ,
TRU3  {TRU3  = Total Resource Utilization for         RBS} ,
TRU4  {TRU4  = Total Resource Utilization for         FS } : real;
ACI1          : array [1..50] of real;
ACI2          : array [1..50] of real;
ACI3          : array [1..50] of real;
ACI4          : array [1..50] of real;
ACOI1         : array [1..50] of real;
ACOI2         : array [1..50] of real;
ACOI3         : array [1..50] of real;
ACOI4         : array [1..50] of real;
AGI1          : array [1..50] of real;
AGI2          : array [1..50] of real;
AGI3          : array [1..50] of real;
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
AGI4          : array [1..50] of real;
ARI1          : array [1..50] of real;
ARI2          : array [1..50] of real;
ARI3          : array [1..50] of real;
ARI4          : array [1..50] of real;
AINTI1        : array [1..50] of real;
AINTI2        : array [1..50] of real;
AINTI3        : array [1..50] of real;
AINTI4        : array [1..50] of real;
AINDI1        : array [1..50] of real;
AINDI2        : array [1..50] of real;
AINDI3        : array [1..50] of real;
AINDI4        : array [1..50] of real;
Optimal_Cost1: array [1..4]  of real;
Optimal_Cost2: array [1..4]  of real;
Optimal_Cost3: array [1..4]  of real;
Optimal_Cost4: array [1..4]  of real;
COS           : array [1..50] of real;
RES           : array [1..50] of real;
duration      : array [1..50] of real;
resource      : array [1..50] of real;

        {***********************************}
        {*                                 *}
        {*   PART # 2 : MODULARITY         *}
        {*                                 *}
        {***********************************}

{1} { ___________________________________
     |                                   |
     | This function determines the maximum value |
     | among several values.             |
     |                                   |
     |___________________________________|}

function max(x,y : integer) : integer;
  begin
    if (x < y) then
       max := y
    else max := x
  end {max};

{2} { ___________________________________
     |                                   |
     | This function determines the minimum value |
     | among several values.             |
     |                                   |
     |___________________________________|}

function min(x,y : integer) : integer;
  begin
    if (x < y) then
       min := x
    else min := y
  end {min};

{3} { ___________________________________
     |                                   |
     | This procedure is used to read the adjusted |
     | duration value for each activity and to |
     | produce the Cost_Based schedule.  |
     |___________________________________|}

procedure readinf1;
  var K: integer;
  begin
      repeat
      write('      How many events do you have ? > ');
      readln(events);
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
 until ((events in [2..maxnodes]) and (activities in [1..maxarcs]) and (activities >= events -1));

   writeln;
   write('      Please enter activities with their start event');
   writeln(' and finish event');
   writeln;
   writeln(' ******************************** ');
   writeln(' CM-DSS IS READING THE DATABASE ');
   writeln(' ******************************** ');
   writeln;
   writeln (' PLEASE WAIT......            ');
   writeln;
   Delay (15000);
   assign (fileD1, 'c:prodD1.dat');
   reset (fileD1);
   while not eof (fileD1) do
    begin
      for K := 1 to activities do
      begin
       readln (fileD1,Dur[K],COS[K],RES[K]);
      end;
    end;

   for K := 1 to activities do
     repeat
       write('      Activity (  ', K, '  )  ','     ');
       read(i[K], j[K]);
         if (i[K] >= j[K]) then
            writeln('      Start must be less than finish');
         if (Dur[K] < 0) then
            writeln('      Duration must be positive or zero');
     until (Dur[K] >= 0) and (i[K] < j[K]);
   close (fileD1);
   end {readinf1};

{4} { ________________________________________
     |                                        |
     | This procedure is used to read the adjusted |
     | duration value for each activity and to  |
     | produce the Time_Based schedule.         |
     |________________________________________|}

procedure readinf2;
  var K: integer;
  begin
   assign (fileD2, 'c:prodD2.dat');
   reset (fileD2);
   while not eof (fileD2) do
    begin
      for K := 1 to activities do
      begin
       readln (fileD2, Dur[K],COS[K],RES[K]);
      end;
    end;

   close (fileD2);
   end {readinf2};

{5} { ________________________________________
     |                                        |
     | This procedure is used to read the adjusted |
     | duration value for each activity and to  |
     | produce the Resource-Based schedule.     |
     |________________________________________|}

procedure readinf3;
  var k: integer;
  begin
   assign (fileD3, 'c:prodD3.dat');
   reset (fileD3);
   while not eof (fileD3) do
    begin
      for k := 1 to activities do
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
      begin
       readln (fileD3, Dur[k],COS[k],RES[k]);
      end;
    end;

   close (fileD3);
   end (readinf3);

(6) { ________________________________________
     |                                         |
     | This procedure is used to read the adjusted |
     | duration value for each activity and to  |
     | produce the Feasible schedule.           |
     |_________________________________________|}

procedure readinf4;
  var K: integer;
  begin
   assign (fileD4, 'c:prodD4.dat');
   reset (fileD4);
   while not eof (fileD4) do
    begin
      for K := 1 to activities do
      begin
       readln (fileD4,Dur[K],COS[K],RES[K]);
      end;
    end;

   close (fileD4);
   end (readinf4);


(7) { ________________________________________
     |                                         |
     | This procedure is used to perform the CPM |
     | computations for CBS.                    |
     |_________________________________________|}


procedure CPM1;
 var K,P : integer;
 begin
    writeln('          *********************************************');
    writeln('          CM-DSS IS ANALYZING THE COST BASED SCHEDULE');
    writeln('          *********************************************');
    writeln;
    writeln('          PLEASE WAIT......                ');
    writeln;
    Delay (15000);
    for P := 1 to events do
     begin
       EST[P] := 0;
       LST[P] := INF
     end;
     K := 1;

    repeat
       EST[j[K]] := max(EST[j[K]], EST[i[K]] + Dur[K]);
       K:= K + 1
    until (K > activities);
    K := activities;
    LST[events] := EST[events];

    repeat
       LST[i[K]] := min(LST[i[K]], LST[j[K]] - Dur[K]);
       K:= K - 1
    until (K < 1);

   TARI1    := 0;
   TACOI1   := 0;
   TOTAL1   := 0;
   TACI1    := 0;
   ACI1[1]  := 0;
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
AGI1[1]  := 0;
ACOI1[1] := 0;
ARI1 [1] := 0;
for K := 1 to activities do
  begin
     ES[K]   := EST[i[K]];
     EF[K]   := ES[K] + Dur[K];
     LF[K]   := LST[j[K]];
     LS[K]   := LF[K] - Dur[K];
     TF[K]   := LF[K] - ES[K] - Dur[K];
     INTF[K] := max(0, EST[j[K]] - LST[i[K]] - Dur[K]);
     INDF[K] := LST[j[K]] - LST[i[K]] - Dur[K];
     FF[K]   := EST[j[K]] - EST[i[K]] - Dur[K];
     if TF[K] = 0 then ACI1[K] := 100
     else ACI1[K]   := (1 - (TF[K])/(Dur[K] + TF[K])) * 100;
     if INTF[K] = 0 then AINTI1[K] := 100
     else AINTI1[K] := ( 1 - (INTF[K])/(Dur[K] + INTF[K])) * 100;
     if INDF[K] = 0 then AINDI1[K] := 100
     else AINDI1[K] := ( 1 - (INDF[K])/(Dur[K] + INDF[K])) * 100;
     TACI1    := TACI1 + ACI1[K];
     ACOI1[K] := ((COS[K])/(Total_Cost1)) * 100;
     TACOI1   := TACOI1 + COS[K];
     ARI1[K]  := (((RES[K])*(Dur[K])/(TRU1))) * 100;
     TARI1    := TARI1 + ((RES[K])*(Dur[K]));
     TOTAL1   := (ACI1[K] + ACOI1[K] + ARI1[K] + AINTI1[K] + AINDI1[K]);
     AGI1[K]  := ((TOTAL1)/(5));
  end;
     ProjectDur1 := EF[activities];
end {CPM1};

{8}{ _________________________________________
   |                                         |
   | This procedure is used to perform the CPM |
   | computations for TBS.                   |
   |_________________________________________|}

procedure CPM2;
 var K,P : integer;
 begin

   writeln('        *********************************************');
   writeln('        CM-DSS IS ANALYZING THE TIME BASED SCHEDULE');
   writeln('        *********************************************');
   writeln;
   writeln('        PLEASE WAIT......              ');
   writeln;
   Delay (15000);
   for P := 1 to events do
    begin
      EST[P] := 0;
      LST[P] := INF
    end;
    K := 1;

   repeat
      EST[j[K]] := max(EST[j[K]], EST[i[K]] + Dur[K]);
      K:= K + 1
   until (K > activities);
   K := activities;
   LST[events] := EST[events];

   repeat
      LST[i[K]] := min(LST[i[K]], LST[j[K]] - Dur[K]);
      K:= K - 1
   until (K < 1);

   TARI2    := 0;
   TACOI2   := 0;
   TOTAL2   := 0;
   TACI2    := 0;
   ACI2[1]  := 0;
   AGI2[1]  := 0;
   for K := 1 to activities do
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
     begin
        ES[K]    := EST[i[K]];
        EF[K]    := ES[K] + Dur[K];
        LF[K]    := LST[j[K]];
        LS[K]    := LF[K] - Dur[K];
        TF[K]    := LF[K] - ES[K] - Dur[K];
        INTF[K] := max(0, EST[j[K]] - LST[i[K]] - Dur[K]);
        INDF[K] := LST[j[K]] - LST[i[K]] - Dur[K];
        FF[K]    := EST[j[K]] - EST[i[K]] - Dur[K];
        if TF[K] = 0 then ACI2[K] := 100
        else ACI2[K]   := (1 - (TF[K])/(Dur[K] + TF[K])) * 100;
        if INTF[K] = 0 then AINTI2[K] := 100
        else AINTI2[K] := ( 1 - (INTF[K])/(Dur[K] + INTF[K])) * 100;
        if INDF[K] = 0 then AINDI2[K] := 100
        else AINDI2[K] := ( 1 - (INDF[K])/(Dur[K] + INDF[K])) * 100;
        TACI2    := TACI2 + ACI2[K];
        ACOI2[K] := ((COS[K])/(Total_Cost2)) * 100;
        ARI2[K]  := ((RES[K])*(Dur[K])/(TRU2)) * 100;
        TOTAL2   := (ACI2[K] + ACOI2[K] + ARI2[K] + AINTI2[K] + AINDI2[K]);
        AGI2[K]  := (TOTAL2)/(5);
        TACOI2   := TACOI2 + COS[K];
        TARI2    := TARI2 + ((RES[K])*(Dur[K]));
     end;
        ProjectDur2 := EF[activities];
end (CPM2);

(9)( ________________________________________
     |                                        |
     | This procedure is used to perform the CPM |
     | computations for RBS.                  |
     |________________________________________|)


procedure CPM3;
 var K,P : integer;
 begin

    writeln('         **************************************************');
    writeln('         CM-DSS IS ANALYZING THE RESOURCE BASED SCHEDULE');
    writeln('         **************************************************');
    writeln;
    writeln('         PLEASE WAIT......               ');
    writeln;
    Delay (15000);

    for P := 1 to events do
     begin
       EST[P] := 0;
       LST[P] := INF
     end;
     K := 1;

    repeat
       EST[j[K]] := max(EST[j[K]], EST[i[K]] + Dur[K]);
       K:= K + 1
    until (K > activities);
    K := activities;
    LST[events] := EST[events];

    repeat
       LST[i[K]] := min(LST[i[K]], LST[j[K]] - Dur[K]);
       K:= K - 1
    until (K < 1);

    TARI3    := 0;
    TACOI3   := 0;
    TOTAL3   := 0;
    TACI3    := 0;
    ACI3[1]  := 0;
    AGI3[1]  := 0;
    for K := 1 to activities do
     begin
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
        ES[K]    := EST[i[K]];
        EF[K]    := ES[K] + Dur[K];
        LF[K]    := LST[j[K]];
        LS[K]    := LF[K] - Dur[K];
        TF[K]    := LF[K] - ES[K] - Dur[K];
        INTF[K] := max(0, EST[j[K]] - LST[i[K]] - Dur[K]);
        INDF[K] := LST[j[K]] - LST[i[K]] - Dur[K];
        FF[K]    := EST[j[K]] - EST[i[K]] - Dur[K];
        if TF[K] = 0 then ACI3[K] := 100
        else ACI3[K]    := (1 - (TF[K])/(Dur[K] + TF[K])) * 100;
        if INTF[K] = 0 then AINTI3[K] := 100
        else AINTI3[K] := ( 1 - (INTF[K])/(Dur[K] + INTF[K])) * 100;
        if INDF[K] = 0 then AINDI3[K] := 100
        else AINDI3[K] := ( 1 - (INDF[K])/(Dur[K] + INDF[K])) * 100;
        TACI3    := TACI3 + ACI3[K];
        ACOI3[K] := ((COS[K])/(Total_Cost3)) * 100;
        ARI3[K]  := ((RES[K])*(Dur[K])/(TRU3)) * 100;
        TOTAL3   := (ACI3[K] + ACOI3[K] + ARI3[K] + AINTI3[K] + AINDI3[K]);
        AGI3[K]  := (TOTAL3)/(5);
        TACOI3   := TACOI3 + COS[K];
        TARI3    := TARI3 + ((RES[K])*(Dur[K]));
     end;
        ProjectDur3 := EF[activities];
end {CPM3};


{10}{ ____________________________________
     |                                    |
     | This procedure is used to perform the CPM |
     | computations for FS.               |
     |____________________________________|}

procedure CPM4;
 var K,P : integer;
 begin

    writeln('        ******************************************');
    writeln('        CM-DSS IS ANALYZING THE FEASIBLE SCHEDULE');
    writeln('        ******************************************');
    writeln;
    writeln('        PLEASE WAIT......              ');
    writeln;
    Delay (15000);

    for P := 1 to events do
     begin
       EST[P] := 0;
       LST[P] := INF
     end;
     K := 1;

    repeat
       EST[j[K]] := max(EST[j[K]], EST[i[K]] + Dur[K]);
       K:= K + 1
    until (K > activities);
    K := activities;
    LST[events] := EST[events];

    repeat
       LST[i[K]] := min(LST[i[K]], LST[j[K]] - Dur[K]);
       K:= K - 1
    until (K < 1);

    TARI4    := 0;
    TACOI4   := 0;
    TOTAL4   := 0;
    TACI4    := 0;
    ACI4[1]  := 0;
    AGI4[1]  := 0;
    for K := 1 to activities do
     begin
       ES[K]   := EST[i[K]];
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
        EF[K]    := ES[K] + Dur[K];
        LF[K]    := LST[j[K]];
        LS[K]    := LF[K] - Dur[K];
        TF[K]    := LF[K] - ES[K] - Dur[K];
        INTF[K] := max(0, EST[j[K]] - LST[i[K]] - Dur[K]);
        INDF[K] := LST[j[K]] - LST[i[K]] - Dur[K];
        FF[K]    := EST[j[K]] - EST[i[K]] - Dur[K];
        if TF[K] = 0 then ACI4[K] := 100
        else ACI4[K]   := (1 - (TF[K])/(Dur[K] + TF[K])) * 100;
        if INTF[K] = 0 then AINTI4[K] := 100
        else AINTI4[K] := ( 1 - (INTF[K])/(Dur[K] + INTF[K])) * 100;
        if INDF[K] = 0 then AINDI4[K] := 100
        else AINDI4[K] := ( 1 - (INDF[K])/(Dur[K] + INDF[K])) * 100;
        TACI4    := TACI4 + ACI4[K];
        ACOI4[K] := ((COS[K])/(Total_Cost4)) * 100;
        ARI4[K]  := ((RES[K])*(Dur[K])/(TRU4)) * 100;
        TOTAL4   := (ACI4[K] + ACOI4[K] + ARI4[K] + AINTI4[K] + AINDI4[K]);
        AGI4[K]  := (TOTAL4)/(5);
        TACOI4   := TACOI4 + COS[K];
        TARI4    := TARI4 + ((RES[K]))*((Dur[K]));
      end;
        ProjectDur4 := EF[activities];

end {CPM4};


{11} { ____________________________________________
      |                                            |
      | This procedure is used to write the results |
      | of the CPM analysis for the Cost_Based      |
      | schedule.                                   |
      |____________________________________________|}


procedure path1;  {Calculate CPM for the Cost_Based Schedule}
  var K : integer;
  begin
    assign (fileE1,'c:prodE1.dat');
    rewrite(fileE1);
    writeln(fileE1,' ');
    writeln(fileE1,'***** CASE # 1 : THE COST_BASED SCHEDULE *******');
    writeln(fileE1,' ');
    writeln(fileE1,'==========================================================================');
    write(fileE1,'Act.No',' i ', ' j ', ' Dur ', ' ES ', ' EF ',' LS ',' LF ',' TF ');
    writeln(fileE1,' FF ', ' INTF ', 'INDF ','ACI ',' AINTI ',' AINDI ','  ACOI',' ARI', ' AGI');
    writeln(fileE1,'==========================================================================');
    writeln;
    for K := 1 to activities do
      begin
        write(fileE1,K:3,i[K]:4,j[K]:4,Dur[K]:4,ES[K]:4,EF[K]:4,LS[K]:4,LF[K]:4,TF[K]:4,FF[K]:4,INTF[K]:4);
       writeln(fileE1,INDF[K]:4,ACI1[K]:8:2,AINTI1[K]:8:2,AINDI1[K]:8:2,ACOI1[K]:6:2,ARI1[K]:6:2,AGI1[K]:6:2);
      end;
    writeln;
    writeln (fileE1,'==========================================================================');
    writeln (fileE1);
    writeln (fileE1,'1. The critical activities have Criticality Index of 100%');
    writeln (fileE1,'2. Total Cost                                         = $',Total_Cost1 :8:3);
    writeln (fileE1,'3. Schedule Duration "activities in parallel"         =  ',EF[activities],' hours');
    writeln (fileE1,'4. Schedule Duration "activities in series"           =  ',TAD1:5:0,     ' hours');
                    RES1 := TRU1;
    writeln (fileE1,'5. Total Resource Utilization                         = ', RES1 :6:0,' man-hours');
                    NLI1 := ( 1 - (EF[activities])/(TAD1)) * 100;
    writeln (fileE1,'6. Network Logic Index                                = ', NLI1:4:2, ' %');
                    SCI1 := ((TACI1)/(activities));
    writeln (fileE1,'7. Schedule Criticality Index                         = ', SCI1:4:2, ' %');
                    SCOI1:= ((TACOI1)/(activities));
    writeln (fileE1,'8. Schedule Cost Index                                =  $', SCOI1:6:2,' /activity ');
                    SRI1  := ((TARI1)/(activities));
    writeln (fileE1,'9. Schedule Resource Index                                    = ', SRI1 :6:2,'units of
resource/activity');
                    Cost1 := Total_Cost1;

   close (fileE1);
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
end (path1);

{12} {
      This procedure is used to write the results
      of the CPM analysis for the Time_Based
      schedule.
                                                    }

procedure path2;  {Calculate CPM for the Time_Based Schedule}
  var K : integer;
  begin
    assign (fileE2,'c:prodE2.dat');
    rewrite(fileE2);
    writeln(fileE2,' ');
    writeln(fileE2,'***** CASE # 2 : THE TIME_BASED SCHEDULE *******');
    writeln(fileE2,' ');
    writeln(fileE2,'==========================================================================');
    write(fileE2,'Act.No',' i ', ' j ', ' Dur ', ' ES ', ' EF ',' LS ',' LF ',' TF ');
    writeln(fileE2,' FF ', ' INTF ', 'INDF ','ACI ',' AINTI ',' AINDI ',' ACOI',' ARI ', ' AGI');
    writeln(fileE2,'==========================================================================');
    writeln;
    for K := 1 to activities do
      begin
        write(fileE2,K:3,i[K]:4,j[K]:4,Dur[K]:4,ES[K]:4,EF[K]:4,LS[K]:4,LF[K]:4,TF[K]:4,FF[K]:4,INTF[K]:4);
        writeln(fileE2,INDF[K]:4,ACI2[K]:8:2,AINTI2[K]:8:2,AINDI2[K]:8:2,ACOI2[K]:6:2,ARI2[K]:6:2,AGI2[K]:8:2);
      end;
    writeln;
    writeln (fileE2,'==========================================================================');
    writeln (fileE2);
    writeln (fileE2,'1. The critical activities have Criticality Index of 100%');
    writeln (fileE2,'2. Total Cost                                        = $', Total_Cost2 :8:3);
    writeln (fileE2,'3. Schedule Duration "activities in parallel"        = ', EF[activities],' hours');
    writeln (fileE2,'4. Schedule Duration "activities in series"          = ', TAD2:5:0,     ' hours');
                  RES2 := TRU2;
    writeln (fileE2,'5. Total Resource Utilization                        = ', RES2 :6:0,' man-hours');
                  NLI2 := ( 1 - (EF[activities])/(TAD2)) * 100;
    writeln (fileE2,'6. Network Logic Index                               = ', NLI2 :4:2,' %');
                  SCI2 := ((TACI2)/(activities));
    writeln (fileE2,'7. Schedule Criticality Index                        = ', SCI2 :4:2,' %');
                  SCOI2:= ((TACOI2)/(activities));
    writeln (fileE2,'8. Schedule Cost Index                               = $', SCOI2:6:2, '/activity');
                  SRI2  := ((TARI2)/(activities));
    writeln (fileE2,'9. Schedule Resource Index                                = ', SRI2 :6:2,'units of
resource/activity');
                  Cost2 := Total_Cost2;

    close (fileE2);

end (path2);

{13}{
      This procedure is used to write the results
      of the CPM analysis for the Resource_Based
      schedule.
                                                    }

procedure path3;  {Calculate CPM for the Resource_Based Schedule}
  var K : integer;
  begin
    assign (fileE3,'c:prodE3.dat');
    rewrite(fileE3);
    writeln(fileE3,' ');
    writeln(fileE3,'***** CASE # 3 : THE RESOURCE_BASED SCHEDULE *******');
    writeln(fileE3,' ');
    writeln(fileE3,'==========================================================================');
    write(fileE3,'Act.No',' i ', ' j ', ' Dur ', ' ES ', ' EF ',' LS ',' LF ',' TF ');
    writeln(fileE3,' FF ', ' INTF ', 'INDF ','ACI ',' AINTI ',' AINDI ',' ACOI',' ARI', 'AGI');
    writeln(fileE3,'==========================================================================');
    writeln;
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
    for K := 1 to activities do
      begin
        write(fileE3,K:3,i[K]:4,j[K]:4,Dur[K]:4,ES[K]:4,EF[K]:4,LS[K]:4,LF[K]:4,TF[K]:4,FF[K]:4,INTF[K]:4);
        writeln(fileE3,INDF[K]:4,ACI3[K]:8:2,AINTI3[K]:8:2,AINDI3[K]:8:2,ACOI3[K]:6:2,ARI3[K]:6:2,AGI3[K]:8:2);
      end;
    writeln;
    writeln (fileE3,'==============================================================================');
    writeln (fileE3);
    writeln (fileE3,'1. The critical activities have Criticality Index of 100%');
    writeln (fileE3,'2. Total Cost                                       = $',Total_Cost3 :8:3);
    writeln (fileE3,'3. Schedule Duration "activities in parallel"      = ', EF[activities],' hours');
    writeln (fileE3,'4. Schedule Duration "activities in series"        = ', TAD3:5:0,      ' hours');
                    RES3 := TRU3;
    writeln (fileE3,'5. Total Resource Utilization                       = ', RES3:6:0,' man-hours');

                    NLI3 := ( 1 - (EF[activities])/(TAD3)) * 100;
    writeln (fileE3,'6. Network Logic Index                              =  ', NLI3  :4:2,' %');
                    SCI3 := ((TACI3)/(activities));
    writeln (fileE3,'7. Schedule Criticality Index                       =  ', SCI3  :4:2,' %');
                    SCOI3:= ((TACOI3)/(activities));
    writeln (fileE3,'8. Schedule Cost Index                              = $', SCOI3:6:2,' /activity');
                    SRI3  := ((TARI3)/(activities));
    writeln (fileE3,'9. Schedule Resource Index                              =   ', SRI3 :6:2,' units of
resource/activity ');
                    Cost3:= Total_Cost3;

    close (fileE3);

end {path3};


{14}{ ____________________________________________
     |                                            |
     | This procedure is used to write the results|
     | of the CPM analysis for the Feasible schedule.|
     |____________________________________________|}

Procedure path4;  {Calculate CPM for the Feasible Schedule}
  var K : integer;
  begin
    assign (fileE4,'c:prodE4.dat');
    rewrite(fileE4);
    writeln(fileE4,' ');
    writeln(fileE4,'***** CASE # 4 : THE FEASIBLE SCHEDULE *******');
    writeln(fileE4,' ');
    writeln(fileE4,'==============================================================================');
    write(fileE4,'Act.No',' i ', ' j ', ' Dur ', ' ES ', ' EF ',' LS ',' LF ',' TF ');
    writeln(fileE4,' FF ', ' INTF ', 'INDF ','ACI ',' AINTI ',' AINDI ',' ACOI',' ARI', ' AGI');
    writeln(fileE4,'==============================================================================');
    writeln;
    for K := 1 to activities do
      begin
        write(fileE4,K:3,i[K]:4,j[K]:4,Dur[K]:4,ES[K]:4,EF[K]:4,LS[K]:4,LF[K]:4,TF[K]:4,FF[K]:4,INTF[K]:4);
        writeln(fileE4,INDF[K]:4,ACI4[K]:8:2,AINTI4[K]:8:2,AINDI4[K]:8:2,ACOI4[K]:6:2,ARI4[K]:6:2,AGI4[K]:8:2);
      end;
    writeln;
    writeln (fileE4,'==============================================================================');
    writeln (fileE4);
    writeln (fileE4,'1. The critical activities have Criticality Index of 100%');
    writeln (fileE4,'2. Total Cost                                       =  $',Total_Cost4 :8:3);
    writeln (fileE4,'3. Schedule Duration "activities in parallel"      =   ',EF[activities],' hours');
    writeln (fileE4,'4. Schedule Duration "activities in series"        =   ', TAD4:5:0,'     hours');
                    RES4 := TRU4;
    writeln (fileE4,'5. Total Resource Utilization                       =   ', RES4 :6:0,' man-hours');
                    NLI4 := ( 1 - (EF[activities])/(TAD4)) * 100;
    writeln (fileE4,'6. Network Logic Index                              =   ', NLI4 :4:2,' %');
                    SCI4 := ((TACI4)/(activities));
    writeln (fileE4,'7. Schedule Criticality Index                       =   ', SCI4 :4:2,' %');
                    SCOI4:= ((TACOI4)/(activities));
    writeln (fileE4,'8. Schedule Cost Index                              = $', SCOI4:6:2,' /activity');
                    SRI4  := ((TARI4)/(activities));
    writeln (fileE4,'9. Schedule Resource Index                              =    ', SRI4 :6:2,' units of
resource/activity');
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
                    Cost4 := Total_Cost4;
     close (fileE4);
  end (path4);

(15){ ______________________________________________
     |                                              |
     | This procedure is used to determine the least|
     | cost value for each activity and the         |
     | corresponding duration and crew size values. |
     |______________________________________________|}

     {**************************************************}
     {.........THE COST-BASED SCHEDULE................}
     {**************************************************}

procedure L_cost;

begin  {Read the Database and Develop the Cost_Based Schedule}
   writeln;
   writeln ('      **********************************');
   writeln ('      PLEASE RESPOND TO EACH QUESTION');
   writeln ('      **********************************');
   writeln;
   write ('      How many activities do you have ? >  ' );
   readln (activities);
   assign (fileA, 'c:prodA.dat');
   reset (fileA);
   assign (fileB1, 'c:prodB1.dat');
   rewrite(fileB1);
   assign (fileC1,'c:prodC1.dat');
   rewrite(fileC1);
   while not eof (fileA) do
     begin
       for M:= 1 to 4 do
         begin
           readln (fileA,Pro.MH[M],Pro.PR[M],Pro.CS[M],Pro.UC[M]);
           Pro.AH[M]:= Pro.MH[M]/(Pro.PR[M] * Pro.CS[M]);
           Pro.TC[M]:= Pro.AH[M] * Pro.UC[M] * Pro.CS[M];
          writeln(fileB1,Pro.MH[M]:3:0,Pro.PR[M]:5:2,Pro.CS[M]:4,Pro.UC[M]:5:0,Pro.AH[M]:6:2,Pro.TC[M]:8:2);
         end;  {end for loop}
     end;     {end while loop}

   close (fileA);
   reset (fileB1);

   while not eof (fileB1) do
     begin
       writeln('IF YOU HAVE SOME CONSTRAINED ACTIVITIES');
       writeln;
       writeln('PLEASE READ THE FOUR MODULARS FOR EACH ACTIVITY');
       writeln;
       writeln('AND SELECT ONE MODULAR ONLY.');
       writeln;
       writeln('IF YOU DO NOT HAVE CONSTRAINED ACTIVITIES, PLEASE ENTER ( 0, 0, 0)');
       writeln;
       writeln('PSCSYS WILL GIVE YOU SOME TIME TO READ THIS INFORMATION.');

       writeln('          ************************************');
       writeln('      Act.','      ','MH','       ','PR','      ','CS','     ','UC','     ',' AH','    ',' TC');
       writeln('          ************************************');
       writeln(fileC1,'**********************************************************');
       writeln(fileC1,'Act.','      ','MH','       ','PR','      ','CS','     ','UC','     ',' AH','    ',' TC');
       writeln(fileC1,'**********************************************************');
       for N:= 1 to activities do
         begin
          writeln(fileC1,N);
          writeln(N);
          for M := 1 to 4 do
             begin
               readln(fileB1,Act[N].MH[M],Act[N].PR[M],Act[N].CS[M],Act[N].UC[M],Act[N].AH[M],Act[N].TC[M]);
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
            write(fileC1,'        ',Act[N].MH[M]:5,'  ',Act[N].PR[M]:5:3,'  ',Act[N].CS[M]:5,'  ');
            writeln(fileC1,Act[N].UC[M]:5:0,'  ',Act[N].AH[M]:5:3,'  ',Act[N].TC[M]:5:3);
            writeln(Act[N].UC[M]:5:0,'  ',Act[N].AH[M]:5:3,'  ',Act[N].TC[M]:5:3);
          end;
        writeln(fileC1,'====================================================');
        writeln('====================================================');
        Delay(3000);
       end;
    end;
 Close (fileB1);
 Close (fileC1);

 assign (fileD1, 'c:prodD1.dat');
 rewrite(fileD1);

 resource1    := 0;
 duration1    := 0;
 Least_Cost1  := 0;
 Total_Cost1  := 0;
 TAD1         := 0;
 TRU1         := 0;
 for N := 1 to activities do
   begin
     M:= 1;
     resource1    := Act[N].CS[M];
     duration1    := Act[N].AH[M];
     Least_Cost1  := Act[N].TC[M];

     for M := 2 to 4 do
        if Act[N].TC[M] < Least_Cost1
           then
              begin
                resource1    := Act[N].CS[M];
                duration1    := Act[N].AH[M];
                Least_Cost1  := Act[N].TC[M];
                {duration[K]:= duration[N];}
              end
            else
              begin
                resource1    := resource1;
                duration1    := duration1;
                Least_Cost1  := Least_Cost1;
               { duration[K]:= duration[N]; }
              end;
     if resource1 = 1 then
        RES[N] := 0
     else
        RES[N] := resource1;
     duration1 := duration1;
     Optimal_Cost1 [N]:= Least_Cost1;
     TRU1 := TRU1 + (resource1 * duration1);
     TAD1 := TAD1 + duration1;
     Total_Cost1  :=  Total_Cost1 + Optimal_Cost1 [N];
     writeln(fileD1,duration1:5:0,Optimal_Cost1[N]:8:2,RES[N]:4:0);
   end;
   close (fileD1);
   Cost1 := Total_Cost1;
   TAD1  := TAD1;
   TRU1  := TRU1;

  writeln ('          ***********************************************');
  writeln ('          CM-DSS IS DEVELOPING THE COST BASED SCHEDULE ');
  writeln ('          ***********************************************');
  writeln;
  writeln ('          PLEASE WAIT.........');
  writeln;
  Delay (15000);
end;
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
{16}{ ______________________________________________
     |                                              |
     | This procedure is used to determine the least|
     | time value for each activity and the         |
     | corresponding cost and crew size values.     |
     |______________________________________________|}

     {***************************************************}
     {.........THE TIME_BASED SCHEDULE................}
     {***************************************************}

procedure L_Time;

begin  {Read the Database and Develop the Time_Based Schedule}
   assign (fileA, 'c:prodA.dat');
   reset (fileA);
   assign (fileB2, 'c:prodB2.dat');
   rewrite(fileB2);
   assign (fileC2,'c:prodC2.dat');
   rewrite(fileC2);
   while not eof (fileA) do
     begin
       for M:= 1 to 4 do
          begin
            readln (fileA,Pro.MH[M],Pro.PR[M],Pro.CS[M],Pro.UC[M]);
            Pro.AH[M]:= Pro.MH[M]/(Pro.PR[M] * Pro.CS[M]);
            Pro.TC[M]:= Pro.AH[M] * Pro.UC[M] * Pro.CS[M];
            writeln(fileB2,Pro.CS[M]:4,Pro.AH[M]:6:2,Pro.TC[M]:8:2);
          end;  {end for loop}
     end;     {end while loop}

   close (fileA);
   reset (fileB2);

   while not eof (fileB2) do
     begin
       writeln(fileC2,'*********************************************************');
       writeln(fileC2,'Act.',' ' ,'CS',' ',' AH',' ',' TC');
       writeln(fileC2,'*********************************************************');
       for N:= 1 to activities do
         begin
          writeln(fileC2,N);
          for M := 1 to 4 do
             begin
               readln (fileB2,Act[N].CS[M],Act[N].AH[M],Act[N].TC[M]);
               writeln(fileC2,'        ',Act[N].CS[M]:5,' ',Act[N].AH[M]:5:3,' ',Act[N].TC[M]:5:3);
             end;
          writeln(fileC2,'==================================================');
         end;
     end;
  Close (fileB2);
  Close (fileC2);

  assign (fileD2, 'c:prodD2.dat');
  rewrite(fileD2);

  resource2   := 0;
  duration2   := 0;
  Least_Cost2 := 0;
  Total_Cost1 := 0;
  Total_Cost2 := 0;
  TAD2        := 0;
  TRU2        := 0;
  for N := 1 to activities do
    begin
      M:= 1;
      resource2   := Act[N].CS[M];
      duration2   := Act[N].AH[M];
      Least_Cost2 := Act[N].TC[M];

      for M := 2 to 4 do
         if Act[N].AH[M] < duration2
            then
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
            begin
              resource2   := Act[N].CS[M];
              duration2   := Act[N].AH[M];
              Least_Cost2 := Act[N].TC[M];
              {duration[K]:= duration[N];}
            end
          else
            begin
              resource2   := resource2;
              duration2   := duration2;
              Least_Cost2 := Least_Cost2;
             { duration[K]:= duration[N]; }
            end;
      if resource2 = 1 then
         RES[N] := 0
      else
         RES [N] := resource2;
      duration2 := duration2;
      Optimal_Cost2 [N]:= Least_Cost2;
      TRU2 := TRU2 + (resource2 * duration2);
      TAD2 := TAD2 + duration2;
      Total_Cost2  :=  Total_Cost2 + Optimal_Cost2 [N];
      writeln(fileD2,duration2:5:0,Optimal_Cost2[N]:8:2,RES[N]:4:0);
    end;
    close (fileD2);
    Cost2 := Total_Cost2;
    TAD2 := TAD2;
    TRU2 := TRU2;


   writeln ('          ***********************************************');
   writeln ('          CM_DSS IS DEVELOPING THE TIME BASED SCHEDULE ');
   writeln ('          ***********************************************');
   writeln;
   writeln ('          PLEASE WAIT.........');
   writeln;
   Delay (15000);


end; {end of L_Time}

{17}{ ______________________________________________
     |                                              |
     | This procedure is used to determine the least |
     | resource value for the project and to         |
     | determine the corresponding cost and time     |
     | values for each activity.                     |
     |______________________________________________|}

     {*************************************************}
     {...........THE RESOURCE BASED SCHEDULE..........}
     {*************************************************}

{Read Database and Develop the Resource_Based Schedule}

Procedure L_Resource;
 begin
   assign (fileA, 'c:prodA.dat');
   reset (fileA);
   assign (fileB3, 'c:prodB3.dat');
   rewrite(fileB3);
   assign (fileC3,'c:prodC3.dat');
   rewrite(fileC3);
   while not eof (fileA) do
     begin
       for M:= 1 to 4 do
          begin
            readln (fileA,Pro.MH[M],Pro.PR[M],Pro.CS[M],Pro.UC[M]);
            Pro.AH[M]:= Pro.MH[M]/(Pro.PR[M] * Pro.CS[M]);
            Pro.TC[M]:= Pro.AH[M] * Pro.UC[M] * Pro.CS[M];
            writeln(fileB3,Pro.CS[M]:4,Pro.AH[M]:6:2,Pro.TC[M]:8:2);
          end;  {end for loop}
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
  end;     {end while loop}

 close (fileA);
 reset (fileB3);

 while not eof (fileB3) do
   begin
    writeln(fileC3,'*********************************************************');
    writeln(fileC3,'Act.',' ','CS',' ',' AH',' ',' TC');
    writeln(fileC3,'*********************************************************');
    for N:= 1 to activities do
      begin
       writeln(fileC3,N);
       for M := 1 to 4 do
          begin
            readln (fileB3,Act[N].CS[M],Act[N].AH[M],Act[N].TC[M]);
            writeln(fileC3,'        ',Act[N].CS[M]:5,'  ',Act[N].AH[M]:5:3,'  ',Act[N].TC[M]:5:3);
          end;
       writeln(fileC3,'==================================================');
      end;
   end;
Close (fileB3);
Close (fileC3);

assign (fileD3, 'c:prodD3.dat');
rewrite(fileD3);

{Total_Cost2 := 0;}
resource3   := 0;
duration3   := 0;
Least_Cost3 := 0;
Total_Cost3 := 0;
TAD3        := 0;
TRU3        := 0;
for N := 1 to activities do
  begin
    M:= 1;
    resource3   := Act[N].CS[M];
    duration3   := Act[N].AH[M];
    Least_Cost3 := Act[N].TC[M];

    for M := 2 to 4 do
       if Act[N].CS[M] < resource3
          then
             begin
               resource3   := Act[N].CS[M];
               duration3   := Act[N].AH[M];
               Least_Cost3 := Act[N].TC[M];
               {duration[K]:= duration[N];}
             end
           else
             begin
               resource3   := resource3;
               duration3   := duration3;
               Least_Cost3 := Least_Cost3;
              { duration[K]:= duration[N]; }
             end;
    if resource3 = 1 then
       RES[N] := 0
    else
       RES [N]   := resource3;
    duration3    := duration3;
    Optimal_Cost3 [N] := Least_Cost3;
    TRU3          := TRU3 + (resource3 * duration3);
    TAD3          := TAD3 + duration3;
    Total_Cost3  := Total_Cost3 + Optimal_Cost3 [N];
   writeln(fileD3,duration3:5:0,Optimal_Cost3[N]:8:2,RES[N]:4:0);
  end;
  close (fileD3);
  Cost3 := Total_Cost3;
  TAD3 := TAD3;
  TRU3 := TRU3;
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
    writeln ('          ****************************************************');
    writeln ('          CM_DSS IS DEVELOPING THE RESOURCE BASED SCHEDULE ');
    writeln ('          ****************************************************');
    writeln;
    writeln ('          PLEASE WAIT.........');
    writeln;
    Delay (15000);

end; {end of L_Time}

{18}{ ______________________________________________
     |                                              |
     | This procedure is used to read the constraint|
     | value imposed on each activity by the planner|
     | (if any) and to determine the feasible schedule.|
     |______________________________________________|}

     {****************************************************}
     {...........THE FEASIBLE SCHEDULE..................}
     {****************************************************}

{Read the Constraints and the Database and Develop the Feasible Schedule}

procedure Feasible;

begin
   assign (fileB4, 'c:prodB4.dat');
   rewrite (fileB4);
   assign (fileB2, 'c:prodB2.dat');
   reset (fileB2);
   assign (fileB5, 'c:prodB5.dat');
   rewrite (fileB5);
   writeln ('          ***********************************************');
   writeln ('          CM-DSS WANTS TO PRODUCE THE FEASIBLE SCHEDULE');
   writeln ('          ***********************************************');
   writeln;
   writeln;
   for N := 1 to activities do
    begin
      M := 1;
      writeln ('Please enter one modular of constraints for activity no.', N );
      write ('  Resource ', ' Time ', ' &  Budget >  ');
      readln (Act[N].CS[M],Act[N].AH[M],Act[N].TC[M]);
      writeln;
      if Act[N].CS[M] > 0 then
        begin
          writeln (fileB4,Act[N].CS[M],Act[N].AH[M],Act[N].TC[M]);
          writeln (fileB4,Act[N].CS[M],Act[N].AH[M],Act[N].TC[M]);
          writeln (fileB4,Act[N].CS[M],Act[N].AH[M],Act[N].TC[M]);
          writeln (fileB4,Act[N].CS[M],Act[N].AH[M],Act[N].TC[M]);
          for M := 1 to 4 do
            begin
              readln (fileB2,Act[N].CS[M],Act[N].AH[M],Act[N].TC[M]);
              writeln(fileB5,Act[N].CS[M],Act[N].AH[M],Act[N].TC[M]);
            end;
        end
      else if Act[N].CS[M] = 0 then
        begin
          for M := 1 to 4 do
            begin
              readln (fileB2,Act[N].CS[M],Act[N].AH[M],Act[N].TC[M]);
              writeln(fileB4,Act[N].CS[M],Act[N].AH[M],Act[N].TC[M]);
            end;
        end; { end if }
    end;  { end N for loop }

    assign (fileC4,'c:prodC4.dat');
    rewrite(fileC4);
    reset (fileB4);

    while not eof (fileB4) do
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
    begin
      writeln(fileC4,'**************************************************');
      writeln(fileC4,'Act.',' CS',' ',' AH',' ',' TC');
      writeln(fileC4,'**************************************************');
      for N:= 1 to activities do
        begin
         writeln(fileC4,N);
         for M := 1 to 4 do
            begin
              readln (fileB4,Act[N].CS[M],Act[N].AH[M],Act[N].TC[M]);
              write(fileC4,'        ',Act[N].CS[M]:5,'  ',Act[N].AH[M]:5:3,'  ',Act[N].TC[M]:5:3);
            end;
         writeln(fileC4,'==================================================');
        end;
     end;
  Close (fileB4);
  Close (fileC4);

  assign (fileD4, 'c:prodD4.dat');
  rewrite(fileD4);
  resource4   := 0;
  duration4   := 0;
  Least_Cost4 := 0;
  Total_Cost4 := 0;
  TAD4        := 0;
  TRU4        := 0;
  for N := 1 to activities do
    begin
      M:= 1;
      resource4   := Act[N].CS[M];
      duration4   := Act[N].AH[M];
      Least_Cost4 := Act[N].TC[M];

      for M := 2 to 4 do
         if Act[N].TC[M] < Least_Cost4
            then
              begin
                resource4   := Act[N].CS[M];
                duration4   := Act[N].AH[M];
                Least_Cost4 := Act[N].TC[M];
                end
             else
              begin
                resource4   := resource4  ;
                duration4   := duration4  ;
                Least_Cost4 := Least_Cost4;
               end;

       if resource4 = 1 then
         RES[N] := 0
      else
         RES[N] := resource4;
      Optimal_Cost4 [N]:= Least_Cost4;
      TRU4 := TRU4 + (resource4 * duration4);
      TAD4 := TAD4 + duration4;
      Total_Cost4  :=  Total_Cost4 + Optimal_Cost4 [N];
      writeln(fileD4,duration4:5:0,Optimal_Cost4[N]:8:2,RES[N]:4:0);
    end;
    close (fileD4);
    Cost4 := Total_Cost4;
    TAD4  := TAD4;
    TRU4  := TRU4;

    writeln ('         ********************************************');
    writeln ('         CM-DSSIS DEVELOPING THE FEASIBLE SCHEDULE ');
    writeln ('         ********************************************');
    writeln;
    writeln ('         PLEASE WAIT.........');
    writeln;
    Delay (15000);

end; {end of Feasible}
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
{***********************************}
{*                                 *}
{*    PART # 3 : THE MAIN BODY     *}
{*                                 *}
{***********************************}

begin {Program CM_DSS}

  textcolor (2);
  writeln('     ***************************************************** ');
  writeln('     *  *********************************************  * ');
  writeln('     *  *  ***************************************  *  * ');
  writeln('     *  *                                          *  * ');
  writeln('     *  *                WELCOME TO                *  * ');
  writeln('     *  *                                          *  * ');
  writeln('     *  *             CM - DSS PROGRAM             *  * ');
  writeln('     *  *                                          *  * ');
  writeln('     *  *                   for                    *  * ');
  writeln('     *  *                                          *  * ');
  writeln('     *  *  Planning, Scheduling, and Controlling   *  * ');
  writeln('     *  *                                          *  * ');
  writeln('     *  *          Construction Projects           *  * ');
  writeln('     *  *                                          *  * ');
  writeln('     *  *              (c) Copyright               *  * ');
  writeln('     *  *                    by                    *  * ');
  writeln('     *  *                                          *  * ');
  writeln('     *  *          Osama Jawad Alkayyali           *  * ');
  writeln('     *  *                                          *  * ');
  writeln('     *  *                  1993.                   *  * ');
  writeln('     *  *                                          *  * ');
  writeln('     *  *  ***************************************  *  * ');
  writeln('     *  *********************************************  * ');
  writeln('     ***************************************************** ');

  Delay (15000);
  clrscr;
  textcolor (3);
  writeln('     ************************************************************ ');
  writeln('     *                                                          * ');
  writeln('     *  CM_DSS is a CPM-based program. It is developed to       * ');
  writeln('     *  help the project manager and/or planner in planning     * ');
  writeln('     *  an activity under variable productivity, time,          * ');
  writeln('     *  resource, and budget constraints, and to generate       * ');
  writeln('     *  four CPM-based scheduling alternatives.                 * ');
  writeln('     *                                                          * ');
  writeln('     *  CM_DSS is capable of considering different types        * ');
  writeln('     *  of constraints such as time, resources, and budget      * ');
  writeln('     *  simultaneously. In addition, it provides the project    * ');
  writeln('     *  manager with the flexibility to select a target         * ');
  writeln('     *  schedule among these alternatives by using a priority * ');
  writeln('     *  matrix, and to examine different scheduling scenarios * ');
  writeln('     *  by conducting What-If analyses.                         * ');
  writeln('     *                                                          * ');
  writeln('     *  PLEASE load the Database file on the hard disk before * ');
  writeln('     *  continuing with CM-DSS.                                 * ');
  writeln('     *                                                          * ');
  writeln('     ************************************************************ ');
  Delay (15000);
  clrscr;
  textcolor (3);

            {******************************************}
            {        The Learning Curve Effect         }
            {******************************************}

  assign (fileA1, 'c:proda1.dat');
  reset  (fileA1);
  assign (fileA,'c:proda.dat');
  rewrite(fileA);
  writeln;
  writeln('Do You Want To Consider The Learning Effect? YES or NO');
  readln(answer1);
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
IF answer1 = 'YES' THEN   {...if answer1...}
  begin
    Delay(3000);
    writeln('          ************************************');
    writeln('          THIS IS THE LEARNING CURVE FEATURE');
    writeln('          ************************************');
    writeln;
    writeln('          *********************************');
    writeln('          PLEASE RESPOND TO EACH QUESTION');
    writeln('          *********************************');
    clrscr;
    writeln;
    write('How many activities do you have?  >  ');
    read(activities);
    writeln;
    for N := 1 to activities do
     begin
        write('What is the execution order of activity  (', N ,' ) ?  >  ');
        read(number);
         if number = 1 then
           begin
             Factor := 1.0;
             for M := 1  to 4 do
             begin
                 readln(fileA1, Act[N].MH[M], Act[N].PR[M], Act[N].CS[M], Act[N].UC[M]);
                 Act[N].MH[M] := Act[N].MH[M] * Factor;
                 writeln(fileA,Act[N].MH[M]:3:0, Act[N].PR[M]:5:2, Act[N].CS[M]:4, Act[N].UC[M]:4:0);
             end; {end of M loop}
           end
        else if number = 2 then
           begin
             Factor := 0.80;
             for M := 1  to 4 do
             begin
                 readln(fileA1, Act[N].MH[M], Act[N].PR[M], Act[N].CS[M], Act[N].UC[M]);
                 Act[N].MH[M] := Act[N].MH[M] * Factor;
                 writeln(fileA,Act[N].MH[M]:3:0, Act[N].PR[M]:5:2, Act[N].CS[M]:4, Act[N].UC[M]:4:0);
             end; {end of M loop}
           end
        else if number = 3 then
           begin
             Factor := 0.702;
             for M := 1  to 4 do
              begin
                 readln(fileA1, Act[N].MH[M], Act[N].PR[M], Act[N].CS[M], Act[N].UC[M]);
                 Act[N].MH[M] := Act[N].MH[M] * Factor;
                 writeln(fileA,Act[N].MH[M]:3:0, Act[N].PR[M]:5:2, Act[N].CS[M]:4, Act[N].UC[M]:4:0);
              end; {end of M loop}
           end
        else if number = 4 then
           begin
             Factor := 0.640;
             for M := 1  to 4 do
             begin
                 readln(fileA1, Act[N].MH[M], Act[N].PR[M], Act[N].CS[M], Act[N].UC[M]);
                 Act[N].MH[M] := Act[N].MH[M] * Factor;
                 writeln(fileA,Act[N].MH[M]:3:0, Act[N].PR[M]:5:2, Act[N].CS[M]:4, Act[N].UC[M]:4:0);
             end; {end of M loop}
        end;
    end; {end of N loop}

    close (fileA1);
    close (fileA);

        L_Cost;    {******** The First Schedule *******}
        readinf1;
        sortevent;
        complete(ok);
        if ok then
          begin
            CPM1;
            path1;
          end;
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
L_Time;    {******** The Second Schedule *******}
readinf2;
sortevent;
complete(ok);
if ok then
  begin
    CPM2;
    path2;
  end;
L_Resource; {******* The Third Schedule ********}
readinf3;
sortevent;
complete(ok);
if ok then
  begin
    CPM3;
    path3;
  end;
Feasible;  {******** The Fourth Schedule ********}
readinf4;
sortevent;
complete(ok);
if ok then
   begin
     CPM4;
     path4;
   end;
         {******* The Worth Matrix Part *******}
clrscr;
textcolor(3);
writeln ('          **********************************');
writeln ('          THIS IS THE WORTH MATRIX FEATURE');
writeln ('          **********************************');
writeln;
REPEAT
 writeln('DO YOU WANT TO RUN WHAT-IF ANALSIS?  YES or NO  >  ');
 readln(answer3);
 if answer3 = 'YES' then
  begin
   writeln;
   writeln('Do you accept the Feasible Schedule as a target schedule ?');
   write(' Please respond by YES or NO ');
   readln(answer2);
 if answer2 = 'NO' then
   begin
     worth1 := 0;
     worth2 := 0;
     worth3 := 0;
     worth4 := 0;
     writeln;
     write('      Enter relative importance RI for DURATION >  ');
     read (RI);
     writeln;
     write('      Enter the contractual duration >  ');
     read (CD);
     RV1 := (CD/ProjectDur1) * 10.00;
     RV2 := (CD/ProjectDur2) * 10.00;
     RV3 := (CD/ProjectDur3) * 10.00;
     RV4 := (CD/ProjectDur4) * 10.00;
     sub1 := RI * RV1;
     Worth1 := sub1 + Worth1;
     sub2 := RI * RV2;
     Worth2 := sub2 + Worth2;
     sub3 := RI * RV3;
     Worth3 := sub3 + Worth3;
     sub4 := RI * RV4;
     Worth4 := sub4 + Worth4;

     RV1 := 0; RV2 := 0; RV3 := 0; RV4 := 0;
     writeln;
     write('      Enter relative importance RI for TOTAL COST >  ');
     read (RI);
     writeln;
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
write('      Enter the project budget >  ');
read (BUDGET);
RV1 := ((BUDGET)/(Cost1)) * 10.00;
RV2 := ((BUDGET)/(Cost2)) * 10.00;
RV3 := ((BUDGET)/(Cost3)) * 10.00;
RV4 := ((BUDGET)/(Cost4)) * 10.00;
sub1 := RI * RV1;
Worth1 := sub1 + Worth1;
sub2 := RI * RV2;
Worth2 := sub2 + Worth2;
sub3 := RI * RV3;
Worth3 := sub3 + Worth3;
sub4 := RI * RV4;
Worth4 := sub4 + Worth4;

RV1 := 0; RV2 := 0; RV3 := 0; RV4 := 0;
writeln;
write('      Enter relative importance RI for RESOURCE UTILIZATION >  ');
read (RI);
writeln;
write('      Enter the project MaxResource >  ');
readln (MAXRESOURCE);
RV1 := (MAXRESOURCE/TRU1) * 10.00;
RV2 := (MAXRESOURCE/TRU2) * 10.00;
RV3 := (MAXRESOURCE/TRU3) * 10.00;
RV4 := (MAXRESOURCE/TRU4) * 10.00;
sub1 := RI * RV1;
Worth1 := sub1 + Worth1;
sub2 := RI * RV2;
Worth2 := sub2 + Worth2;
sub3 := RI * RV3;
Worth3 := sub3 + Worth3;
sub4 := RI * RV4;
Worth4 := sub4 + Worth4;

RV1 := 0; RV2 := 0; RV3 := 0; RV4 := 0;
writeln;
write('      Enter relative importance RI for NETWORK CRITICALITY INDEX >  ');
read (RI);
writeln;
RV1 := (1 - (NLI1)/(100)) * 10.00;
RV2 := (1 - (NLI2)/(100)) * 10.00;
RV3 := (1 - (NLI3)/(100)) * 10.00;
RV4 := (1 - (NLI4)/(100)) * 10.00;
sub1 := RI * RV1;
Worth1 := sub1 + Worth1;
sub2 := RI * RV2;
Worth2 := sub2 + Worth2;
sub3 := RI * RV3;
Worth3 := sub3 + Worth3;
sub4 := RI * RV4;
Worth4 := sub4 + Worth4;

RV1 := 0; RV2 := 0; RV3 := 0; RV4 := 0;
writeln;
write('      Enter relative importance RI for SCHEDULE CRITICALITY INDEX >  ');
read (RI);
writeln;
RV1 := (1 - (SCI1)/(100)) * 10.00;
RV2 := (1 - (SCI2)/(100)) * 10.00;
RV3 := (1 - (SCI3)/(100)) * 10.00;
RV4 := (1 - (SCI4)/(100)) * 10.00;
sub1 := RI * RV1;
Worth1 := sub1 + Worth1;
sub2 := RI * RV2;
Worth2 := sub2 + Worth2;
sub3 := RI * RV3;
Worth3 := sub3 + Worth3;
sub4 := RI * RV4;
Worth4 := sub4 + Worth4;

writeln;
    writeln ('      Worth Of The Feasible Schedule       = ', Worth1:7:3);
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
                writeln ('      Worth Of The Cost_Based Schedule     = ', Worth2:7:3);
                writeln ('      Worth Of The Time_Based Schedule     = ', Worth3:7:3);
                writeln ('      Worth Of The Resource_Based Schedule = ', Worth4:7:3);
                writeln;
                writeln ('      The schedule that has the highest worth value is the best');
                writeln;
                writeln ('      ************************************************');
                writeln ('                      THANK YOU FOR USING CM-DSS          ');
                writeln ('      ************************************************');
             writeln;
             end  {end of if answer2}
          else
             begin
                writeln  ('      ************************************************');
                writeln  ('                      THANK YOU FOR USING CM-DSS          ');
                writeln  ('      ************************************************');
             end;
          end; { end of if answer3}
        writeln('DO YOU WANT TO RUN WHAT-IF ANALSIS?  YES or NO  >  ');
        readln(answer3);
        UNTIL (answer3 = 'NO');
       end
   ELSE { start of else answer1 }
     begin
       L_Cost;    {******** The First Schedule *******}
       readinf1;
       sortevent;
       complete(ok);
       if ok then
         begin
          CPM1;
          path1;
         end;
       L_Time;    {******** The Second Schedule *******}
       readinf2;
       sortevent;
       complete(ok);
       if ok then
         begin
          CPM2;
          path2;
         end;
       L_Resource; {******* The Third Schedule ********}
       readinf3;
       sortevent;
       complete(ok);
       if ok then
        begin
          CPM3;
          path3;
        end;
       Feasible;  {******** The Fourth Schedule ********}
       readinf4;
       sortevent;
       complete(ok);
       if ok then
         begin
          CPM4;
          path4;
         end;
             {******* The Worth Matrix Part *******}
       clrscr;
       textcolor(3);

       writeln ('          ********************************');
       writeln ('          THIS IS THE WORTH MATRIX FEATURE');
       writeln ('          ********************************');

     REPEAT
       writeln;
       writeln('DO YOU WANT TO RUN WHAT-IF ANALSIS?  YES or NO  >  ');
       readln(answer3);
       if answer3 = 'YES' then
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
    begin
    writeln;
    writeln('Do you accept the Feasible Schedule as a target schedule ?');
    write(' Please respond by YES or NO ');
    readln(answer2);
 if answer2 = 'NO' then
     begin
         worth1 := 0;
         worth2 := 0;
         worth3 := 0;
         worth4 := 0;
         writeln;
         write('      Enter relative importance RI for DURATION >  ');
         readln (RI);
         writeln;
         write('      Enter the contractual duration >  ');
         readln (CD);
         RV1 := (CD/ProjectDur1) * 10;
         RV2 := (CD/ProjectDur2) * 10;
         RV3 := (CD/ProjectDur3) * 10;
         RV4 := (CD/ProjectDur4) * 10;
         sub1 := RI * RV1;
         Worth1 := sub1 + Worth1;
         sub2 := RI * RV2;
         Worth2 := sub2 + Worth2;
         sub3 := RI * RV3;
         Worth3 := sub3 + Worth3;
         sub4 := RI * RV4;
         Worth4 := sub4 + Worth4;

         writeln;
         write('      Enter relative importance RI for TOTAL COST >  ');
         readln (RI);
         writeln;
         write('      Enter the project budget >  ');
         readln (BUDGET);
         RV1 := (BUDGET/Cost1) * 10;
         RV2 := (BUDGET/Cost2) * 10;
         RV3 := (BUDGET/Cost3) * 10;
         RV4 := (BUDGET/Cost4) * 10;
         sub1 := RI * RV1;
         Worth1 := sub1 + Worth1;
         sub2 := RI * RV2;
         Worth2 := sub2 + Worth2;
         sub3 := RI * RV3;
         Worth3 := sub3 + Worth3;
         sub4 := RI * RV4;
         Worth4 := sub4 + Worth4;
         writeln;
         write('      Enter relative importance RI for RESOURCE UTILIZATION >  ');
         readln (RI);
         writeln;
         write('      Enter the project MaxResource >  ');
         readln (MAXRESOURCE);
         RV1 := (MAXRESOURCE/TRU1) * 10;
         RV2 := (MAXRESOURCE/TRU2) * 10;
         RV3 := (MAXRESOURCE/TRU3) * 10;
         RV4 := (MAXRESOURCE/TRU4) * 10;
         sub1 := RI * RV1;
         Worth1 := sub1 + Worth1;
         sub2 := RI * RV2;
         Worth2 := sub2 + Worth2;
         sub3 := RI * RV3;
         Worth3 := sub3 + Worth3;
         sub4 := RI * RV4;
         Worth4 := sub4 + Worth4;

         writeln;
         write('      Enter relative importance RI for NETWORK CRITICALITY INDEX >  ');
         readln (RI);
         writeln;
         RV1 := (1 - (NLI1)/(100)) * 10;
         RV2 := (1 - (NLI2)/(100)) * 10;
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
            RV3 := (1 - (NLI3)/(100)) * 10;
            RV4 := (1 - (NLI4)/(100)) * 10;
            sub1 := RI * RV1;
            Worth1 := sub1 + Worth1;
            sub2 := RI * RV2;
            Worth2 := sub2 + Worth2;
            sub3 := RI * RV3;
            Worth3 := sub3 + Worth3;
            sub4 := RI * RV4;
            Worth4 := sub4 + Worth4;

            writeln;
            write('      Enter relative importance RI for SCHEDULE CRITICALITY INDEX >  ');
            readln (RI);
            writeln;
            RV1 := (1 - (SCI1)/(100)) * 10;
            RV2 := (1 - (SCI2)/(100)) * 10;
            RV3 := (1 - (SCI3)/(100)) * 10;
            RV4 := (1 - (SCI4)/(100)) * 10;
            sub1 := RI * RV1;
            Worth1 := sub1 + Worth1;
            sub2 := RI * RV2;
            Worth2 := sub2 + Worth2;
            sub3 := RI * RV3;
            Worth3 := sub3 + Worth3;
            sub4 := RI * RV4;
            Worth4 := sub4 + Worth4;

            writeln;
                writeln ('      Worth Of The Feasible Schedule       = ', Worth1:7:3);
                writeln ('      Worth Of The Cost_Based Schedule     = ', Worth2:7:3);
                writeln ('      Worth Of The Time_Based Schedule     = ', Worth3:7:3);
                writeln ('      Worth Of The Resource_Based Schedule = ', Worth4:7:3);
                writeln;
                writeln ('      The schedule that has the highest worth value is the best');
                writeln;
       writeln ('      ***************************************************');
       writeln ('                     THANK YOU FOR USING CM-DSS          ');
       writeln ('      ***************************************************');
       writeln;
   end
  else
     begin
       writeln ('      ***************************************************');
       writeln ('                     THANK YOU FOR USING CM-DSS          ');
       writeln ('      ***************************************************');
     end
 end; { end of if answer2 }

writeln('DO YOU WANT TO RUN WHAT-IF ANALSIS?  YES or NO  >  ');
readln(answer3);
UNTIL (answer3 = 'NO');
end;
end.
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

APPENDIX (III) -- CM-DSS FILES

File (fileA1.dat)

40	0.79	3	30
40	0.88	4	30
40	1.00	5	30
40	0.92	6	30
90	0.86	2	28
90	1.00	3	28
90	0.84	4	28
90	0.80	6	28
70	0.79	3	30
70	0.87	4	30
70	1.00	5	30
70	0.85	7	30
80	0.60	4	30
80	0.89	6	30
80	1.00	8	30
80	0.58	9	30
95	0.82	2	30
95	0.90	3	30
95	1.00	6	30
95	0.88	4	30
72	0.78	2	35
72	0.88	4	35
72	1.00	6	35
72	0.82	8	35
0	1.00	1	0
0	1.00	1	0
0	1.00	1	0
0	1.00	1	0
56	0.85	4	45
56	1.00	7	45
56	0.90	6	45
56	0.75	3	45
96	0.80	4	40
96	1.00	8	40
96	0.92	6	40
96	0.72	2	40
84	0.79	2	40
84	0.84	4	40
84	1.00	6	40
84	0.82	3	40
80	0.85	2	40
80	1.00	5	40
80	0.90	4	40
80	0.75	10	40
64	0.89	3	35
64	0.92	4	35
64	1.00	8	35
64	0.75	6	35
56	0.77	2	40
56	1.00	4	40
56	0.82	7	40
56	0.79	6	40
72	0.62	4	40
72	0.74	5	40
72	1.00	6	40
72	0.69	3	40
75	0.83	6	45
75	1.00	9	45
75	0.88	7	45
75	0.79	4	45
72	0.79	5	45
72	0.84	4	45
72	1.00	6	45
72	0.72	3	45
64	0.90	4	35
64	1.00	8	35
64	0.88	5	35

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

64	0.75	3	35
80	0.76	6	35
80	0.90	7	35
80	1.00	5	35
80	0.70	3	35
72	0.80	3	30
72	0.84	4	30
72	1.00	6	30
72	0.82	5	30
56	0.76	3	25
56	1.00	7	25
56	0.86	4	25
56	0.82	5	25
90	0.88	2	35
90	0.92	3	35
90	1.00	5	35
90	0.90	4	35
96	0.74	3	25
96	1.00	8	25
96	0.88	4	25
96	0.79	2	25

File (fileA.dat)

40	0.79	3	30
40	0.88	4	30
40	1.00	5	30
40	0.92	6	30
90	0.86	2	28
90	1.00	3	28
90	0.84	4	28
90	0.80	6	28
70	0.79	3	30
70	0.87	4	30
70	1.00	5	30
70	0.85	7	30
80	0.60	4	30
80	0.89	6	30
80	1.00	8	30
80	0.58	9	30
95	0.82	2	30
95	0.90	3	30
95	1.00	6	30
95	0.88	4	30
72	0.78	2	35
72	0.88	4	35
72	1.00	6	35
72	0.82	8	35
0	1.00	1	0
0	1.00	1	0
0	1.00	1	0
0	1.00	1	0
56	0.85	4	45
56	1.00	7	45
56	0.90	6	45
56	0.75	3	45
96	0.80	4	40
96	1.00	8	40
96	0.92	6	40
96	0.72	2	40
84	0.79	2	40
84	0.84	4	40
84	1.00	6	40
84	0.82	3	40
80	0.85	2	40
80	1.00	5	40
80	0.90	4	40
80	0.75	10	40
64	0.89	3	35
64	0.92	4	35
64	1.00	8	35
64	0.75	6	35
56	0.77	2	40

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

56	1.00	4	40
56	0.82	7	40
56	0.79	6	40
72	0.62	4	40
72	0.74	5	40
72	1.00	6	40
72	0.69	3	40
75	0.83	6	45
75	1.00	9	45
75	0.88	7	45
75	0.79	4	45
72	0.79	5	45
72	0.84	4	45
72	1.00	6	45
72	0.72	3	45
64	0.90	4	35
64	1.00	8	35
64	0.88	5	35
64	0.75	3	35
80	0.76	6	35
80	0.90	7	35
80	1.00	5	35
80	0.70	3	35
72	0.80	3	30
72	0.84	4	30
72	1.00	6	30
72	0.82	5	30
56	0.76	3	25
56	1.00	7	25
56	0.86	4	25
56	0.82	5	25
90	0.88	2	35
90	0.92	3	35
90	1.00	5	35
90	0.90	4	35
96	0.74	3	25
96	1.00	8	25
96	0.88	4	25
96	0.79	2	25

File (fileB1.dat)

40	0.79	3	30	16.88	1518.99
40	0.88	4	30	11.36	1363.64
40	1.00	5	30	8.00	1200.00
40	0.92	6	30	7.25	1304.35
90	0.86	2	28	52.33	2930.23
90	1.00	3	28	30.00	2520.00
90	0.84	4	28	26.79	3000.00
90	0.80	6	28	18.75	3150.00
70	0.79	3	30	29.54	2658.23
70	0.87	4	30	20.11	2413.79
70	1.00	5	30	14.00	2100.00
70	0.85	7	30	11.76	2470.59
80	0.60	4	30	33.33	4000.00
80	0.89	6	30	14.98	2696.63
80	1.00	8	30	10.00	2400.00
80	0.58	9	30	15.33	4137.93
95	0.82	2	30	57.93	3475.61
95	0.90	3	30	35.19	3166.67
95	1.00	6	30	15.83	2850.00
95	0.88	4	30	26.99	3238.64
72	0.78	2	35	46.15	3230.77
72	0.88	4	35	20.45	2863.64
72	1.00	6	35	12.00	2520.00
72	0.82	8.	35	10.98	3073.17
0	1.00	1	0	0.00	0.00
0	1.00	1	0	0.00	0.00
0	1.00	1	0	0.00	0.00
0	1.00	1	0	0.00	0.00
56	0.85	4	45	16.47	2964.71
56	1.00	7	45	8.00	2520.00
56	0.90	6	45	10.37	2800.00

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

56	0.75	3	45	24.89	3360.00
96	0.80	4	40	30.00	4800.00
96	1.00	8	40	12.00	3840.00
96	0.92	6	40	17.39	4173.91
96	0.72	2	40	66.67	5333.33
84	0.79	2	40	53.16	4253.16
84	0.84	4	40	25.00	4000.00
84	1.00	6	40	14.00	3360.00
84	0.82	3	40	34.15	4097.56
80	0.85	2	40	47.06	3764.71
80	1.00	5	40	16.00	3200.00
80	0.90	4	40	22.22	3555.56
80	0.75	10	40	10.67	4266.67
64	0.89	3	35	23.97	2516.85
64	0.92	4	35	17.39	2434.78
64	1.00	8	35	8.00	2240.00
64	0.75	6	35	14.22	2986.67
56	0.77	2	40	36.36	2909.09
56	1.00	4	40	14.00	2240.00
56	0.82	7	40	9.76	2731.71
56	0.79	6	40	11.81	2835.44
72	0.62	4	40	29.03	4645.16
72	0.74	5	40	19.46	3891.89
72	1.00	6	40	12.00	2880.00
72	0.69	3	40	34.78	4173.91
75	0.83	6	45	15.06	4066.27
75	1.00	9	45	8.33	3375.00
75	0.88	7	45	12.18	3835.23
75	0.79	4	45	23.73	4272.15
72	0.79	5	45	18.23	4101.27
72	0.84	4	45	21.43	3857.14
72	1.00	6	45	12.00	3240.00
72	0.72	3	45	33.33	4500.00
64	0.90	4	35	17.78	2488.89
64	1.00	8	35	8.00	2240.00
64	0.88	5	35	14.55	2545.45
64	0.75	3	35	28.44	2986.67
80	0.76	6	35	17.54	3684.21
80	0.90	7	35	12.70	3111.11
80	1.00	5	35	16.00	2800.00
80	0.70	3	35	38.10	4000.00
72	0.80	3	30	30.00	2700.00
72	0.84	4	30	21.43	2571.43
72	1.00	6	30	12.00	2160.00
72	0.82	5	30	17.56	2634.15
56	0.76	3	25	24.56	1842.11
56	1.00	7	25	8.00	1400.00
56	0.86	4	25	16.28	1627.91
56	0.82	5	25	13.66	1707.32
90	0.88	2	35	51.14	3579.55
90	0.92	3	35	32.61	3423.91
90	1.00	5	35	18.00	3150.00
90	0.90	4	35	25.00	3500.00
96	0.74	3	25	43.24	3243.24
96	1.00	8	25	12.00	2400.00
96	0.88	4	25	27.27	2727.27
96	0.79	2	25	60.76	3037.97

File (fileB2.dat)

3	16.88	1518.99
4	11.36	1363.64
5	8.00	1200.00
6	7.25	1304.35
2	52.33	2930.23
3	30.00	2520.00
4	26.79	3000.00
6	18.75	3150.00
3	29.54	2658.23
4	20.11	2413.79
5	14.00	2100.00
7	11.76	2470.59
4	33.33	4000.00

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

6	14.98	2696.63
8	10.00	2400.00
9	15.33	4137.93
2	57.93	3475.61
3	35.19	3166.67
6	15.83	2850.00
4	26.99	3238.64
2	46.15	3230.77
4	20.45	2863.64
6	12.00	2520.00
8	10.98	3073.17
1	0.00	0.00
1	0.00	0.00
1	0.00	0.00
1	0.00	0.00
4	16.47	2964.71
7	8.00	2520.00
6	10.37	2800.00
3	24.89	3360.00
4	30.00	4800.00
8	12.00	3840.00
6	17.39	4173.91
2	66.67	5333.33
2	53.16	4253.16
4	25.00	4000.00
6	14.00	3360.00
3	34.15	4097.56
2	47.06	3764.71
5	16.00	3200.00
4	22.22	3555.56
10	10.67	4266.67
3	23.97	2516.85
4	17.39	2434.78
8	8.00	2240.00
6	14.22	2986.67
2	36.36	2909.09
4	14.00	2240.00
7	9.76	2731.71
6	11.81	2835.44
4	29.03	4645.16
5	19.46	3891.89
6	12.00	2880.00
3	34.78	4173.91
6	15.06	4066.27
9	8.33	3375.00
7	12.18	3835.23
4	23.73	4272.15
5	18.23	4101.27
4	21.43	3857.14
6	12.00	3240.00
3	33.33	4500.00
4	17.78	2488.89
8	8.00	2240.00
5	14.55	2545.45
3	28.44	2986.67
6	17.54	3684.21
7	12.70	3111.11
5	16.00	2800.00
3	38.10	4000.00
3	30.00	2700.00
4	21.43	2571.43
6	12.00	2160.00
5	17.56	2634.15
3	24.56	1842.11
7	8.00	1400.00
4	16.28	1627.91
5	13.66	1707.32
2	51.14	3579.55
3	32.61	3423.91
5	18.00	3150.00
4	25.00	3500.00
3	43.24	3243.24
8	12.00	2400.00

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

4	27.27	2727.27
2	60.76	3037.97

File (file83.dat)

3	16.88	1518.99
4	11.36	1363.64
5	8.00	1200.00
6	7.25	1304.35
2	52.33	2930.23
3	30.00	2520.00
4	26.79	3000.00
6	18.75	3150.00
3	29.54	2658.23
4	20.11	2413.79
5	14.00	2100.00
7	11.76	2470.59
4	33.33	4000.00
6	14.98	2696.63
8	10.00	2400.00
9	15.33	4137.93
2	57.93	3475.61
3	35.19	3166.67
6	15.83	2850.00
4	26.99	3238.64
2	46.15	3230.77
4	20.45	2863.64
6	12.00	2520.00
8	10.98	3073.17
1	0.00	0.00
1	0.00	0.00
1	0.00	0.00
1	0.00	0.00
4	16.47	2964.71
7	8.00	2520.00
6	10.37	2800.00
3	24.89	3360.00
4	30.00	4800.00
8	12.00	3840.00
6	17.39	4173.91
2	66.67	5333.33
2	53.16	4253.16
4	25.00	4000.00
6	14.00	3360.00
3	34.15	4097.56
2	47.06	3764.71
5	16.00	3200.00
4	22.22	3555.56
10	10.67	4266.67
3	23.97	2516.85
4	17.39	2434.78
8	8.00	2240.00
6	14.22	2986.67
2	36.36	2909.09
4	14.00	2240.00
7	9.76	2731.71
6	11.81	2835.44
4	29.03	4645.16
5	19.46	3891.89
6	12.00	2880.00
3	34.78	4173.91
6	15.06	4066.27
9	8.33	3375.00
7	12.18	3835.23
4	23.73	4272.15
5	18.23	4101.27
4	21.43	3857.14
6	12.00	3240.00
3	33.33	4500.00
4	17.78	2488.89
8	8.00	2240.00
5	14.55	2545.45
3	28.44	2986.67

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
6 17.54 3684.21
7 12.70 3111.11
5 16.00 2800.00
3 38.10 4000.00
3 30.00 2700.00
4 21.43 2571.43
6 12.00 2160.00
5 17.56 2634.15
3 24.56 1842.11
7  8.00 1400.00
4 16.28 1627.91
5 13.66 1707.32
2 51.14 3579.55
3 32.61 3423.91
5 18.00 3150.00
4 25.00 3500.00
3 43.24 3243.24
8 12.00 2400.00
4 27.27 2727.27
2 60.76 3037.97
```

File (fileB4.dat)

```
3 1.6880000000E+01 1.5189900000E+03
4 1.1360000000E+01 1.3636400000E+03
5 8.0000000000E+00 1.2000000000E+03
6 7.2500000000E+00 1.3043500000E+03
2 5.2330000000E+01 2.9302300000E+03
3 3.0000000000E+01 2.5200000000E+03
4 2.6790000000E+01 3.0000000000E+03
6 1.8750000000E+01 3.1500000000E+03
3 2.9540000000E+01 2.6582300000E+03
4 2.0110000000E+01 2.4137900000E+03
5 1.4000000000E+01 2.1000000000E+03
7 1.1760000000E+01 2.4705900000E+03
4 3.3330000000E+01 4.0000000000E+03
6 1.4980000000E+01 2.6966300000E+03
8 1.0000000000E+01 2.4000000000E+03
9 1.5330000000E+01 4.1379300000E+03
2 5.7930000000E+01 3.4756100000E+03
3 3.5190000000E+01 3.1666700000E+03
6 1.5830000000E+01 2.8500000000E+03
4 2.6990000000E+01 3.2386400000E+03
2 4.6150000000E+01 3.2307700000E+03
4 2.0450000000E+01 2.8636400000E+03
6 1.2000000000E+01 2.5200000000E+03
8 1.0980000000E+01 3.0731700000E+03
1 0.0000000000E+00 0.0000000000E+00
1 0.0000000000E+00 0.0000000000E+00
1 0.0000000000E+00 0.0000000000E+00
1 0.0000000000E+00 0.0000000000E+00
4 1.6470000000E+01 2.9647100000E+03
7 8.0000000000E+00 2.5200000000E+03
6 1.0370000000E+01 2.8000000000E+03
3 2.4890000000E+01 3.3600000000E+03
4 3.0000000000E+01 4.8000000000E+03
8 1.2000000000E+01 3.8400000000E+03
6 1.7390000000E+01 4.1739100000E+03
2 6.6670000000E+01 5.3333300000E+03
2 5.3160000000E+01 4.2531600000E+03
4 2.5000000000E+01 4.0000000000E+03
6 1.4000000000E+01 3.3600000000E+03
3 3.4150000000E+01 4.0975600000E+03
2 4.7060000000E+01 3.7647100000E+03
5 1.6000000000E+01 3.2000000000E+03
4 2.2220000000E+01 3.5555600000E+03
10 1.0670000000E+01 4.2666700000E+03
3 2.3970000000E+01 2.5168500000E+03
4 1.7390000000E+01 2.4347800000E+03
8 8.0000000000E+00 2.2400000000E+03
6 1.4220000000E+01 2.9866700000E+03
2 3.6360000000E+01 2.9090900000E+03
4 1.4000000000E+01 2.2400000000E+03
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
7 9.7600000000E+00 2.7317100000E+03
6 1.1810000000E+01 2.8354400000E+03
4 2.9030000000E+01 4.6451600000E+03
5 1.9460000000E+01 3.8918900000E+03
6 1.2000000000E+01 2.8800000000E+03
3 3.4780000000E+01 4.1739100000E+03
6 1.5060000000E+01 4.0662700000E+03
9 8.3300000000E+00 3.3750000000E+03
7 1.2180000000E+01 3.8352300000E+03
4 2.3730000000E+01 4.2721500000E+03
5 1.8230000000E+01 4.1012700000E+03
4 2.1430000000E+01 3.8571400000E+03
6 1.2000000000E+01 3.2400000000E+03
3 3.3330000000E+01 4.5000000000E+03
4 1.7780000000E+01 2.4888900000E+03
8 8.0000000000E+00 2.2400000000E+03
5 1.4550000000E+01 2.5454500000E+03
3 2.8440000000E+01 2.9866700000E+03
6 1.7540000000E+01 3.6842100000E+03
7 1.2700000000E+01 3.1111100000E+03
5 1.6000000000E+01 2.8000000000E+03
3 3.8100000000E+01 4.0000000000E+03
3 3.0000000000E+01 2.7000000000E+03
4 2.1430000000E+01 2.5714300000E+03
6 1.2000000000E+01 2.1600000000E+03
5 1.7560000000E+01 2.6341500000E+03
3 2.4560000000E+01 1.8421100000E+03
7 8.0000000000E+00 1.4000000000E+03
4 1.6280000000E+01 1.6279100000E+03
5 1.3660000000E+01 1.7073200000E+03
2 5.1140000000E+01 3.5795500000E+03
3 3.2610000000E+01 3.4239100000E+03
5 1.8000000000E+01 3.1500000000E+03
4 2.5000000000E+01 3.5000000000E+03
3 4.3240000000E+01 3.2432400000E+03
8 1.2000000000E+01 2.4000000000E+03
4 2.7270000000E+01 2.7272700000E+03
2 6.0760000000E+01 3.0379700000E+03
```

__File (fileC1.dat)__

```
**************************************
        CS    AH       TC
**************************************
1
        3  16.880  1518.990
        4  11.360  1363.640
        5  8.000   1200.000
        6  7.250   1304.350
****************************************
2
        2  52.330  2930.230
        3  30.000  2520.000
        4  26.790  3000.000
        6  18.750  3150.000
****************************************
3
        3  29.540  2658.230
        4  20.110  2413.790
        5  14.000  2100.000
        7  11.760  2470.590
****************************************
4
        4  33.330  4000.000
        6  14.980  2696.630
        8  10.000  2400.000
        9  15.330  4137.930
****************************************
5
        2  57.930  3475.610
        3  35.190  3166.670
        6  15.830  2850.000
        4  26.990  3238.640
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
****************************************
6
          2  46.150  3230.770
          4  20.450  2863.640
          6  12.000  2520.000
          8  10.980  3073.170
****************************************
7
          1  0.000  0.000
          1  0.000  0.000
          1  0.000  0.000
          1  0.000  0.000
****************************************
8
          4  16.470  2964.710
          7  8.000  2520.000
          6  10.370  2800.000
          3  24.890  3360.000
****************************************
9
          4  30.000  4800.000
          8  12.000  3840.000
          6  17.390  4173.910
          2  66.670  5333.330
****************************************
10
          2  53.160  4253.160
          4  25.000  4000.000
          6  14.000  3360.000
          3  34.150  4097.560
****************************************
11
          2  47.060  3764.710
          5  16.000  3200.000
          4  22.220  3555.560
          10  10.670  4266.670
****************************************
12
          3  23.970  2516.850
          4  17.390  2434.780
          8  8.000  2240.000
          6  14.220  2986.670
****************************************
13
          2  36.360  2909.090
          4  14.000  2240.000
          7  9.760  2731.710
          6  11.810  2835.440
****************************************
14
          4  29.030  4645.160
          5  19.460  3891.890
          6  12.000  2880.000
          3  34.780  4173.910
****************************************
15
          6  15.060  4066.270
          9  8.330  3375.000
          7  12.180  3835.230
          4  23.730  4272.150
****************************************
16
          5  18.230  4101.270
          4  21.430  3857.140
          6  12.000  3240.000
          3  33.330  4500.000
****************************************
17
          4  17.780  2488.890
          8  8.000  2240.000
          5  14.550  2545.450
          3  28.440  2986.670
****************************************
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
18
          6  17.540  3684.210
          7  12.700  3111.110
          5  16.000  2800.000
          3  38.100  4000.000
****************************************
19
          3  30.000  2700.000
          4  21.430  2571.430
          6  12.000  2160.000
          5  17.560  2634.150
****************************************
20
          3  24.560  1842.110
          7  8.000  1400.000
          4  16.280  1627.910
          5  13.660  1707.320
****************************************
21
          2  51.140  3579.550
          3  32.610  3423.910
          5  18.000  3150.000
          4  25.000  3500.000
****************************************
22
          3  43.240  3243.240
          8  12.000  2400.000
          4  27.270  2727.270
          2  60.760  3037.970
****************************************
```

<u>File (fileC2.dat)</u>

```
**********************************
Act.       CS      AH       TC
**********************************
1
             3  16.880  1518.990
             4  11.360  1363.640
             5  8.000  1200.000
             6  7.250  1304.350
==================================
2
             2  52.330  2930.230
             3  30.000  2520.000
             4  26.790  3000.000
             6  18.750  3150.000
==================================
3
             3  29.540  2658.230
             4  20.110  2413.790
             5  14.000  2100.000
             7  11.760  2470.590
=================================
4
             4  33.330  4000.000
             6  14.980  2696.630
             8  10.000  2400.000
             9  15.330  4137.930
=================================
5
             2  57.930  3475.610
             3  35.190  3166.670
             6  15.830  2850.000
             4  26.990  3238.640
=================================
6
             2  46.150  3230.770
             4  20.450  2863.640
             6  12.000  2520.000
             8  10.980  3073.170
=================================
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

7

	1	0.000	0.000
	1	0.000	0.000
	1	0.000	0.000
	1	0.000	0.000

==============================
8

	4	16.470	2964.710
	7	8.000	2520.000
	6	10.370	2800.000
	3	24.890	3360.000

==============================
9

	4	30.000	4800.000
	8	12.000	3840.000
	6	17.390	4173.910
	2	66.670	5333.330

==============================
10

	2	53.160	4253.160
	4	25.000	4000.000
	6	14.000	3360.000
	3	34.150	4097.560

==============================
11

	2	47.060	3764.710
	5	16.000	3200.000
	4	22.220	3555.560
	10	10.670	4266.670

==============================
12

	3	23.970	2516.850
	4	17.390	2434.780
	8	8.000	2240.000
	6	14.220	2986.670

==============================
13

	2	36.360	2909.090
	4	14.000	2240.000
	7	9.760	2731.710
	6	11.810	2835.440

==============================
14

	4	29.030	4645.160
	5	19.460	3891.890
	6	12.000	2880.000
	3	34.780	4173.910

==============================
15

	6	15.060	4066.270
	9	8.330	3375.000
	7	12.180	3835.230
	4	23.730	4272.150

==============================
16

	5	18.230	4101.270
	4	21.430	3857.140
	6	12.000	3240.000
	3	33.330	4500.000

==============================
17

	4	17.780	2488.890
	8	8.000	2240.000
	5	14.550	2545.450
	3	28.440	2986.670

==============================
18

	6	17.540	3684.210
	7	12.700	3111.110
	5	16.000	2800.000
	3	38.100	4000.000

==============================
19

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
              3  30.000  2700.000
              4  21.430  2571.430
              6  12.000  2160.000
              5  17.560  2634.150
==============================
20
              3  24.560  1842.110
              7  8.000  1400.000
              4  16.280  1627.910
              5  13.660  1707.320
==============================
21
              2  51.140  3579.550
              3  32.610  3423.910
              5  18.000  3150.000
              4  25.000  3500.000
==============================
22
              3  43.240  3243.240
              8  12.000  2400.000
              4  27.270  2727.270
              2  60.760  3037.970
==============================
```

File (fileC3.dat)

```
*********************************
Act.   CS     AH      TC
*********************************
1
              3  16.880  1518.990
              4  11.360  1363.640
              5  8.000  1200.000
              6  7.250  1304.350
==============================
2
              2  52.330  2930.230
              3  30.000  2520.000
              4  26.790  3000.000
              6  18.750  3150.000
==============================
3
              3  29.540  2658.230
              4  20.110  2413.790
              5  14.000  2100.000
              7  11.760  2470.590
==============================
4
              4  33.330  4000.000
              6  14.980  2696.630
              8  10.000  2400.000
              9  15.330  4137.930
==============================
5
              2  57.930  3475.610
              3  35.190  3166.670
              6  15.830  2850.000
              4  26.990  3238.640
==============================
6
              2  46.150  3230.770
              4  20.450  2863.640
              6  12.000  2520.000
              8  10.980  3073.170
==============================
7
              1  0.000  0.000
              1  0.000  0.000
              1  0.000  0.000
              1  0.000  0.000
==============================
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

8

4	16.470	2964.710
7	8.000	2520.000
6	10.370	2800.000
3	24.890	3360.000

==============================

9

4	30.000	4800.000
8	12.000	3840.000
6	17.390	4173.910
2	66.670	5333.330

==============================

10

2	53.160	4253.160
4	25.000	4000.000
6	14.000	3360.000
3	34.150	4097.560

==============================

11

2	47.060	3764.710
5	16.000	3200.000
4	22.220	3555.560
10	10.670	4266.670

==============================

12

3	23.970	2516.850
4	17.390	2434.780
8	8.000	2240.000
6	14.220	2986.670

==============================

13

2	36.360	2909.090
4	14.000	2240.000
7	9.760	2731.710
6	11.810	2835.440

==============================

14

4	29.030	4645.160
5	19.460	3891.890
6	12.000	2880.000
3	34.780	4173.910

==============================

15

6	15.060	4066.270
9	8.330	3375.000
7	12.180	3835.230
4	23.730	4272.150

==============================

16

5	18.230	4101.270
4	21.430	3857.140
6	12.000	3240.000
3	33.330	4500.000

==============================

17

4	17.780	2488.890
8	8.000	2240.000
5	14.550	2545.450
3	28.440	2986.670

==============================

18

6	17.540	3684.210
7	12.700	3111.110
5	16.000	2800.000
3	38.100	4000.000

==============================

19

3	30.000	2700.000
4	21.430	2571.430
6	12.000	2160.000
5	17.560	2634.150

==============================

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
20
           3  24.560  1842.110
           7  8.000  1400.000
           4  16.280  1627.910
           5  13.660  1707.320
===============================
21
           2  51.140  3579.550
           3  32.610  3423.910
           5  18.000  3150.000
           4  25.000  3500.000
===============================
22
           3  43.240  3243.240
           8  12.000  2400.000
           4  27.270  2727.270
           2  60.760  3037.970
===============================
```

File (fileC4.dat)

```
***********************************************
Act. CS     AH       TC
***********************************************
1
           3  16.880  1518.990
           4  11.360  1363.640
           5  8.000  1200.000
           6  7.250  1304.350
====================================================
2
           2  52.330  2930.230
           3  30.000  2520.000
           4  26.790  3000.000
           6  18.750  3150.000
====================================================
3
           3  29.540  2658.230
           4  20.110  2413.790
           5  14.000  2100.000
           7  11.760  2470.590
====================================================
4
           4  33.330  4000.000
           6  14.980  2696.630
           8  10.000  2400.000
           9  15.330  4137.930
====================================================
5
           2  57.930  3475.610
           3  35.190  3166.670
           6  15.830  2850.000
           4  26.990  3238.640
====================================================
6
           2  46.150  3230.770
           4  20.450  2863.640
           6  12.000  2520.000
           8  10.980  3073.170
====================================================
7
           1  0.000  0.000
           1  0.000  0.000
           1  0.000  0.000
           1  0.000  0.000
====================================================
8
           4  16.470  2964.710
           7  8.000  2520.000
           6  10.370  2800.000
           3  24.890  3360.000
====================================================
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
9
            4  30.000  4800.000
            8  12.000  3840.000
            6  17.390  4173.910
            2  66.670  5333.330
====================================================
10
            2  53.160  4253.160
            4  25.000  4000.000
            6  14.000  3360.000
            3  34.150  4097.560
====================================================
11
            2  47.060  3764.710
            5  16.000  3200.000
            4  22.220  3555.560
            10  10.670  4266.670
====================================================
12
            3  23.970  2516.850
            4  17.390  2434.780
            8  8.000  2240.000
            6  14.220  2986.670
====================================================
13
            2  36.360  2909.090
            4  14.000  2240.000
            7  9.760  2731.710
            6  11.810  2835.440
====================================================
14
            4  29.030  4645.160
            5  19.460  3891.890
            6  12.000  2880.000
            3  34.780  4173.910
====================================================
15
            6  15.060  4066.270
            9  8.330  3375.000
            7  12.180  3835.230
            4  23.730  4272.150
====================================================
16
            5  18.230  4101.270
            4  21.430  3857.140
            6  12.000  3240.000
            3  33.330  4500.000
====================================================
17
            4  17.780  2488.890
            8  8.000  2240.000
            5  14.550  2545.450
            3  28.440  2986.670
====================================================
18
            6  17.540  3684.210
            7  12.700  3111.110
            5  16.000  2800.000
            3  38.100  4000.000
====================================================
19
            3  30.000  2700.000
            4  21.430  2571.430
            6  12.000  2160.000
            5  17.560  2634.150
====================================================
20
            3  24.560  1842.110
            7  8.000  1400.000
            4  16.280  1627.910
            5  13.660  1707.320
====================================================
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

21

	2	51.140	3579.550
	3	32.610	3423.910
	5	18.000	3150.000
	4	25.000	3500.000

==

22

	3	43.240	3243.240
	8	12.000	2400.000
	4	27.270	2727.270
	2	60.760	3037.970

==

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

MODIFICATIONS NEEDED TO TRANSFER THE PROGRAM TO A BATCH MODE

```
(1)  procedure readinf1;
  var K: integer;
  begin
   assign (fileD1, 'c:prodD1.dat');
   reset (fileD1);
   while not eof (fileD1) do
    begin
      for K := 1 to activities do
      begin
       readln (fileD1,Dur[K],COS[K],RES[K]);
      end;
    end;
   for K := 1 to activities do
     repeat
       write('      Activity (  ', K, '  )  ','     ');
       read(i[K], j[K]);
         if (i[K] >= j[K]) then
            writeln('      Start must be less than finish');
         if (Dur[K] < 0) then
            writeln('      Duration must be positive or zero');
     until (Dur[K] >= 0) and (i[K] < j[K]);
   close (fileD1);
   end {readinf1};


(2)
  assign (fileA1, 'c:proda1.dat');
  reset  (fileA1);
  assign (fileA,'c:proda.dat');
  rewrite(fileA);
  writeln;
 answer1 := 'YES';
 IF answer1 = 'YES' THEN   {...if answer1...}
   begin
      writeln('*************************************');
      writeln('THIS IS THE LEARNING CURVE FEATURE');
      writeln('*************************************');
      writeln;
      writeln('**********************************');
      writeln('PLEASE RESPOND TO EACH QUESTION');
      writeln('**********************************');
      clrscr;
      writeln;
      activities := 22;
      answer3 := 0.80  {learning rate};
      read(answer3);
      if answer3 = 0.85 then
      BEGIN
      for N := 1 to activities do
       begin
         write('What is the execution order of activity  (', N ,' ) ?  >  ');
         read(number);
          if number = 1 then
            begin
              Factor := 1.0;
              for M := 1  to 4 do
              begin
                  readln(fileA1, Act[N].MH[M], Act[N].PR[M], Act[N].CS[M], Act[N].UC[M]);
                  Act[N].MH[M] := Act[N].MH[M] * Factor;
                  writeln(fileA,Act[N].MH[M]:3:0, Act[N].PR[M]:5:2, Act[N].CS[M]:4, Act[N].UC[M]:4:0);
              end; {end of M loop}
            end
          else if number = 2 then
            begin
              Factor := 0.85;
              for M := 1  to 4 do
              begin
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
            readln(fileA1, Act[N].MH[M], Act[N].PR[M], Act[N].CS[M], Act[N].UC[M]);
            Act[N].MH[M] := Act[N].MH[M] * Factor;
            writeln(fileA,Act[N].MH[M]:3:0, Act[N].PR[M]:5:2, Act[N].CS[M]:4, Act[N].UC[M]:4:0);
        end; {end of M loop}
      end
    else if number = 3 then
      begin
        Factor := 0.773;
        for M := 1  to 4 do
         begin
            readln(fileA1, Act[N].MH[M], Act[N].PR[M], Act[N].CS[M], Act[N].UC[M]);
            Act[N].MH[M] := Act[N].MH[M] * Factor;
            writeln(fileA,Act[N].MH[M]:3:0, Act[N].PR[M]:5:2, Act[N].CS[M]:4, Act[N].UC[M]:4:0);
         end; {end of M loop}
      end
    else if number = 4 then
      begin
        Factor := 0.723;
        for M := 1  to 4 do
        begin
            readln(fileA1, Act[N].MH[M], Act[N].PR[M], Act[N].CS[M], Act[N].UC[M]);
            Act[N].MH[M] := Act[N].MH[M] * Factor;
            writeln(fileA,Act[N].MH[M]:3:0, Act[N].PR[M]:5:2, Act[N].CS[M]:4, Act[N].UC[M]:4:0);
        end; {end of M loop}
    end;
end; {end of N loop}
END
else if answer3 = 0.80 then
BEGIN
for N := 1 to activities do
 begin
    write('What is the execution order of activity  (', N ,' ) ?  >  ');
    read(number);
     if number = 1 then
      begin
        Factor := 1.0;
        for M := 1  to 4 do
        begin
            readln(fileA1, Act[N].MH[M], Act[N].PR[M], Act[N].CS[M], Act[N].UC[M]);
            Act[N].MH[M] := Act[N].MH[M] * Factor;
            writeln(fileA,Act[N].MH[M]:3:0, Act[N].PR[M]:5:2, Act[N].CS[M]:4, Act[N].UC[M]:4:0);
        end; {end of M loop}
      end
     else if number = 2 then
      begin
        Factor := 0.80;
        for M := 1  to 4 do
        begin
            readln(fileA1, Act[N].MH[M], Act[N].PR[M], Act[N].CS[M], Act[N].UC[M]);
            Act[N].MH[M] := Act[N].MH[M] * Factor;
            writeln(fileA,Act[N].MH[M]:3:0, Act[N].PR[M]:5:2, Act[N].CS[M]:4, Act[N].UC[M]:4:0);
        end; {end of M loop}
      end
     else if number = 3 then
      begin
        Factor := 0.702;
        for M := 1  to 4 do
         begin
            readln(fileA1, Act[N].MH[M], Act[N].PR[M], Act[N].CS[M], Act[N].UC[M]);
            Act[N].MH[M] := Act[N].MH[M] * Factor;
            writeln(fileA,Act[N].MH[M]:3:0, Act[N].PR[M]:5:2, Act[N].CS[M]:4, Act[N].UC[M]:4:0);
         end; {end of M loop}
      end
     else if number = 4 then
      begin
        Factor := 0.640;
        for M := 1  to 4 do
        begin
            readln(fileA1, Act[N].MH[M], Act[N].PR[M], Act[N].CS[M], Act[N].UC[M]);
            Act[N].MH[M] := Act[N].MH[M] * Factor;
            writeln(fileA,Act[N].MH[M]:3:0, Act[N].PR[M]:5:2, Act[N].CS[M]:4, Act[N].UC[M]:4:0);
        end; {end of M loop}
     end;
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
end; {end of N loop}
END
else if answer3 = 0.75 then
BEGIN
for N := 1 to activities do
 begin
    write('What is the execution order of activity  (', N ,' ) ?  >  ');
    read(number);
     if number = 1 then
       begin
         Factor := 1.0;
         for M := 1  to 4 do
         begin
             readln(fileA1, Act[N].MH[M], Act[N].PR[M], Act[N].CS[M], Act[N].UC[M]);
             Act[N].MH[M] := Act[N].MH[M] * Factor;
             writeln(fileA,Act[N].MH[M]:3:0, Act[N].PR[M]:5:2, Act[N].CS[M]:4, Act[N].UC[M]:4:0);
         end; {end of M loop}
       end
     else if number = 2 then
       begin
         Factor := 0.75;
         for M := 1  to 4 do
         begin
             readln(fileA1, Act[N].MH[M], Act[N].PR[M], Act[N].CS[M], Act[N].UC[M]);
             Act[N].MH[M] := Act[N].MH[M] * Factor;
             writeln(fileA,Act[N].MH[M]:3:0, Act[N].PR[M]:5:2, Act[N].CS[M]:4, Act[N].UC[M]:4:0);
         end; {end of M loop}
       end
     else if number = 3 then
       begin
         Factor := 0.634;
         for M := 1  to 4 do
          begin
             readln(fileA1, Act[N].MH[M], Act[N].PR[M], Act[N].CS[M], Act[N].UC[M]);
             Act[N].MH[M] := Act[N].MH[M] * Factor;
             writeln(fileA,Act[N].MH[M]:3:0, Act[N].PR[M]:5:2, Act[N].CS[M]:4, Act[N].UC[M]:4:0);
          end; {end of M loop}
       end
     else if number = 4 then
       begin
         Factor := 0.562;
         for M := 1  to 4 do
         begin
             readln(fileA1, Act[N].MH[M], Act[N].PR[M], Act[N].CS[M], Act[N].UC[M]);
             Act[N].MH[M] := Act[N].MH[M] * Factor;
             writeln(fileA,Act[N].MH[M]:3:0, Act[N].PR[M]:5:2, Act[N].CS[M]:4, Act[N].UC[M]:4:0);
         end; {end of M loop}
     end;
end; {end of N loop}
END;

close (fileA1);
close (fileA);
```

(3) {******* The Value Matrix Part *******}

```
    clrscr;
    textcolor(3);
    writeln ('***********************************');
    writeln ('THIS IS THE VALUE MATRIX FEATURE');
    writeln ('***********************************');
    writeln;
      begin
        value1 := 0;
        value2 := 0;
        value3 := 0;
        value4 := 0;
        RI := 25;    {for DURATION}
        CD := 110;  {the contractual duration}
        RV1 := (CD/ProjectDur1) * 10.00;
        RV2 := (CD/ProjectDur2) * 10.00;
        RV3 := (CD/ProjectDur3) * 10.00;
        RV4 := (CD/ProjectDur4) * 10.00;
        sub1 := RI * RV1;
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
value1 := sub1 + value1;
sub2 := RI * RV2;
value2 := sub2 + value2;
sub3 := RI * RV3;
value3 := sub3 + value3;
sub4 := RI * RV4;
value4 := sub4 + value4;
RV1 := 0; RV2 := 0; RV3 := 0; RV4 := 0;
writeln;
RI := 25;   {for TOTAL COST}
BUDGET := 50000;  {the project budget}
RV1 := ((BUDGET)/(Cost1)) * 10.00;
RV2 := ((BUDGET)/(Cost2)) * 10.00;
RV3 := ((BUDGET)/(Cost3)) * 10.00;
RV4 := ((BUDGET)/(Cost4)) * 10.00;
sub1 := RI * RV1;
value1 := sub1 + value1;
sub2 := RI * RV2;
value2 := sub2 + value2;
sub3 := RI * RV3;
value3 := sub3 + value3;
sub4 := RI * RV4;
value4 := sub4 + value4;
RV1 := 0; RV2 := 0; RV3 := 0; RV4 := 0;
writeln;
RI := 20;   {for RESOURCE UTILIZATION}
MAXRESOURCE := 1450; {the project MaxResource}
readln (MAXRESOURCE);
RV1 := (MAXRESOURCE/TRU1) * 10.00;
RV2 := (MAXRESOURCE/TRU2) * 10.00;
RV3 := (MAXRESOURCE/TRU3) * 10.00;
RV4 := (MAXRESOURCE/TRU4) * 10.00;
sub1 := RI * RV1;
value1 := sub1 + value1;
sub2 := RI * RV2;
value2 := sub2 + value2;
sub3 := RI * RV3;
value3 := sub3 + value3;
sub4 := RI * RV4;
value4 := sub4 + value4;
RV1 := 0; RV2 := 0; RV3 := 0; RV4 := 0;
writeln;
RI := 15;   {for NETWORK CRITICALITY INDEX}
RV1 := (1 - (NLI1)/(100)) * 10.00;
RV2 := (1 - (NLI2)/(100)) * 10.00;
RV3 := (1 - (NLI3)/(100)) * 10.00;
RV4 := (1 - (NLI4)/(100)) * 10.00;
sub1 := RI * RV1;
value1 := sub1 + value1;
sub2 := RI * RV2;
value2 := sub2 + value2;
sub3 := RI * RV3;
value3 := sub3 + value3;
sub4 := RI * RV4;
value4 := sub4 + value4;
RV1 := 0; RV2 := 0; RV3 := 0; RV4 := 0;
RI := 15;  {for SCHEDULE CRITICALITY INDEX}
RV1 := (1 - (SCI1)/(100)) * 10.00;
RV2 := (1 - (SCI2)/(100)) * 10.00;
RV3 := (1 - (SCI3)/(100)) * 10.00;
RV4 := (1 - (SCI4)/(100)) * 10.00;
sub1 := RI * RV1;
value1 := sub1 + value1;
sub2 := RI * RV2;
value2 := sub2 + value2;
sub3 := RI * RV3;
value3 := sub3 + value3;
sub4 := RI * RV4;
value4 := sub4 + value4;
writeln;
    writeln ('      Value Of The Feasible Schedule        = ', value1:7:3);
    writeln ('      Value Of The Cost_Based Schedule      = ', value2:7:3);
    writeln ('      Value Of The Time_Based Schedule      = ', value3:7:3);
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

```
        writeln ('        Value Of The Resource_Based Schedule = ', value4:7:3);
        writeln;
        writeln ('        The schedule that has the highest value is the best');
        writeln;
end;
```


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Learning-Curve Chart

Unit No.	75% (b = −.415) Unit Time	Total Time	80% (b = −.322) Unit Time	Total Time	85% (b = −.234) Unit Time	Total Time	90% (b = −.152) Unit Time	Total Time
1	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
2	.750	1.750	.800	1.800	.850	1.850	.900	1.900
3	.634	2.384	.702	2.502	.773	2.623	.846	2.746
4	.562	2.946	.640	3.142	.723	3.345	.810	3.556
5	.513	3.459	.596	3.738	.686	4.031	.783	4.339
6	.475	3.934	.562	4.229	.657	4.688	.762	5.101
7	.446	4.380	.534	4.834	.634	5.322	.744	5.845
8	.422	4.802	.512	5.346	.614	5.936	.729	6.574
9	.402	5.204	.493	5.839	.597	6.533	.716	7.290
10	.385	5.589	.477	6.315	.583	7.116	.705	7.994
11	.370	5.958	.462	6.777	.570	7.686	.695	8.689
12	.357	6.315	.449	7.227	.558	8.244	.685	9.374
13	.345	6.660	.438	7.665	.548	8.792	.677	10.05
14	.334	6.994	.428	8.092	.539	9.331	.670	10.72
15	.325	7.319	.418	8.511	.530	9.861	.663	11.38
16	.316	7.635	.410	8.920	.522	10.38	.656	12.04
17	.309	7.944	.402	9.322	.515	10.90	.650	12.69
18	.301	8.245	.394	9.716	.508	11.41	.644	13.33
19	.295	8.540	.387	10.10	.501	11.91	.639	13.97
20	.288	8.828	.381	10.49	.495	12.40	.634	14.61
21	.283	9.111	.375	10.86	.490	12.89	.630	15.24
22	.277	9.388	.370	11.23	.484	13.38	.625	15.86
23	.272	9.660	.364	11.59	.479	13.86	.621	16.48
24	.267	9.928	.359	11.95	.475	14.33	.617	17.10
25	.263	10.19	.355	12.31	.470	14.80	.613	17.71
30	.244	11.45	.335	14.02	.450	17.09	.596	20.73
35	.229	12.62	.318	15.64	.434	19.29	.583	23.67
40	.216	13.72	.305	17.19	.421	21.43	.571	26.54
45	.206	14.77	.294	18.68	.410	23.50	.561	29.37

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.