OPTIMAL RESOURCE ALLOCATION
IN ACTIVITY NETWORKS

By

PRABHAKAR MAHADEO GHARE

Bachelor of Engineering
University of Bombay
Bombay, India
1949

Master of Science
Oklahoma State University
Stillwater, Oklahoma
1963

Submitted to the Faculty of the Graduate School
of the Oklahoma State University
in partial fulfillment of
the requirements for
the degree of
DOCTOR OF PHILOSOPHY
August, 1965

OPTIMAL RESOURCE ALLOCATION

IN ACTIVITY NETWORKS

Thesis Approved:

W. J. Fabrycky
Thesis Adviser

Dean of the Graduate School

PREFACE

This investigation will be based on the assumption that the duration of an activity can be controlled by the modification of resources allocated to that activity. This assumption, although widely accepted as valid, is rarely incorporated in an activity network model. The primary objective of this dissertation will be to present an activity network model based on the assumption of the controllability of activity duration and to show that an optimum allocation procedure can be derived. This proposed model will also differ from the more conventional model in the interpretation of the concept of resources. In the proposed model, the resources are considered as a flow instead of a cost or in units of dollars per unit of time rather than in dollars themselves.

Interest in this area developed in the Spring of 1964 when the writer was studying a course in system theory at Oklahoma State University taught by Dr. Richard Cummins. During the discussion of electrical components and networks the amazing similarity between these activities and activity networks revealed itself. This led to the adaption of the principles of system theory to optimize a project described by an activity network. It is a pleasure to acknowledge

indebtedness to Dr. Richard Cummins who provided the basis for this development.

Thanks are also due to Professors Wilson J. Bentley, Wolter J. Fabrycky, Robert A. Hultquist, Paul A. McCollum, and Paul E. Torgersen for their guidance of my doctoral program and this investigation. Professors Bentley, Fabrycky, and Torgersen also deserve special acknowledgement on a personal basis. Without the help and encouragement given by them from time-to-time, the completion of a four year program of graduate studies would have been impossible. Miss Velda Davis and Mrs. Linda Mackey deserve credit for the excellent job of typing the manuscript. Lastly, but not the least, indebtedness is due to my wife, Shakuntala, for help in ways too numerous to mention.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Figure | Page

CHAPTER I

INTRODUCTION

Two major restrictions confront the decision-maker when a large project is to be planned. First, a certain sequence must be maintained between the activities comprising the project. Second, the total amount of resources available for the execution of the project are limited. Several models exist that may be used to represent the required sequence of activities, but the question of allocating limited resources is rarely treated. The solution proposed in this dissertation is based on the assumption that the duration of an activity can be controlled by the amount of resources allocated. An activity network model is presented that can be used to find the optimum allocation of resources for the project.

Present Project Planning Models

The analysis of a project as an activity network is of comparatively recent origin. An early application of major importance was made by Malcolm, Rosebom, Clark, and Fazzar (1) in 1958 in connection with the Polaris program. The method was called Program Evaluation and Review Technique or

PERT. Another technique known as the Critical Path Method, or CPM, was introduced by Kelley and Walker (2) in 1959. Although other similar methods have been developed, PERT and CPM are the most commonly used in activity network analysis.

The acceptance of these network analysis techniques has been widespread in project management. Three main reasons may be given for the rapid adoption of PERT and CPM. First, the concepts are simple to understand and apply. Second, they provide a convenient means for enforcing objectivity in planning which might otherwise be left to intuition or tradition. Finally, the techniques reduce the complexity of the planning problem by considering the overall project in terms of its component activities.

Both PERT and CPM are concerned only with the sequencing of activities. The problem of limited resources has not been made a part of these models. The aspect of limited resources has been approached in an indirect way by an algorithm known as PERT-COST. This algorithm minimizes the cost of the project by considering the effect of expediting certain activities at the expense of slowing progress on others. The cost problem was also solved by the use of Linear Programming by Charnes and Cooper (3).

The problem of resource allocation in direct form was considered by Weist (4). Kelley (5) considered the same problem as Weist, but used an empirical approach. In each case, the requirements for resources for the project are not assumed to be fixed. A comprehensive summary of the available network planning methods may be found in Muth and Thompson (6) and Moder and Phillips (7). There are no existing methods for controlling the duration of a project by varying the allocation of resources to the activities.

Description of PERT - CPM Models

Both PERT and CPM are very similar in their logical format. All projects are characterized as sets of activities required to complete the project. Activities are characterized by sets of terminal events designated 'start' and 'end'. The physical nature of the project constrains the execution of these activities to some specified order. This specified order gives rise to a large number of precedence and succession relationships between the events and the activities. The principal relationships are:

(1) The 'start' event of an activity precedes the 'end' event for the same activity by a time duration called 'the activity time'.

(2) The 'start' event of an activity succeeds all 'end' events for all activities preceding it.

(3) The event 'project start' precedes all activities and events in the project.

(4) The event 'project end' succeeds all activities and events in the project.

The relationships described above can be expressed graphically in the form of a network. The nodes of the network represent events, and the arcs represent activities. The direction of time flow is shown on each arc from the 'start' towards the 'end'. Thus, all arrows point away from the event 'project start' and towards the event 'project end'. Each event on this network is labeled by a non-negative integer. It is usually more convenient to have $i < j$, whenever event i precedes event j directly or indirectly. However, this is not mathematically necessary. An activity that has its 'start' labeled i and its 'end' labeled j is represented by the double subscript (i,j).

Associated with each activity is an estimate of the expected time required for its completion. This is represented by the lengths of the arcs in the network. The basic difference in CPM and PERT occurs from the different methods of arriving at the time estimate. In CPM only one time estimate is made, and this value, Y_{ij}, is treated as an algebraic variable. In PERT, three time estimates are made:

(1) The probable earliest completion time, a.

(2) Probable longest time for completion, b.

(3) The most probable time for completion, m.

In this case Y_{ij} is defined as a random variable with a Beta distribution and with range from a to b and mode m. The expected value of this distribution is used as the length of the corresponding arc.

Similarly, there are two chronological times $TE_{(i)}$ and $TL_{(i)}$ associated with each event or node of the graph:

(1) $TE_{(i)}$ = the earliest possible time of the

occurrence of event, i, for a given project start, $TE_{(o)}$.

(2) $TL_{(i)}$ = the latest possible time of occurrence of event, i, which would not be incompatible with a given project end, $TL_{(s)}$.

From the network relationships between events and activities, it is possible to develop the following recursive relationships for computing $TE_{(i)}$ and $TL_{(i)}$.

$$TE_{(i)} = \begin{cases} TE_{(o)} & i = o \\ \max(TE_{(k)} + Y_{ki}) & \text{for all } (k,i) \varepsilon P \end{cases}$$

and

$$TL_{(i)} = \begin{cases} TL_{(s)} & i = s \\ \min(TL_{(k)} - Y_{ik}) & \text{for all } (i,k) \varepsilon P. \end{cases}$$

From these equations, it is possible to compute the following information about each activity (i,j) included in the project P:

(1) Earliest starting time, $TE_{(i)}$.

(2) Latest starting time, $TL_{(j)} - Y_{ij}$.

(3) Earliest completion time, $TE_{(i)} + Y_{ij}$.

(4) Latest completion time, $TL_{(j)}$.

(5) Maximum available time, $TL_{(j)} - TE_{(i)}$.

(6) Slack, $TL_{(j)} - TE_{(i)} - Y_{ij}$.

The above information becomes the basic foundation on which management decisions about a particular activity are based. Slack represents the amount of latitude available to the decision-maker in the scheduling of that activity. If the slack is zero, the activity is critical because the sequencing decision is no longer controlled by the decision-maker. In CPM it can be shown that there is always a connected chain of critical activities from project start to project end (8). This is called the critical path. In PERT networks, there is no single critical path, but each activity has a certain probability of being critical. The activities which may become critical with a high probability are considered more critical than those having a smaller probability of becoming critical.

The Question of Limited Resources

In the basic CPM and PERT models, emphasis is given to the sequence of activities and the expected times of the occurrence of events. A major aspect of the decision environment, the limited availability of resources, is usually not made an explicit part of the model except in a few minor ways. However, this does not imply that a successful application of these techniques is possible without the consideration of the limited availability of resources. This consideration has to be implicitly accounted for during implementation. The two most common ways of achieving this are:

(1) Assign some kind of priority ranking to the activities, such as criticality.

(2) Assign a schedule under the assumption of unlimited resource availability and follow the schedule whenever the resources are available, and introduce delays otherwise.

The explicit introduction of the aspect of limited resources in an activity network model is beset with many problems. A few of these are:

(1) Lack of explicit criteria for the evaluation of effectiveness.

(2) Varying policies of resource management in different organizations.

(3) Lack of sufficient knowledge of the exact relationship between resource allocation and the completion time for an activity.

(4) The non-homogeneous and discrete nature of most input resources.

(5) Possible interaction between the sequence of activities and their resource requirements.

In view of the above problems, the formulation of a mathematical model is conceptually difficult, and, at best, complex. However, certain assumptions must be made if formulation is to be attempted. The model developed in this dissertation is based on the following assumptions:

(1) Elapsed time between the start and the end of a project is the measure of effectiveness.

(2) Optimum effectiveness is achieved by minimizing elapsed time.

(3) Resources are considered to be continuously divisible, homogeneous, and interchangeable.

(4) A relationship between resource allocation and activity time exists.

(5) Possible interactions between activities are disregarded.

Project Control Through Resource Allocation

The effectiveness function for a decision situation is presented by Churchman, Ackoff, and Arnoff (9) as

$$E = f(x_i, y_j).$$

Two classes of variables are involved. Those directly under control of the decision-maker are designated x_i and those not directly under his control are designated y_j. Barnard (10) defines all factors upon which the outcome depends as limiting factors. Those limiting factors which can be successfully altered to modify effectiveness are called strategic factors. Thus, both the x_i variables and the y_j variables are limiting factors. Only the x_i variables are strategic factors.

In an activity network model, the sequence relations and resource limitations are factors over which the decision-maker has no control. These are the states of nature, y_j. On the other hand, the individual resource allocations to the activities are directly under management control. These are the strategic factors, x_i. In the present models for activity network planning only the environmental factors y_j are considered. The decision-maker has no positive control over the duration of the activities comprising the project.

In this dissertation, both types of variables are incorporated in the model. This allows the decision-maker to control the activities in the project in order to optimize effectiveness rather than merely following a schedule determined by the environmental factors. This new approach is the major contribution of this investigation. Although the mathematical model presented is restricted by its assumptions, the insight it provides should be of value in project management.

CHAPTER II

FUNDAMENTAL PREREQUISITES

When the effectiveness measure is the elapsed time between the event 'project start' and 'project end', intuitive considerations indicate that optimal resource allocation would be that allocation which forces simultaneous completion of all activities preceding an event. This chapter will justify this intuitive reasoning. In addition, the concept of slack and critical path defined for the basic CPM model will be shown to hold for the resource allocation model under development. Finally, the nature of the resource allocation-activity duration concept will be explored with the objective of establishing certain criteria for optimal resource allocation.

The Basic Resource-Time Function

In the planning of a project, it is generally accepted that there is an inverse relationship between the amount of resources allocated to an activity and the time needed for its completion. This means that an activity could be expedited by allocating additional resources. Conversely, the completion of an activity could be delayed by curtailing the resources allocated. For example, if the resource is labor, the activity can be expedited by hiring additional labor or delayed by laying off part of the labor force.

Control of the activity completion time through the modification of allocated resources has certain limitations. For example, if laying-off of labor leaves too few workers, it might not be possible to complete the job at all regardless of the time duration involved. At the other extreme, if too many workmen are employed, the activity may not be expedited beyond a certain limit. The additional workmen may only be in each others way and not contribute much to the completion of the activity. Thus, there would be a limiting restriction on the useful resource allocation at the extremes. Thus, there may be a range of resource allocation between the feasible minimum and a feasible maximum within which control would be effected. The resource-time relationship together with its limitations is the basis of the resource allocation model. Assumptions regarding this relationship are given in the paragraphs which follow.

Assumption 1. For every activity A_k included in the project P, there exists a smallest possible resource allocation $R_k(min)$ such that for any resource allocation R_k less than $R_k(min)$, it is physically impossible to complete the activity. The time necessary for the completion of A_k with the minimum resource allocation $R_k(min)$ is designated $t_k(max)$. Thus, $t_k(max)$ is the latest completion time for A_k and it would not be possible to prolong the activity any further through the adjustment of resource allocation.

<u>Assumption 2</u>. For every activity A_k included in the project P, there exists a shortest possible time $t_k(\text{min})$ such that for any time t_k less than $t_k(\text{min})$, it would not be possible to complete the activity. The resource allocation necessary for the completion of A_k in the shortest time $t_k(\text{min})$ is designated $R_k(\text{max})$. Any allocation R_k greater than $R_k(\text{max})$ would not expedite the completion of A_k further than $t_k(\text{min})$. The extra resource allocation $R_k - R_k(\text{max})$ would be idle, and only $R_k(\text{max})$ would be utilized in the completion of the activity A_k.

<u>Assumption 3</u>. The region of feasibility is defined as the region within the limits

$$t_k(\text{min}) \leq t_k \leq t_k(\text{max})$$

and

$$R_k(\text{min}) \leq R_k \leq R_k(\text{max}).$$

Within this region of feasibility, there exists a unique relation between the resource allocation R_k and the time required for the completion of the activity t_k. This relation is a continuous monotonically decreasing function. Thus, if R_k decreases t_k increases, and if R_k increases t_k decreases. The comparison of two alternative resource allocations $R_{k.1}$ and $R_{k.2}$ for the same activity A_k under Assumption 3 indicates that

$$R_{k.1} < R_{k.2} \quad \Longleftrightarrow \quad t_{k.1} > t_{k.2}$$

and

$$R_{k.1} > R_{k.2} \quad \Longleftrightarrow \quad t_{k.1} < t_{k.2}.$$

The function described by Assumptions 1, 2, and 3 is illustrated in Figure 2.1. The relationship assumed between the resource allocation and the completion time can be used to establish certain rules for optimum resource allocation when the total resource availability in a time period immediately preceding a terminal event is limited. Optimum effectiveness in this case would be the earliest chronological occurrence of the particular terminal event.

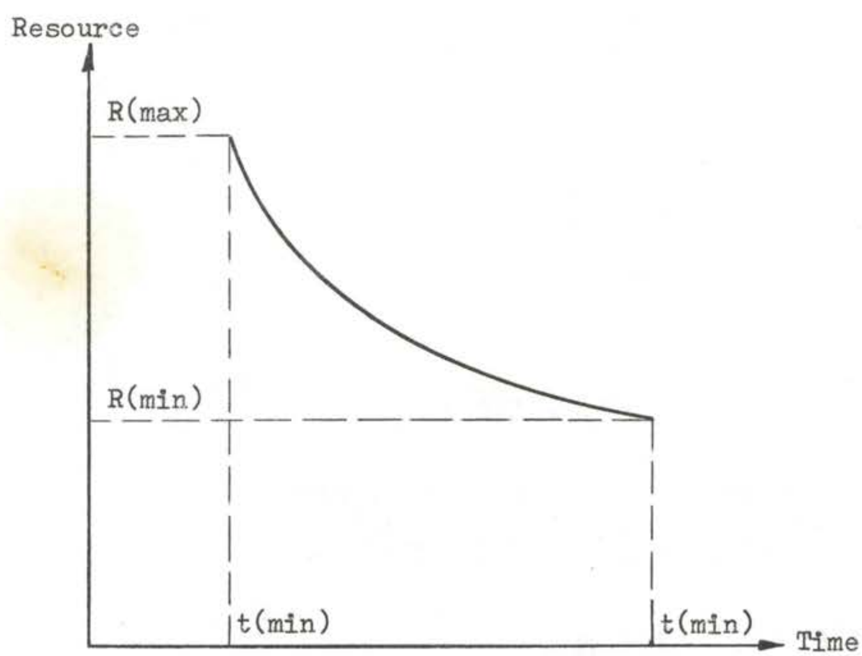

Figure 2.1. Resource-Time Function

Case 1 - Two Activities Starting at the Same Time

Let $P = \{A_1, A_2\}$ be a project consisting of two activities, A_1 and A_2, having the events E_1 and E_2 as its project start and project end, respectively. This situation is illustrated in Figure 2.2.

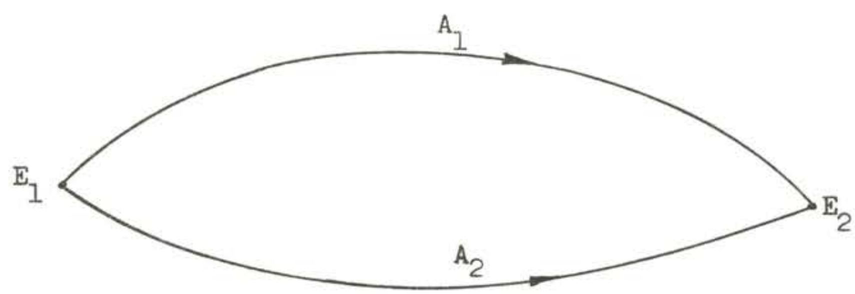

Figure 2.2. Two Activities Starting at the Same Time

Also, let

TE_1 = time of event E_1

TE_2 = time of event E_2

t_1 = duration of activity A_1

t_2 = duration of activity A_2

R_1 = resource allocation for A_1

R_2 = resource allocation for A_2

R = aggregate resource availability.

From the precedence-succession relationship, it can be seen that $TE_2 = TE_1 + \max(t_1, t_2)$. And, the aggregate resource limitations can be expressed as $R_1 + R_2 \leq R$. Optimum effectiveness would be the least time execution of the project, subject to this aggregate resource restriction, or the minimum possible elapsed time $TE_2 - TE_1$. The problem can now be restated as

$$\text{Minimize} \quad \max\left\{t_1, t_2\right\}$$

$$\text{Subject to} \quad R_1 + R_2 \leq R.$$

<u>Theorem 1</u>: For two activities starting and ending at the same time, the resource allocation will be optimum when the corresponding t_1 and t_2 are equal, provided the allocations R_1 and R_2 are within the limits of feasibility for the respective activities.

<u>Proof</u>: Let R_1 and R_2 be allocations for which the corresponding activity times t_1 and t_2 are equal. Let there be another allocation, (R_1* and R_2*) and let the corresponding activity times be (t_1* and t_2*). Also, let $R_1* > R_1$. Then, since $R_1 + R_2 = R_1* + R_2*$, $R_1* > R_1$ implies $R_2* < R_2$. Correspondingly, from the monotonically decreasing resource-time relationship, these two inequalities imply that $t_1* < t_1$ and $t_2* > t_2$ or $t_1* = t_1 + \delta_1$ and $t_2* = t_2 - \delta_2$, where δ_1 and δ_2 are positive. Therefore, $\max\left\{t_1*, t_2*\right\} = \max\left\{t_1 + \delta_1, t_2 - \delta_2\right\}$. But since $t_1 = t_2$, $\max\left\{t_1*, t_2^*\right\} = \max\left\{t_1 + \delta_1, t_2 - \delta_2\right\} = t_1 + \delta_1$. But $\max\left\{t_1, t_2\right\} = \max\left\{t_1, t_1\right\} = t_1$ or $\max\left\{t_1*, t_2*\right\} > \max\left\{t_1, t_2\right\}$. A similar result may be obtained by letting $R_2* > R_2$.

Therefore, no other allocation (R_1^*, R_2^*) could result in a shorter completion time for the two-activity project, P, and the theorem is proved. In this optimum allocation, there is no slack time.

Theorem 2: If $t_1(\max)$ is less than $t_2(\max)$, and if t_2 corresponding to an allocation $R_2 = R - R_1(\min)$ is greater than $t_1(\max)$ but smaller than $t_2(\max)$, $\{R_1(\min), R - R_1(\min)\}$ represents an optimum allocation.

Proof: Since $t_2 < t_2(\max)$, the optimum solution is feasible. Let there be any other allocation R_1^* and R_2^*, and let the corresponding activity times be t_1^* and t_2^*. If $R_1^* < R_1(\min)$, the activity A_1 can never be completed and, consequently, the allocation is not feasible. If $R_1^* > R_1(\min)$, then $R_2^* = R - R_1^* < R - R_1(\min)$ or $R_2^* < R_2$. Also, $R_1^* > R_1(\min)$ implies that $t_1^* < t_1(\max)$, and $R_2^* < R_2$ implies that $t_2^* > t_2$. Since $t_2 > t_1(\max)$ it follows that $t_2^* > t_2 > t_1(\max) > t_1^*$. Therefore, $\max\{t_2^*, t_1^*\} = t_2^*$, $\max\{t_2, t_1(\max)\} = t_2$, and $\max\{t_2^*, t_1^*\} > \max\{t_2, t_1(\max)\}$. Thus, no other allocation (R_1^*, R_2^*) could result in a shorter completion time for the two activity project, P, and the theorem is proved. In this optimum allocation, there exists a slack in the activity A_1.

If the parameters of the activities and the resource restriction do not satisfy the conditions for either Theorem 1 or Theorem 2, there is no feasible optimum allocation set $\{R_1, R_2\}$. This is due to one of the following reasons

(1) The aggregate resource availability is too

small to enable a simultaneous execution of the two activities. This happens when $R < R_1(\min) + R_2(\min)$.

(2) The aggregate resource availability is too large, and there are no optimum allocations that can utilize all the available resources. In this case, the optimum allocation set includes some idle resources in addition to maximum allocations for the activities. However, in this second case, there does exist an optimum allocation at a lower level of aggregate resource restriction. The proof of these statements follows the same pattern of that for Theorem 1 and Theorem 2.

It is now possible to compute the slack in activities A_1 and A_2 comprising a two activity project under the conditions of optimum allocation. The slack of $A_1 = TE_2 - TE_1 - t_1$, and the slack of $A_2 = TE_2 - TE_1 - t_2$. Substituting the value $\max\{t_1, t_2\}$ for $TE_2 - TE_1$ the slack of $A_1 = \max\{t_1, t_2\} - t_1$ and the slack of $A_2 = \max\{t_1, t_2\} - t_2$.

If the optimization of the project were under the conditions of Theorem 1, $t_1 = t_2$ and both slacks are zero. If optimization were under the conditions of Theorem 2, then $t_2 > t_1(\max)$ and $t_1 = t_1(\max)$ and, hence, the slack of A_2 is zero and the slack of A_1 is $t_2 - t_1(\max)$. The slack of $A_1 = t_2 - t_1(\max) = \max\{t_2, t_1(\max)\} - t_1(\max) = TE_2 - TE_1 - t_1(\max)$. These slack computations under the conditions of Theorem 1 and Theorem 2 can be summarized in the following important theorem.

Theorem 3: Slack exists following an activity A_k in a project with an optimum resource allocation, only if $TE_j - TE_i$ excedes $t_k(max)$.

Case 2 - Two Activities Having Different Starting Times

Let $P = \{A_1, A_2\}$ be a project consisting of two activities, A_1 and A_2, and having the event E_3 as its 'project end'. Let E_1 and E_2 be the starting events of activities A_1 and A_2, respectively. This situation is illustrated in Figure 2.3.

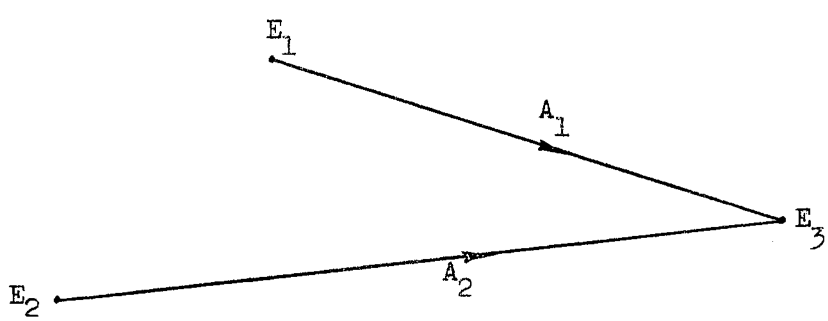

Figure 2.3. Two Activities Starting at Different Times

Also, let

TE_1 = time of event E_1

TE_2 = time of event E_2

TE_3 = time of event E_3

t_1 = duration of activity A_1

t_2 = duration of activity A_2

R_1 = resource allocation for A_1

R_2 = resource allocation for A_2.

From the precedence-succession relationship, it can be seen that $TE_3 = \max(TE_1 + t_1, TE_2 + t_2)$. And, the aggregate resource limitation can be expressed as $R_1 + R_2 \leq R$. Optimum effectiveness would be the least time execution of the project, or the earliest cronological event E_3 for a given set of starting events E_1 and E_2.

Theorem 4: For two activities starting at different times, the resource allocation would be optimum when $TE_1 + t_1 = TE_2 + t_2$ provided the corresponding resource allocations R_1 and R_2 are within the limits of feasibility for the respective activities.

Proof: Let R_1 and R_2 be the allocations for which $TE_1 + t_1 = TE_2 + t_2$. Let there be any other allocation R_1 and R_2, and let the corresponding activity times be t_1* and t_2*. Also, let R_1* be greater than R_1. Then, since $R_1 + R_2 = R = R_1* + R_2*$ and $R_1* > R_1$ implies $R_2* < R_2$. Correspondingly, from the monotonically decreasing resource-time relationship these two inequalities imply that $t_1* < t_1$ and $t_2* > t_2$ or $t_1* = t_1 + \delta_1$

and $t_2^* = t_2 - \delta_2$. Where δ_1 and δ_2 are positive. Therefore, $\max\{TE_1 + t_1^*, TE_2 + t_2^*\} = \max\{TE_1 + t_1 + \delta_1, TE_2 + t_2 - \delta_2\}$. But, $TE_1 + t_1 = TE_2 + t_2$. Hence, $\max\{TE_1 + t_1 + \delta_1, TE_2 + t_2 - \delta_2 = TE_1 + t_1 + \delta_1$. This amount is greater than $TE_1 + t_1$ since δ_1 is positive or, $\max\{TE_1 + t_1^*, TE_2 + t_2^*\} > \{TE_1 + t_1, TE_2 + t_2\}$. A similar result would be obtained by letting R_2^* be greater than R_2. Therefore, no other allocation (R_1^*, R_2^*) could result in an earlier occurrence of event E_3, and the theorem is proved. In this optimum allocation, there is no slack time.

Theorem 5: If $TE_1 + t_1(max)$ is less than $TE_2 + t_2(max)$, and if TE_3 corresponding to an allocation $R_2 = R - R_1(min)$ is greater than $TE_1 + t_1(max)$, but smaller than $TE_2 + t_2(max)$, $\{R_1(min), R - R_1(min)\}$ represents an optimum allocation.

Proof: Let there be another allocation R_1^* and R_2^*, and let the corresponding activity times be t_1^* and t_2^*. If $R_1^* < R_1(min)$, the activity A_1 can never be completed, and, consequently, the allocation is not feasible. If $R_1^* > R_1(min)$, this implies that R_2^* is less than $R - R_1(min)$, or $R_2^* < R_2$. Also, $R_1^* > R_1(min)$ implies that $t_1^* < t_1(max)$, and $R_2^* < R_2$ implies that $t_2^* > t_2$. Since, $TE_2 + t_2 > TE_1 + t_1(max)$, $TE_2 + t_2^* > TE_2 + t_2 > TE_1 + t_1(max) > TE_1 + t_1^*$. Therefore, $\max\{TE_2 + t_2^*, TE_1 + t_1^*\} = TE_2 + t_2^*$, $\max\{TE_2 + t_2, TE_1 + t_1(max)\} = TE_2 + t_2$, and $TE_3^* > TE_3$. Thus, no other allocation R_1^*, R_2^* could result in an earlier occurrence of event E_3 and the theorem is proved. In this optimum allocation, there exists a slack in the activity A_1. Under all other conditions of the parameters of the activities and resource restrictions, there is no optimum allocation set $\{R_1, R_2\}$ as explained before.

The Critical Path With Optimum Allocation

The conclusions in the previous section, although derived for restricted cases, can easily be extended. In cases where there are multiple (more than two) activities preincident at an event, Theorems 1, 2, 4, and 5 can be repeatedly applied for pairwise optimization of the resource allocation. The conditions for the existence of slack (Theorems 2 and 5) and of no slack (Theorems 1 and 4) remain invariant through these repeated applications.

By successive application of these principles to all events included in a project, starting with the event 'project end' and proceeding backwards, excluding the event 'project start', the allocation of resources to all activities in the project can be optimized. An algorithm for accomplishing this based on the principles of network analysis of system theory is described in subsequent chapters. This extension of optimization principles to the entire activity network, when viewed in light of Theorem 3, leads to the following important corrolaries regarding an optimum feasible resource allocation:

(1) Among all activities preincident at an event E_j, the activity for which the quantity $TE_i + t_k$ is a maximum (E_i and E_j being the terminal events of activity A_k) does not have a slack.

(2) For each event, other than 'project start', there exists at least one preincident activity which does not have a slack. All such activities can be called "critical."

(3) There exists at least one critical path in an activity network under the conditions of optimum resource allocation, if a critical path is defined as an unbroken sequence of critical activities from 'project start' to 'project end'.

At this point, a remarkable similarity is noticed in the above corrolaries and the corresponding conclusions obtained by Kelley (2) and Levy, Thompson, and Weist (8) for the basic critical path method (CPM) model. It can be said that these are invariants in the optimization of the resource allocation.

CHAPTER III

AN ANALOGY FROM SYSTEM THEORY

Many important innovations in management science result from the discovery of analogies. This search enables existing solution methods from other fields having similar or analogous characteristics to be utilized without unnecessary duplication of research effort. An analogy from system theory is disucssed in this chapter. The first section will describe the similarity between the electromechanical systems and the activity systems, the second will describe the nature of an activity as a two-terminal system component, and the third will describe a necessary modification required for the use of system theory.

Similarity Between Electromechanical and Activity Systems

The composite system of an activity network together with its relationships and resource allocations is quite complex. In recent years, system theory has been well developed for the analysis of complex systems. To quote Koenig and Blackwell (11):

> If the systems to be analyzed were composed of only two-terminal components, with mathematical

> counterparts in electrical circuit theory, there would be no real need to search for a more general analysis procedure.

Because of the generality of the procedure, its adaption to the system of activities and resource allocations is promising. However, a critical analysis of the mathematical characteristics of the system components is necessary before any such adaption can be attempted.

Koenig and Blackwell (11) state the following necessary prerequisites for the analysis of a physical system:

(1) A mathematical description of each component.

(2) A mathematical description of how the components are combined to form a system.

The components of the activity-resource allocation systems are the individual activities as well as the slacks at the terminations of the activities. The fundamental characteristics of these components that need analysis are the resource allocation and the time duration. In the case of individual activities, the mathematical relationship between these two is uniquely defined.

In the case of slacks, the resource allocation is determined by the preincident activity and time duration by the two events, the end of the preincident activity, and the start of the postincident activity. Thus, the mathematical description for both types of components is complete, and the first prerequisite is satisfied.

A complete description of how the components, activities, and slacks are combined to form the system is inherent in the formulation of the activity network. Thus, the second prerequisite is also satisfied.

Besides these prerequisites, the variables of the analysis, the resource allocation and the time duration, are analogous to the variables in the analysis of an electrical circuit, the current and the e.m.f. The resource allocation is actually a flow of resources and would be very similar mathematically to the electric current which is a flow of electrons. The time duration of an activity is a measurement taken only with respect to the two-terminal events of the activity and would be similar in characteristic to the electrical e.m.f. which is also measured in the same fashion. Thus, the analogy between the electromechanical system and the activity-resource allocation system is complete.

The analytical procedures of system theory can be confidently applied to the activity-resource allocation system. However, one important difference in the two systems must be recognized although it does not present any mathematical difficulty in the application. This difference lies in the fact that the variables of the electrical system are time-dependent, whereas time itself is a variable in the activity-resource allocation system.

Activity as a Two-Terminal Component

The representation of individual activities as system components takes the form of an oriented line segment. The two terminals of the line segment would be representative of the two events 'start' and 'end' for the activity as shown in Figure 3.1. The orientation of the line segment would be dictated by the direction of time flow. The measurement of the time duration of the activity could be achieved hypothetically by placing a conceptual time-meter, T, across the two terminal events in a manner similar to the connection of a voltmeter. The resource allocation could be measured by placing an imaginary resource-allocation-meter, R, immediately at the end of, and in series with, the oriented line segment similar to the connection of an ammeter. Figure 3.1 then represents the terminal graph of the activity as a two terminal component.

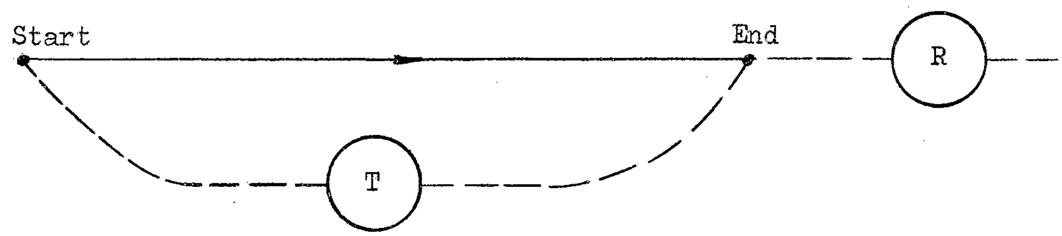

Figure 3.1. A Two-Terminal Component

The terminal equations, which record the mathematical characteristics of the two-terminal component can then be stated as follows for real activities:

$$\begin{aligned} T &= T(\max) \quad \text{if} \quad R = R(\min) \\ &= T(\min) \quad \text{if} \quad R \geq R(\max) \\ &= \Phi(R) \quad \text{if} \quad R(\min) \leq R \leq R(\max). \end{aligned} \qquad (3.1)$$

And, for slack activities,

$$R = R_D. \tag{3.2}$$

In Equation (3.2), the subscript D indicates a driver, absorber, or any other specified function. In the case of slack activities, the resource allocation is uniquely defined as being equal to the resource allocation of the pre-incident activity. The nature of slack activities is that of an absorber, or reverse driver component. It should be noted that the component terminal equations are usually non-linear but uni-valued and, consequently, have uniquely defined inverses. This will be very helpful in the analytical solution of the system.

The representation of activities individually is simply a collection of oriented line segments. However, a complete project or activity network is an integrated system. The entire system, including the precedence-succession relationships between these activities must, therefore, be represented in the form of a linear graph. Hence, in the subsequent analysis of the system, heavy reliance must be placed on the definitions, theorems, and postulates of linear-graph theory. These are usually expounded in great detail in many leading books in the fields of system theory, operations research, and mathematics (11), (12), (13), (14). A short summary of these is included in the Appendix for reference purposes.

Minimum Allocation Diagram

The time required for the completion of an activity is dependent upon the resource allocation to that activity. If the activity is to be represented by a line, and if the length of the line is to represent the magnitude of the time required, there would be many possible time representations of a single activity. Thus, there could be infinitely many representations or diagrams for every activity network, depending upon the individual resource allocations. One such representation would be optimal with the elapsed time between the 'project start' and the 'project end' being a minimum.

Another such diagram would result if it were decided to allocate the minimum possible amount of resources to all activities. This would be called a minimum allocation diagram. Such a minimum allocation diagram has some very useful properties and could be used as the effectiveness measure of the system. This would be a necessary complement to the analysis of the system with the help of system theory as the usual procedures of system theory do not entail an effectiveness function. These properties of the minimum allocation diagram can be described as follows:

(1) As the time duration for each activity is uniquely defined in a minimum allocation diagram, it can be concluded that there is at least one preincident activity at each

event which does not have slack. Such a no-slack activity could again be called critical. As each event has at least one preincident critical activity, the following important conclusion can be drawn. In a minimum allocation diagram, there exists a tree, all branches of which are critical. This would be called minimum-critical-tree.

(2) All chords of the minimum critical tree comprise of preincident activities at various events. Consequently, all of them would have non-negative slack. The slack would be zero if there are more than one preincident critical activities for any event. Otherwise, the slack would be positive.

(3) It can be noted that the events in a minimum allocation diagram are not necessarily balanced for the flow of the resource allocation. This means that the sum total of the resource allocations of all the preincident activities does not necessarily equal the sum total for all the postincident activities, or in other words, the cutset equations are not satisfied. However, this does not constitute a major obstacle in the usefulness of the minimum allocation diagram, because attention is centered

on the minimum critical tree only, and no reference is made to the cutset equations.

Once a minimum critical tree is found from a minimum allocation diagram, it forms the basis for the formulation of system equations. The choice of this particular tree is dependent on the fact that if an activity is critical in a minimum allocation diagram, it would also be critical in any optimum allocation diagram. This can be stated as the following theorem.

Theorem 6: In an optimum allocation diagram, there can be no positive slack in any activity which was represented by a branch of a minimum critical tree.

Proof: Let Figure 3.2 represent one of the fundamental circuits in the minimum allocation diagram. Activities A_1, A_2, and A_3 are the branches of the minimum-critical-tree and A_4 is one of the chords of the tree, with the slack in A_4 being A_5.

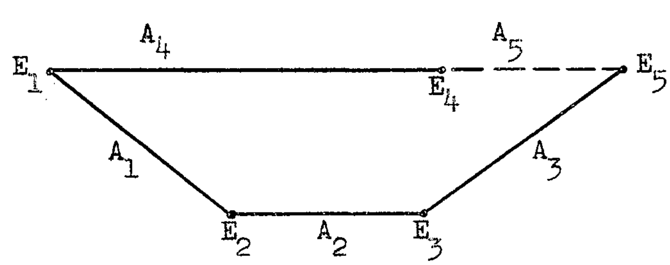

Figure 3.2. A Circuit in the Minimum Allocation Diagram

For any resource allocation set (R_1, R_2, R_3, R_4, R_5), the following quantities would be constrained due to the position of this fundamental circuit in the total system graph, and the necessity of satisfying the cutset equations at each of the four events:

(1) $R_1 + R_4$ would be constrained by the incident resource allocations at E_1.

(2) $R_4 = R_5$.

(3) $R_4 + R_3$ would be constrained by the incident resource allocations at E_5.

(4) $R_2 - R_1$ would be constrained by the incident resource allocations at E_2.

(5) $R_3 - R_2$ would be constrained by the incident resource allocations at E_3.

Now, let there be an optimum allocation set (R_1*, R_2*, R_3*, R_4*) and let this set of resource allocations result in a slack in activity A_3. The fundamental circuit can now be represented as shown in Figure 3.3. The slack in activity A_3 can be represented as a slack activity A_6. Since the slack is non-negative, $TE_4 - TE_1 > TE_5 - TE_1$ or $t_4 > t_1 + t_2 + t_3$. Also, since the schedule is feasible, $t_4 < t_4(\max)$ and $R_4 > R_4(\min)$. Let there be another resource allocation set ($R_1* - \delta$, $R_2* - \delta$, $R_3* - \delta$, $R_4* + \delta$) and let the corresponding activity times be t_1*, t_2*, t_3*, and t_4*, respectively. Then

$$R_1* - \delta + R_4* + \delta = R_1* + R_4*$$

$$R_4* + \delta + R_3* - \delta = R_4* + R_3*$$

$$R_2^* - \delta - R_1^* + \delta = R_2^* - R_1^*$$
$$R_3^* - \delta - R_2^* + \delta = R_3^* - R_2^*.$$

This implies that if constraints 1, 2, 3, 4, and 5 have been satisfied by the allocation set $(R_1^*, R_2^*, R_3^*, R_4^*)$, then they are also satisfied by $(R_1^* - \delta, R_2^* - \delta, R_3^* - \delta, R_4^* + \delta)$.

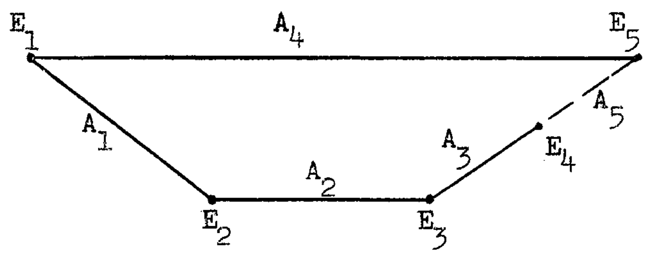

Figure 3.3. Revised Circuit

If δ is a small positive quantity, the monotonically decreasing resource allocation-activity time relationship implies that $t_4^* < t_4$. Also, δ can be chosen small enough so that the relationship $t_4^* > t_1^* + t_2^* + t_3^*$ is not violated. Then, it can be concluded that $t_4^* < t_4$ or $\max\{t_4^*, t_1^* + t_2^* + t_3^*\} < \max\{t_4, t_1 + t_2 + t_3\}$. Also, $TE_4^* - TE_1^* < TE_4 - TE_1$ or $(R_1^*, R_2^*, R_3^*, R_4^*)$ is not an optimum resource allocation set and the theorem is proved.

CHAPTER IV

FORMULATION OF THE SYSTEM MODEL

The objective of this chapter is to set up a procedure for writing the entire set of system relationships algebraically so that a solution may be effected. In so doing, the methods of system theory will be applied with the modifications of Chapters II and III. The procedure will be described step-by-step with the aid of a numerical example. The first section will describe the example problem under consideration, the second will present a proper solution tree, the third will give the system equations, and the fourth will present the mathematical programming format. A simplex solution for the example of this chapter is given in Chapter V.

Description of the Problem

Figure 4.1 is the network representation of a project made up of eight activities, A_1, A_2, A_3, A_4, A_5, A_6, A_7, and A_8. The resource allocation-activity time relationships of the individual activities are given in Table IV-I. In the region of feasibility between the extreme points of Table IV-I, the resource allocation-activity time function is assumed to be linear. This assumption would not normally be true but is used in this illustrative example. This will make the final mathematical programming format linear. Thus, solution by the simplex procedure will be possible.

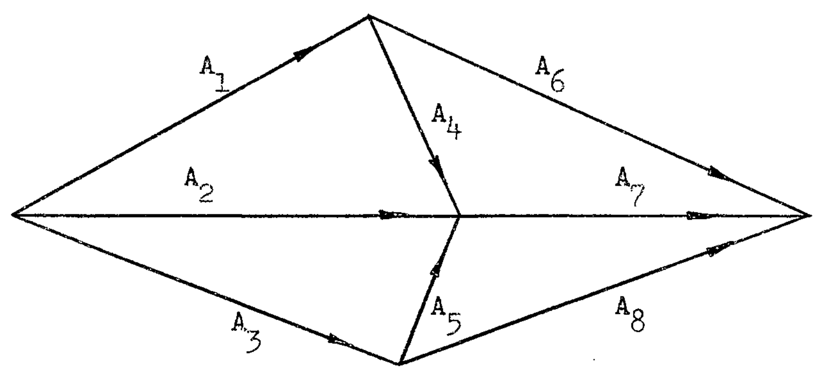

Figure 4.1. Network Diagram of a Project

TABLE IV-I

RESOURCE TIME FUNCTIONS FOR THE ACTIVITIES

Activity	T(max)	T(min)	R(min)	R(max)
A_1	20	10	10	30
A_2	30	6	20	40
A_3	15	10	20	40
A_4	20	10	15	40
A_5	20	10	20	30
A_6	30	20	10	40
A_7	40	20	20	30
A_8	30	10	15	40

If Θ is defined as the variable $\Theta = T - T(min)$ for each activity, the following relationships are evident for any activity

$$\Theta(min) = 0 \tag{4.1}$$

$$\Theta(max) = T(max) - T(min). \tag{4.2}$$

The values of $\Theta(max)$ for the eight activities in the project are given in Table IV-II. The problem consists of determining the optimum resource allocations to the individual activities that will result in the least time execution of the project if the maximum availability of resources is 80 units.

TABLE IV-II

$\Theta(max)$ FOR THE ACTIVITIES

Activity	$\Theta(max)$
A_1	10
A_2	24
A_3	5
A_4	10
A_5	10
A_6	10
A_7	20
A_8	20

Selection of the Solution Tree

Step 1. Draw the basic system in the form of a linear graph of the form illustrated in Figure 4.1 (page 34). If the project consists of N_a activities, this linear graph will have N_a edges. If these N_a activities describe N_v events, the graph would have N_v vertices. In the present example, N_a is eight and N_v is five. This graph is called the basic linear graph.

Step 2. Draw the minimum allocation diagram as described in Chapter III. If the basic linear graph has N_a edges and N_v vertices, the minimum allocation diagram would also have N_a edges and N_v vertices. This diagram is shown in Figure 4.2 for the example under consideration. The time associated with each activity in this diagram is the maximum feasible time for that activity and is the time shown in column T(max) of Table IV-I (page 34). This time is shown in parenthesis on the line representing that in Figure 4.2. The dotted lines represent the slacks and the figures in circles adjacent to each vertex show the time of the occurrence of that event under the conditions of minimum resource allocation. (See Figure 4.2 on the following page.)

Step 3. Determine the minimum critical tree as described in Chapter III. This tree would have $N_v - 1$ branches and the chordset would consist of $N_a - N_v + 1$ edges. Inspection of the minimum allocation diagram in Figure 4.2 indicates that the critical tree consists of activities A_1, A_3,

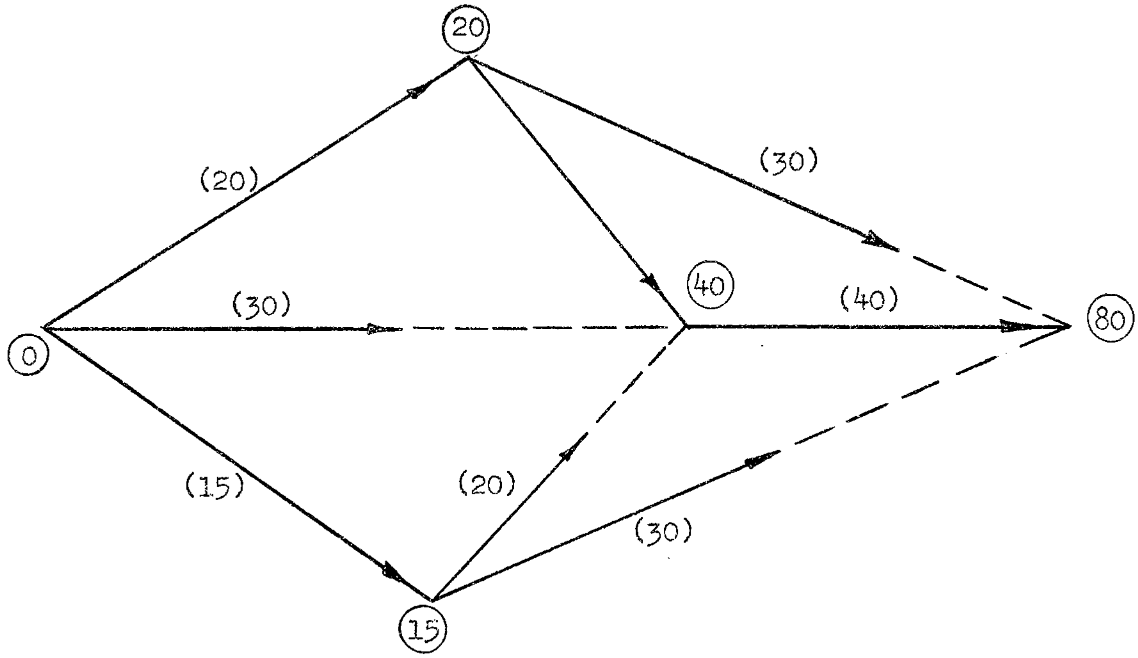

Figure 4.2. Minimum Allocation Diagram

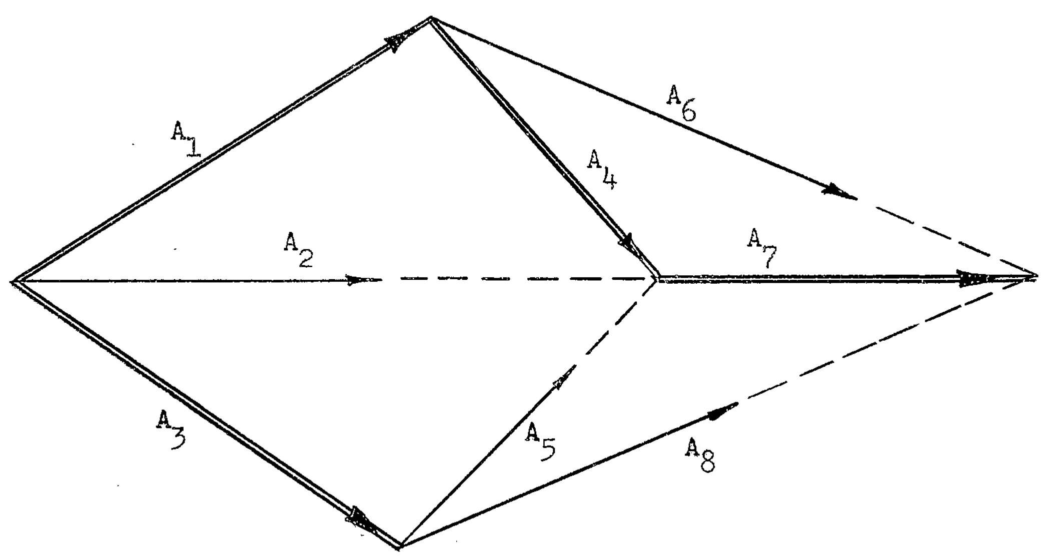

Figure 4.3. Minimum Critical Tree

A_4, and A_7. This is shown in Figure 4.3 (on the preceding page) by double lines. The chordset of this minimum critical tree consists of activities A_2, A_5, A_6, and A_8.

Step 4. Determine the critical path in the minimum allocation diagram. Designate the set of activities included in this path as C. Under any optimal allocation, according to the properties of the minimum allocation diagram described in Chapter III, the set C is sure to be one of the critical paths. Since the effectiveness of an allocation can be measured by the total elapsed time between 'project start' and 'project end', it can also be measured by the sum of the activity times along any critical path, expressed as the activities included in set C

$$E = \Sigma t^k_{ij} \ , \ t^k \varepsilon C. \qquad (4.3)$$

Equation (4.3) defines the effectiveness function to be optimized by the final mathematical programming procedure. In Figure 4.3, the critical path consists of activities A_1, A_4, and A_7, hence, set C consists of these three activities. The effectiveness function in Equation (4.3) can now be written as

$$E = T_1 + T_4 + T_7. \qquad (4.4)$$

And, from the definition of Θ, it follows that

$$E = \Theta_1 + T_1(\min) + \Theta_4 + T_4(\min) + \Theta_7 + T_7(\min). \qquad (4.5)$$

Since $T_1(\min)$, $T_4(\min)$, and $T_7(\min)$ are given constants in the optimization problem, E would be mathematically identical to optimizing another effectiveness function E′ expressed as

$$E' = \Theta_1 + \Theta_4 + \Theta_7. \qquad (4.6)$$

Step 5. Draw the augmented linear graph. This graph would be obtained by augmenting the basic linear graph through the introduction of dummy activities involving the following operations:

(1) Introduce one slack activity between the termination of each activity in the chordset of the minimum critical tree and the succeeding event. Since the chordset consists of $N_a - N_v + 1$ activities, this operation would introduce $N_a - N_v + 1$ slack activities and an equal number of events. The augmented linear graph for the illustrative example is shown in Figure 4.4. A_9, A_{10}, A_{11}, and A_{12} are the four slack activities introduced by this operation.

(2) Introduce one additional activity from the event 'project end' to the event 'project start'. The orientation of this activity would be in a direction opposite to the direction of the orientation of all real

activities. This operation introduces one dummy activity and no additional events to the basic linear graph. This dummy activity is activity A_{13} in the augmented linear graph of Figure 4.4. The resource allocation for this dummy activity would be the negative of the aggregate resource restriction limit. The role of this activity would be similar to that of an external current generator (driver) connected across the two extreme terminals of an electric system.

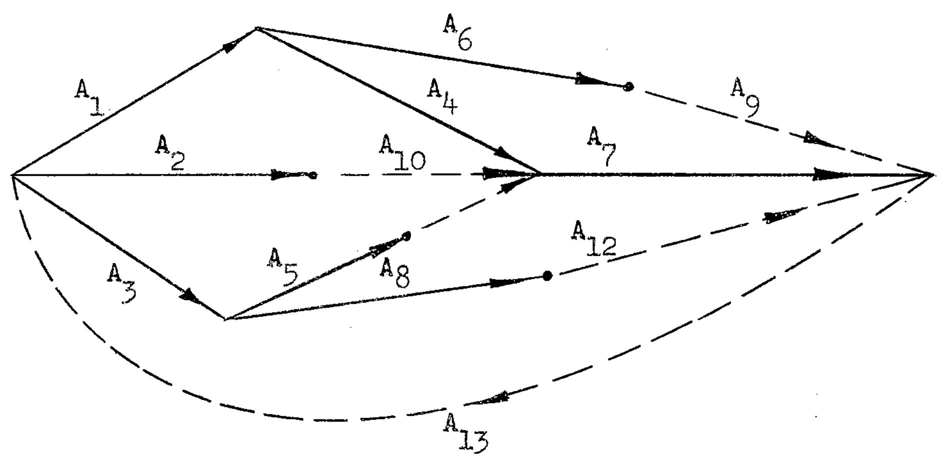

Figure 4.4. Augmented Linear Graph

The total number of activities in the augmented linear graph are then the sum of N_a activities in the basic linear graph, $N_a - N_v + 1$ slack activities introduced in Operation 1 and one dummy activity introduced in Operation 2, or $2N_2 - N_v + 2$ in all. The total number of events in the augmented linear graph are the sum of N_v events in the basic linear graph and $N_a - N_v + 1$ events introduced in Operation 1, or $N_a + 1$ in all.

Step 6. Determine the solution tree. The solution tree chosen would be the minimum critical tree in Step 3 augmented by real activity edges reacting from events on the minimum-critical tree to the additional events introduced in Step 5, Operation 1. The properties of this solution tree can be summarized as follows:

(1) This tree is complete, in that it connects all events in the augmented linear graph. This follows from the fact that the minimum critical tree is complete for the basic linear graph. All additional events created in Step 5, Operation 1 are connected in the formation of the solution tree.

(2) The branch set of this solution tree is identical with the set of real activities comprising the project.

(3) All activities for which the through variable, resource allocation is completely or partially specified, are included in the

chordset. This set consists of the slack activities introduced in Step 5, Operation 1, for which the resource allocation is partially specified and the dummy activity introduced in Step 5, Operation 2 for which the resource allocation is completely specified.

Since the augmented linear graph has $2N_a - N_v + 2$ activities (edges) and $N_a + 1$ events (nodes), it can be concluded (see Appendix) that the solution tree has N_a branches (number of nodes $N_a + 1$ less 1) and $N_a - N_v + 2$ chords (total edges $2N_a - N_v + 2$ less N_a branches). This solution tree is shown in Figure 4.5 by double lines. The branch set consists of activities A_1, A_2, A_3, A_4, A_5, A_6, A_7, and A_8. The chordset consists of activities A_9, A_{10}, A_{11}, A_{12} and A_{13}.

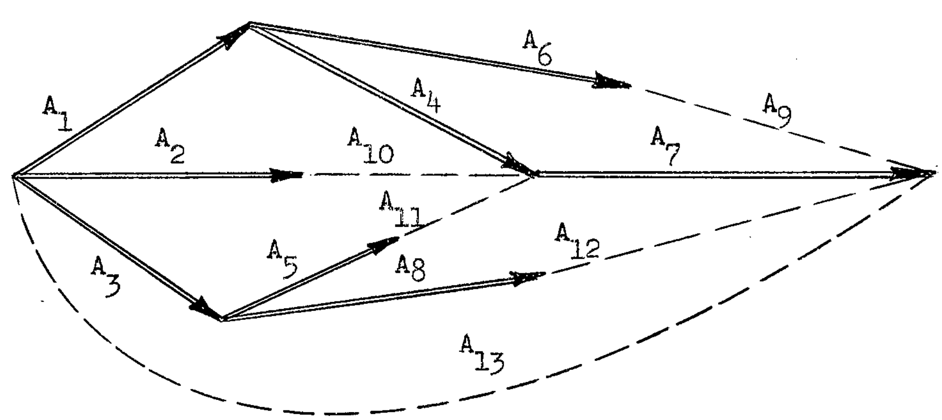

Figure 4.5. Solution Tree of the Augmented Linear Graph

The System Equations

Step 7. Write the component terminal equations. The only variables in the present system that are completely or partially specified are the resource allocations, or through variables. Under these circumstances, it would be desirable to express the system equations in the form of cutset equations. This choice would necessitate that the component terminal equations be explicit in the through variables. An individual component terminal equation explicit in the through variable would take the form

$$\begin{aligned} R_k &= \Phi(t_k) && \text{if} \quad t_k(\min) \le t_k \le t_k(\max) \\ &= R_k(\max) && \text{if} \quad t_k < t_k(\min) \\ &= R_k(\min) && \text{if} \quad t_k > t_k(\max). \end{aligned} \tag{4.7}$$

There are terminal equations of the same identical form for all real activities comprising the project. All N_a of these equations can be represented in matrix notation as

$$R = W \cdot T. \tag{4.8}$$

In Equation (4.8), R is a $(N_a \times 1)$ vector of resource allocations, T is a $(N_a \times 1)$ vector of activity times, and W is a $(N_a \times N_a)$ diagonal matrix whose diagonal elements are the functional elements $\Phi(\ )$ in the individual component terminal equations. Since the relation between activity times T and Θ is linear and one-to-one, the Equation (4.8) also can be written in terms of Θ instead of T as

$$R = W \cdot \Theta. \tag{4.9}$$

In the illustrative example, the assumption of linearity leads to the following component terminal equation from Table IV-I and IV-II:

$$R_1 = 30 - 2\Theta_1 . \tag{4.10}$$

$$R_2 = 40 - \frac{5\Theta_2}{6} \tag{4.11}$$

$$R_3 = 40 - 4\Theta_3 \tag{4.12}$$

$$R_4 = 40 - \frac{5\Theta_4}{2} \tag{4.13}$$

$$R_5 = 30 - \Theta_5 \tag{4.14}$$

$$R_6 = 40 - 3\Theta_6 \tag{4.15}$$

$$R_7 = 30 - \frac{\Theta_7}{2} \tag{4.16}$$

$$R_8 = 40 - \frac{5\Theta_8}{4} . \tag{4.17}$$

The vector R in Equation (4.9) would be an 8×1 vector, and W an 8×8 diagonal matrix. It should be noted that the elements of R and Θ are the same as the branch elements of the solution tree in the augmented linear graph.

Step 8. Write the cutset equations for the augmented linear graph. Since there are N_a branches to the solution tree, there would be N_a cutset equations. These could be represented in matrix form as

$$[I, A_1, A_2] \begin{bmatrix} Y_b \\ Y_{c1} \\ Y_{c2} \end{bmatrix} = 0. \qquad (4.18)$$

In Equation (4.18), Y_b is the vector of through variables for the branch elements of the solution tree. Hence, Y_b is identical to the vector R in Equations (4.8) and (4.9). The vector of through variables, Y_{c1} in the chord elements, is not completely specified. This $(N_a - N_v + 1) \times 1$ vector is a vector of resources remaining idle during the $N_a - N_v + 1$ slack activities. The vector of through variables, Y_{c2}, in the chord elements is completely specified. The only element for which the resource allocation is completely specified in the dummy activity introduced in Step 5, Operation 2. Hence, Y_{c2} is a 1×1 vector.

Since there are N_a branch elements to the solution tree, the matrix I in Equation (4.18) is a $N_a \times N_a$ identity matrix. As the $N_a \times (N_a - N_v + 1)$ matrix is multiplicative with the $(N_a - N_v + 1) \times 1$ vector. Y_{c1} and A_2 is a $N_a \times 1$ column vector multiplicative with the 1×1 vector, Y_{c2}. The cutset equations also can be written in the expanded form

$$[I, A_1] \begin{bmatrix} Y_b \\ Y_{c1} \end{bmatrix} + A_2 \cdot Y_{c2} = 0. \qquad (4.19)$$

Since there are eight branches to the solution tree in the example, there would be eight cutset equations. These are shown in matrix form in Figure 4.6.

$$\underbrace{\begin{bmatrix} 1&0&0&0&0&0&0&0&0&1&1&1\\ 0&1&0&0&0&0&0&0&0&-1&0&0\\ 0&0&1&0&0&0&0&0&0&0&-1&-1\\ 0&0&0&1&0&0&0&0&1&1&1&1\\ 0&0&0&0&1&0&0&0&0&0&-1&0\\ 0&0&0&0&0&1&0&0&-1&0&0&0\\ 0&0&0&0&0&0&1&0&1&0&0&1\\ 0&0&0&0&0&0&0&1&0&0&0&-1 \end{bmatrix}}_{I \qquad\qquad A_1} \times \begin{bmatrix} R_1\\ R_2\\ R_3\\ R_4\\ R_5\\ R_6\\ R_7\\ R_8\\ R_9\\ R_{10}\\ R_{11}\\ R_{12} \end{bmatrix} + \begin{bmatrix} -1\\ 0\\ 0\\ 1\\ 0\\ 0\\ -1\\ 0 \end{bmatrix} \times R_{13} = \begin{bmatrix} 0\\ 0\\ 0\\ 0\\ 0\\ 0\\ 0\\ 0 \end{bmatrix}$$

Figure 4.6. Expanded Form of Equation (4.19)

Step 9. The resource allocations in the individual slack activities are uniquely determined by the resource allocations of the corresponding real activities preceding them. This can be expressed in the form of a set of $N_a - N_v + 1$ equations of the form

$$R_k = R_i \quad , \quad k\varepsilon \text{ chordset and } i\varepsilon \text{ branchset.} \qquad (4.20)$$

Collectively, these equations can be expressed in the matrix form

$$R_s = Z \cdot R. \qquad (4.21)$$

Here, R_s is a $(N_a - N_v + 1) \times 1$ vector of idle resource allocations for the $N_a - N_v + 1$ slack activities, R is a $N_a \times 1$ vector of resource allocations for the real activities (branch elements), and Z is a $(N_a - N_v + 1) \times (N_a)$ relationship matrix. The vector R_s is identical with vector Y_{c1} in Equation (4.19). Hence, in the light of Equation (4.21), the vector $\begin{bmatrix} Y_b \\ Y_{c1} \end{bmatrix}$ in Equation (4.19) can be written in the following modified form:

$$\begin{bmatrix} Y_b \\ Y_{c1} \end{bmatrix} = \begin{bmatrix} R \\ R_s \end{bmatrix} = \begin{bmatrix} R \\ Z \cdot R \end{bmatrix} = \begin{bmatrix} I \\ Z \end{bmatrix} \cdot R. \qquad (4.22)$$

The relations governing the resource allocations of the slack activities in the example are

$$\begin{aligned} R_{10} &= R_2 & R_{12} &= R_8 \\ R_9 &= R_6 & R_{11} &= R_5 . \end{aligned} \qquad (4.23)$$

Collectively, these can be expressed in matrix form as

$$\begin{bmatrix} R_9 \\ R_{10} \\ R_{11} \\ R_{12} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} R_1 \\ R_2 \\ R_3 \\ R_4 \\ R_5 \\ R_6 \\ R_7 \\ R_8 \end{bmatrix} \qquad (4.24)$$

Matrix Z in Equation (4.24) leads to the matrix $\begin{bmatrix} I \\ Z \end{bmatrix}$ in Equation (4.22) as shown in Figure 4.7.

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Figure 4.7. The Matrix $\begin{bmatrix} I \\ Z \end{bmatrix}$

Step 10. Rewrite the cutset equations by substituting Equation (4.22) into Equation (4.19) to obtain

$$[I, A_1] \begin{bmatrix} Y_b \\ Y_{c1} \end{bmatrix} + A_2 \cdot Y_{c2}$$

$$= [I, A_1] \cdot \begin{bmatrix} I \\ Z \end{bmatrix} \cdot R + A_2 \cdot Y_{c2} = 0. \qquad (4.25)$$

Equation (4.25) for the illustrative example is shown in Figure 4.8.

Step 11. At this stage, many of the equations are redundant. In fact, each element which is a chord in the minimum critical tree and is a branch in the solution tree gives rise to one redundant cutset equation. Each of these activities precedes one slack activity as shown in Figure 4.9. The equation of the cutset shown by the dotted circle, which isolates node E_2 from the rest of the linear graph, can be stated as

$$R_{(12)} - R_{(23)} = 0. \qquad (4.26)$$

But, $R_{(12)}$ is equal to $R_{(23)}$ by definition (similar to Equation (4.20). Hence, the cutset Equation (4.26) reduces to $R_{(12)} - R_{(12)} = 0$ or $0 = 0$, which is redundant. The minimum critical tree has $N_a - N_v + 1$ chords, all of which are branches in the solution tree. Hence, there would be $N_a - N_v + 1$ redundant cutset equations. After deleting these redundant equations, there are $N_v - 1$ non-redundant cutset

$$
\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & -1
\end{bmatrix}
\times
\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}
\begin{bmatrix} R_1 \\ R_2 \\ R_3 \\ R_4 \\ R_5 \\ R_6 \\ R_7 \\ R_8 \end{bmatrix}
+
\begin{bmatrix} -1 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ -1 \\ 0 \end{bmatrix}
\times R_{13} =
\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}
$$

Figure 4.8. Expanded Form of Equation (4.25)

equations. Thus, the total number of equations less the redundant equations equals a reduced set of non-redundant equations expressed as

$$N_a - (N_a - N_v + 1) = N_v + 1. \tag{4.27}$$

This reduced set of equations can be expressed in matrix form as

$$J \cdot R + A_2^* \cdot Y_{c2} = 0. \tag{4.28}$$

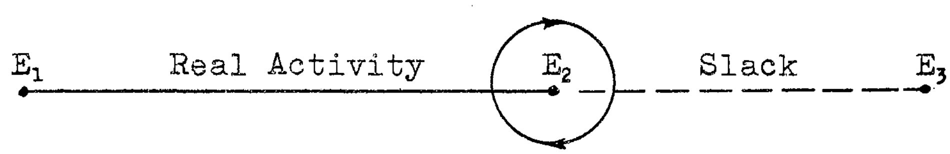

Figure 4.9. Cutset for a Slack Activity

In Equation (4.27) J is a $(N_v - 1) \times N_a$ matrix obtained by reducing the product matrix $[I, A_1] \cdot \begin{bmatrix} I \\ Z \end{bmatrix}$ and A_2^* is a $(N_v - 1) \times 1$ vector obtained by reducing the vector A_2. The product matrix $[I, A_1] \cdot \begin{bmatrix} I \\ Z \end{bmatrix}$ is exhibited in Figure 4.10. It can be noted that the elements in rows 2, 5, 6, and 8 consist entirely of zeros. These four rows give rise to four redundant equations. After deleting these, there are only four non-redundant cutset equations. This number is equal to $5(N_v) - 1$ and satisfies Equation (4.27). Figure 4.10

also exhibits A_2*, the reduced form of A_2.

$$\begin{bmatrix} 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & -1 & 0 & 0 & -1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Figure 4.10. Matrix Product $[I, A_1] \cdot \begin{bmatrix} I \\ Z \end{bmatrix}$

The Mathematical Programming Format

Step 12. Substituting the value of R from Equation (4.9) into the Equation (4.28), the reduced set of cutset equations becomes

$$J \cdot R + A_2{*} \cdot Y_{C2} = J \cdot W \cdot \Theta + A_2{*} \cdot Y_{C2} = 0. \qquad (4.29)$$

Equation (4.29) represents the first set of restrictions which must be satisfied by any feasible resource allocation set. There are $N_V - 1$ equations in this set. From Figure 4.10, the reduced matrix J can be written as

$$\begin{matrix} 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & -1 & 0 & 0 & -1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \end{matrix} .$$

Hence, the set of equations $J \cdot R + A_2^* \cdot Y_{C2} = 0$ becomes

$$\begin{aligned} &(a) \quad R_1 + R_2 + R_5 + R_8 - R_{13} = 0 \\ &(b) \quad R_3 - R_5 - R_8 = 0 \\ &(c) \quad R_2 + R_4 + R_5 + R_6 + R_8 - R_{13} = 0 \\ &(d) \quad R_6 + R_7 + R_8 - R_{13} = 0 \ . \end{aligned} \tag{4.30}$$

Substituting the component terminal Equations (4.10) through (4.17) in Equations (4.30) and putting R_{13} equal to 80 units gives

$$\begin{aligned} &(a) \quad (30 - 2\Theta_1) + (40 - \frac{5\Theta_2}{6}) + 30 - \Theta_5) + (40 - \frac{5\Theta_8}{4}) \\ &\qquad\qquad - 80 = 0 \\ &(b) \quad (40 - 4\Theta_3) - (30 - \Theta_5) - (40 - \frac{5\Theta_8}{4}) = 0 \\ &(c) \quad (40 - \frac{5\Theta_2}{6}) + (40 - \frac{5\Theta_4}{2}) + (30 - \Theta_5) + (40 - \frac{5\Theta_8}{4}) \\ &\qquad\qquad - 80 = 0 \\ &(d) \quad (40 - 3\Theta_6) + (30 - \frac{\Theta_7}{2}) + (40 - \frac{5\Theta_8}{4}) - 80 = 0 \ . \end{aligned} \tag{4.31}$$

Simplifying Equation (4.31), the first set of restrictions is

$$\text{(a)}\quad 60 - 2\Theta_1 - \frac{5\Theta_2}{6} - \Theta_5 - \frac{5\Theta_8}{4} = 0$$

$$\text{(b)}\quad -30 - 4\Theta_3 + \Theta_5 + \frac{5\Theta_8}{4} = 0$$

$$\text{(c)}\quad 110 - \frac{5\Theta_2}{6} - \frac{5\Theta_4}{2} - \Theta_5 - 3\Theta_6 - \frac{5\Theta_8}{4} = 0$$

$$\text{(d)}\quad 30 - 3\Theta_6 - \frac{\Theta_7}{2} - \frac{5\Theta_8}{4} = 0 \; . \qquad (4.32)$$

<u>Step 13</u>. Each activity has a maximum as well as minimum competion time. This means that there are N_a inequalities of the type

$$\Theta_k \leq \Theta_k(\max). \qquad (4.33)$$

If the allocation of resources is optimum, then the branches of the minimum critical tree cannot have a positive slack and an activity can have a slack only if the time between the start and the end of that activity exceeds t(max) for that activity. Thus, if the allocation is optimum, the Inequalities (4.33) are always satisfied for the branches of of the minimum critical tree. Thus, these $N_v - 1$ inequalities present no restrictions on the optimizing process. The remaining $N_a - N_v + 1$ Inequalities (4.33) constitute the second set of restrictions that must be satisfied by any feasible optimum resource allocation.

<u>Step 14</u>. The sets of restrictions described in Step 12 and Step 13, together with the effectiveness function of Equation (4.6), now constitute the complete statement of the problem. This can be summarized as

Optimize $E = \Sigma \Theta_k$, $k \varepsilon C$

Subject to the restrictions

$$J \circ W \circ \Theta + A_2{*} \circ Y_{C2} = 0$$

$$\Theta_k \leq \Theta_k(\max) \text{ for all } k\varepsilon \text{ chordset of minimum critical tree.}$$

This completes the formulation of the resource allocation problem in the generalized mathematical programming format. Since a minimum duration is sought for the project, the effectiveness function would be minimized. The first set consists of $N_v - 1$ restrictions and the second set consists of $N_a - N_v + 1$ restrictions. In all there are $N_v - 1 + N_a - N_v + 1$ or N_a restrictions and N_a variables of the type Θ_k. This insures that a solution always exists, although the actual solution may be difficult to find.

The objective function and the second set of restrictions are linear, but the first set of restrictions would be dependent upon the functional relationships between the activity times and the corresponding resource allocations. If these are assumed to be linear, a solution by the simplex method would be feasible. In most cases, the actual relationships would be non-linear and would render the entire problem non-linear. In that case, the solution would require a complex non-linear programming algorithm.

CHAPTER V

SOLUTION BY THE SIMPLEX ALGORITHM

In the present problem, the objective is the minimization of the total elapsed time between the events 'project start' and 'project end'. The variables of the system are the individual activity times and the corresponding resource allocations. As explicit or implied functional relationships exist between all the variables, it is evident that a mathematical programming technique can be successfully applied in the optimization of the effectiveness function. In the case of linear programming problems, the most effective and general technique has been the simplex method. This chapter will use the simplex algorithm to obtain the optimum solution for the illustrative problem of Chapter IV.

Problem Summary

Numerically, the problem of Chapter IV can now be summarized as requiring the minimization of

$$E' = \Theta_1 + \Theta_4 + \Theta_7 . \qquad (5.1)$$

Subject to the restrictions

$$(1) \quad 60 - 2\Theta_1 - \frac{5\Theta_2}{6} - \Theta_5 - \frac{5\Theta_8}{4} = 0$$

(2) $-30 - 4\Theta_3 + \Theta_5 + \frac{5\Theta_8}{4} = 0$

(3) $110 - \frac{5\Theta_2}{6} - \frac{5\Theta_4}{2} - \Theta_5 - 3\Theta_6 - \frac{5\Theta_8}{4} = 0$

(4) $30 - 3\Theta_6 - \frac{\Theta_7}{2} - \frac{5\Theta_8}{2} = 0$

(5) $\Theta_2 \leq 24$

(6) $\Theta_5 \leq 10$

(7) $\Theta_6 \leq 10$

(8) $\Theta_8 \leq 20$.

Out of the eight restrictions, number 1, 2, 3, and 4 are equalities and 5, 6, 7, and 8 are inequalities. Whenever any of the restrictions are in the form of equalities, a choice between two alternatives is available. The equality may be modified into another equality by introducing one artificial and one slack variable, or the equality may be used to define one of the unknown variables and, thus, reduce both the number of restrictions and the number of variables by one each. Mathematically, the two alternatives are identical but the second is chosen with the belief that it will result in less computation. Thus, addition of restrictions 1 and 2 results in the following:

$$60 - 2\Theta_1 - \frac{5\Theta_2}{6} - \Theta_5 - \frac{5\Theta_8}{4} = 0$$

$$\underline{-30 - 4\Theta_3 + \Theta_5 + \frac{5\Theta_8}{4} = 0}$$

$$30 - 2\Theta_1 - 4\Theta_3 - \frac{5\Theta_2}{6} = 0$$

$$\text{or } \Theta_2 = 36 - 2.4\Theta_1 - 4.8\Theta_3 . \qquad (5.3)$$

Substitution of Equation (5.3) into restriction 5 yields

$$36 - 2.4\Theta_1 - 4.8\Theta_3 \leq 24$$

$$\text{or } \Theta_1 + 2\Theta_3 \geq 5. \qquad (5.4)$$

Equation (5.4) is the modified restriction 5.

Similarly, the subtraction of restriction 4 from restriction 3 along with the substitution for Θ_2 from Equation (5.3) yields

$$110 - \left\{30 - 2\Theta_1 - 4\Theta_3\right\} - \frac{5\Theta_4}{2} - \Theta_5$$

$$-3\Theta_6 - \frac{5\Theta_8}{4} - \left\{30 - 3\Theta_6 - \frac{\Theta_7}{2} - \frac{5\Theta_8}{4}\right\} = 0$$

$$\text{or } 50 - 2\Theta_1 + 4\Theta_3 - \frac{5\Theta_4}{2} + \frac{\Theta_7}{2} = \Theta_5 . \qquad (5.5)$$

Substitution of Equation (5.5) into restriction 6 yields

$$50 + 2\Theta_1 + 4\Theta_3 - \frac{5\Theta_4}{2} + \frac{\Theta_7}{2} \leq 10$$

$$\text{or } \frac{5\Theta_4}{2} - \frac{\Theta_7}{2} - 2\Theta_1 - 4\Theta_3 \geq 40 . \qquad (5.6)$$

Equation (5.6) is the modified restriction 6.

Substitution of the value of Θ_5 from Equation (5.5) into

$$-30 - 4\Theta_3 + 50 + 2\Theta_1 + 4\Theta_3 - \frac{5\Theta_4}{2} + \frac{\Theta_7}{2} + \frac{5\Theta_8}{4} = 0$$

$$\text{or } \Theta_8 = 2\Theta_4 - 1.6\Theta_1 - 0.4\Theta_7 - 16 . \qquad (5.7)$$

Substitution of Equation (5.7) into restriction 8 modifies it to the form

$$1.6\Theta_1 + 0.4\Theta_7 - 2\Theta_4 \geq -36. \quad (5.8)$$

Substitution of Equation (5.8) into restriction 4 yields

$$30 - 3\Theta_6 - 0.5\Theta_7 - \{-20 - 2\Theta_1 + 2.5\Theta_4 - 0.5\Theta_7\} = 0$$

$$\text{or } 3\Theta_6 = 50 + 2\Theta_1 - 2.5\Theta_4 . \quad (5.9)$$

Substitution of Equation (5.9) into restriction 7 modifies it to the form

$$2.5\Theta_4 - 2\Theta_1 \geq 20 . \quad (5.10)$$

Hence, the problem in its reduced form can now be stated as minimize

$$E' = \Theta_1 + \Theta_4 + \Theta_7 .$$

Subject to the restrictions

$$\Theta_1 + 2\Theta_3 \geq 5$$

$$2.5\Theta_4 - 0.5\Theta_7 - 2\Theta_1 - 4\Theta_3 \geq 40$$

$$1.6\Theta_1 + 0.4\Theta_7 - 2\Theta_4 \geq -36$$

$$2.5\Theta_4 - 2\Theta_1 \geq 20 .$$

The non-negativity constraints usually implied in a linear programming problem are not introduced here because of the nature of the resource-time function. A negative value for any variable Θ would not be invalid per se. A negative Θ implies that an attempt is made to expedite the activity beyond t(min) through the allocation of resources over and above the maximum. In physical terms, this means that the activity is allocated maximum resources and the surplus resources remain idle for the duration of the activity. However, the non-negativity constraints must be implied for the dummy variables introduced by the simplex procedure as these variables are not known to obey any function similar to the resource-time function that would validate their negative values.

Solution for the Optimal Program

Solution for the optimum allocation program by the simplex method is exhibited in Figures 5.1, 5.2, 5.3., and 5.4. The optimum program is $\Theta_1 = 2.5$, $\Theta_3 = 1.25$, $\Theta_4 = 10$, and $\Theta_7 = -50$. The negative value for Θ_7 indicates that there is an idle resource allocation associated with activity A_7. The optimal program translated into the numerical values of resource allocations and activity duration is exhibited in Figure 5.7 and the optimum allocation diagram is shown in Figure 5.8. From Figure 5.8 it is seen that the project would be executed in 52.5 units of time. This would be the shortest duration feasible under the resource availability specified.

		1	0	1	1	M	M	M	M	0	0	0	0		
C	Pro	θ_1	θ_3	θ_4	θ_7	S_1	S_2	S_3	S_4	A_1	A_2	A_3	A_4	Const	
M	S_1	1	2			1				-1				5	
M	S_2	-2	-4	2.5	-0.5		1				-1			40	16
0	A_3	1.6		-2	0.4			1				-1		-36	18
M	S_4	-2		2.5					1				-1	20	8 ✓
		-3M	-2M	3M ✓	-0.5M	—	—	M	—	-M	-M	—	-M	85M	

Figure 5.1. Simplex Solution Iteration 1

		1	0	1	1	M	M	M	M	0	0	0	0			
C	Pro	θ_1	θ_3	θ_4	θ_7	S_1	S_2	S_3	S_4	A_1	A_2	A_3	A_4	Const		
M	S_1	1	2			1				-1				5	5	✓
M	S_2		-4		-0.5		1		-1		-1		+1	20		
0	A_3				0.4			1	0.8			-1	-0.8	-20		
1	θ_4	-0.8		1					0.4				-0.4	8	-10	
		M-0.8	-2M	—	-0.5M +0.4	—	—	0	-M+0.4	-M	-M	—	-M-0.4	25M+8		
		✓														

Figure 5.2. Simplex Solution Iteration 2

		1	0	1	1	M	M	M	M	0	0	0	0			
C	Pro	θ_1	θ_3	θ_4	θ_7	S_1	S_2	S_3	S_4	A_1	A_2	A_3	A_4	Const		
1	θ_1	1	2			1				-1				5		
M	S_2		-4		-0.5		1		-1		-1		+1	20	20	✓
0	A_3				0.4			1	0.8			-1	-0.8	-20	25	
1	θ_4		1.6	1		0.8			0.4	-0.8			-0.4	12	-30	
		—	-4M +3.6	—	-0.5M	1.8	—	1	-M+0.4	-1.8	-M	—	M-0.4	20M +17		
													✓			

Figure 5.3. Simplex Solution Iteration 3

		1	0	1	1	M	M	M	M	0	0	0	0		
C	Pro	θ_1	θ_3	θ_4	θ_7	S_1	S_2	S_3	S_4	A_1	A_2	A_3	A_4	Const.	
1	θ_1	1	2			1				-1				5	2.5
0	A_4		-4		-0.5		1		-1		-1		1	20	-5
0	A_3		-3.2				0.8	1	0.0		-0.8	-1		-4	1.25 ✓
1	θ_4			1	-0.2	0.8	0.4			-0.8	-0.4			20	
		—	2	—	-0.2	1.8	0.4	0	0	-1.8	-0.4	—	—	25	
			✓												

Figure 5.4. Simplex Solution Iteration 4

		1	0	1	1	M	M	M	M	0	0	0	0		
C	Pro	θ_1	θ_3	θ_4	θ_7	S_1	S_2	S_3	S_4	A_1	A_2	A_3	A_4	Const.	
1	θ_1	1				1	0.5	0.625		-1	-0.5	-0.625		2.5	
0	A_4				-0.5			-1.25	-1			1.25	1	25	-50
0	θ_3		1				-0.25	-0.3125			0.25	0.3125		1.25	
1	θ_4			1	-0.2	0.8	0.4			-0.8	-0.4			20	-100
		—	—	—	-0.2	1.8	0.9	0.625		-1.8	-0.9	-6.25	—	22.5	

Figure 5.5. Simplex Solution Iteration 5

		1	0	1	1	M	M	M	M	0	0	0	0		
C	Pro	θ_1	θ_3	θ_4	θ_7	S_1	S_2	S_3	S_4	A_1	A_2	A_3	A_4	Const	
1	θ_1	1				1	0.5	0.625		-1	-0.5	-0.625		2.5	
1	θ_7				1			2.5	+2			-2.5	-2	-50	
0	θ_3		1				-0.25	-0.3125			0.25	0.3125		1.25	
1	θ_4			1		0.8	0.4	0.5	0.4	-0.8	-0.4	-0.5	-0.5	10	
		—	—	—	—	1.8	0.9	3.625	2.4	-1.8	-0.9	-3.625	-2.4		

Figure 5.6. Simplex Solution Iteration 6 (Final)

Activity	Θ	Time	Resource Allocation
A_1	2.5	12.5	25
A_2	24	30	20
A_3	1.25	11.25	35
A_4	10	20	15
A_5	10	20	20
A_6	10	30	10
A_7	-50	20	30 + 25 idle
A_8	20	30	15

Figure 5.7. Optimum Allocation Program

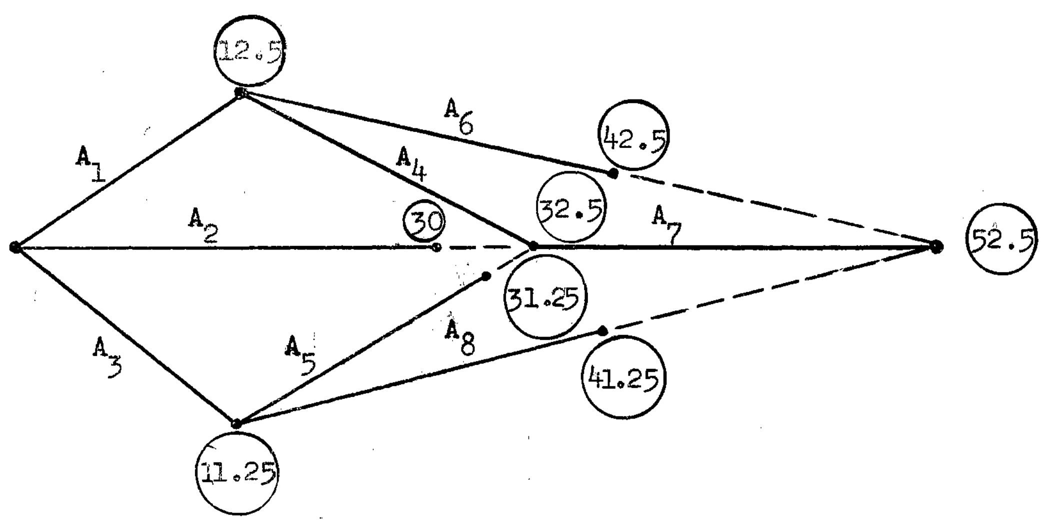

Figure 5.8. Optimum Allocation Diagram

CHAPTER VI

SUMMARY AND CONCLUSIONS

This concluding chapter will be composed of three sections. In the first section, the principal theme of this investigation will be summarized. The second section will consist of some observations and remarks regarding the proposed resource allocation model as it compares with the present critical path method (CPM) model. In the final section, some suggestions for future study are presented.

Summary

This investigation was motivated by a striking resemblance between activity networks and electrical networks. In order that such a comparison be valid, it is necessary that the variables of analysis of the two systems be similar. In electrical networks, the through variable, current, and the across variable, e.m.f., for any component, are functionally related. This functional relationship can be expressed symbolically as $i = \Phi(v)$. In a similar manner, the through variable, resource allocation, and the across variable, time, for any activity in an activity network are functionally related. This functional relationship can be expressed symbolically as $R = \Phi(t)$. These similarities led to the assumption that an activity in an activity network behaves exactly like a two-terminal electrical component in an electrical network. Based on this assumption, it was found that the principles of system theory can be applied to the analysis of an activity network system.

In Chapter I the problem of activity network analysis was described in historical and current contexts. Along with the historical background, a brief outline of one of the commonly accepted models was described. The concepts of slack and critical path developed in this model were later shown to remain unaltered under optimization of resource allocation. The two major limitations of this model which initiated the present investigation were also described. These limitations consisted of the following:

(1) The restriction imposed by the limited availability of resources was not made an integral part of the model.

(2) The concept of managerial control over the planning of activity networks was lacking.

It was believed that both of these limitations could be overcome if it is assumed that the duration of an activity can be controlled by varying the resources allocated to that activity. This assumption provided the basis of the investigation.

In Chapter II some intuitive principles of allocating scarce resources between competing activities were analyzed. A symbolic statement of some simple networks indicated that these intuitive principles were applicable. It was also shown that the application of such principles does not alter the general format of the activity network model, including the slacks of activities and the critical path.

In Chapter III the similarity between a two-terminal system component and activity in a project was critically examined. This examination proved that the description of an activity as a two-terminal component does satisfy all requisite conditions specified by system theory. The minimum allocation diagram was also described in this chapter. The importance of this diagram lay in the fact that the critical tree in this diagram is the best choice for the solution tree in subsequent analysis. The critical path in this diagram was shown to remain critical under optimum allocation. This fact yielded the effectiveness function for the optimization procedure of Chapters IV and V.

Chapter IV describes the adaption of linear graph theory for the formulation of an algorithm for the optimum allocation of scarce resources. This algorithm is described as a step-by-step procedure and lead to the conversion of a resource allocation problem into a mathematical programming problem. The programming problem was shown to be non-linear in general. However, it could be reduced to a linear problem by some simplifying assumption regarding the resource-time function and by disregarding the non-negativity constraints in the linear programming techniques.

A numerical solution of a simplified linear programming problem by the simplex procedure was given in Chapter V. This solution illustrated numerically the procedure of optimum resource allocation for a small activity network. The development of this simple illustrative resource allocation problem into the mathematical programming problem was also described step-by-step.

Observations and Remarks

The model of an activity network presented in this dissertation differs from the more commonly used models in two important aspects. First, in the proposed model, the resources are considered as a flow instead of a cost, or in the units of dollars per year rather than dollars. There is a growing recognition of the fact that the total expenditure of resources is not nearly as critical as the rate of expenditure in project management. This is partly because the limitations of the resource availability are based on the availability within a specified time period. An example of a major project changing over from the concept of total resource expenditure to the rate of flow concept is the Appollo Manned Lunar Project of the National Aeronautics and Space Administration (15).

The second major diversion of the proposed model lies in the introduction of the concept of the control of network planning through the modification of the resource allocations. Traditionally, activity network techniques have been review techniques for pre-established plans. It is proposed in this dissertation that the planning function itself can be made a part of the model through the resource allocation procedures. This in no way reduces the usefulness of the model as a review technique after the planning phase. Thus, the conceptual scope of the model is broadened to include the planning function without altering the basic format of the model.

Areas for Further Study

The algorithm presented in this dissertation is limited as to its immediate application in project management. This is because of two implicit assumptions:

(1) The resource availability is uniform over the duration of the project.

(2) The resource allocation - time duration function is completely known for each activity.

The expanded model of an activity network as proposed is complete in its conceptual form. Considerable research and refinement will be necessary to make the concept useful in practice. Further study on the following areas would prove useful:

(1) Investigation of the nature of the resource allocation activity time duration function

and methods for estimating and approximating the same.

(2) The functioning of the model under non-uniform availability of resources.

(3) The application of Bayesian strategies for planning when the resource-time function is unknown. This area may prove to be most fruitful because it is unlikely that the resource-time function would ever be exactly known.

BIBLIOGRAPHY

(1) Malcolm, D. G., J. H. Rosebom, C. E. Clark, and W. Fazzar. "Application of a Technique for Research and Development," Operations Research (1959), pp. 646-669.

(2) Kelly, J. E., Jr., and M. R. Walker. "Critical Path Planning and Scheduling," Proceedings Eastern Joint Computer Conference (1959), pp. 160-173.

(3) Charnes, A., and W. W. Cooper. "A Network Interpretation and a Directed Sub-Dual Algorithm for Critical Path Scheduling," Journal of Industrial Engineering, Vol. 13 (1962), pp. 213-219.

(4) Weist, J. D. "Schedules for Large Projects With Limited Resources," Operations Research, Vol. 3 (1964), pp. 395-418.

(5) Kelly, J. E., Jr. "Schedules for Large Projects With Limited Resources," Industrial Scheduling (Chapter 21). Editors. J. F. Muth and G. L. Thompson. Englewood Cliffs, New Jersey: Prentice-Hall, 1963.

(6) Muth, J. F., and G. L. Thompson. Industrial Scheduling. Englewood Cliffs, New Jersey: Prentice-Hall, 1963.

(7) Moder, Joseph J., and Cecil R. Phillips. Project Management With CPM and PERT. New York: Reinhold Publishing Company, 1964.

(8) Levy, F. K., G. L. Thompson, and J. D. Weist. "Mathematical Basis of Critical Path Method," Industrial Scheduling (Chapter 22). Editors. J. F. Muth and G. L. Thompson. Englewood Cliffs, New Jersey: Prentice-Hall, 1963.

(9) Churchman, C. W., R. L. Ackoff, and E. L. Arnoff. Introduction to Operations Research. New York: John Wiley and Sons, Inc., 1962.

(10) Barnard, C. I. The Functions of the Executive. Cambridge, Massachusetts: Harvard University Press, 1960.

(11) Koenig, H. E., and W. A. Blackwell. Electro-Mechanical System Theory. New York: McGraw-Hill Book Company, 1961.

(12) Ford, L. R., Jr., and D. R. Fulkerson. Flows in Networks. Princeton, New Jersey: Princeton University Press, 1962.

(13) Ore, O. Theory of Graphs. Providence, Rhode Island: American Mathematical Society Colloquium Publications, 1962.

(14) Berge, C. The Theory of Graphs. London, England: Meuthen and Company, 1962.

(15) Markley, J. T. "Apollo Moon Project," A paper presented to the Winter Conference of the Southwestern Institute of Management Science at Houston, Texas, January 22, 1965.

APPENDIX

FOREWORD TO THE APPENDIX

This Appendix describes the definitions, postulates, and theorems of the linear graph theory which have been referred to in this dissertation. The material presented was compiled from class notes taken while attending a course in system theory taught by Dr. Richard L. Cummins at Oklahoma State University during the Spring of 1964. Additional principles of linear graph theory may be found in references (11), (12), (13), and (14).

Principles of Linear Graph Theory

Definition 1. Oriented element. An oriented line segment together with its distinct end points.

Definition 2. Vertex. An end point of an element.

Definition 3. Oriented linear graph. A collection of oriented elements no two of which have a point in common which is not a vertex.

Definition 4. Subgraph. A subgraph S of a graph G is a subset of the elements of G.

Definition 5. Complement of a subgraph. The complement of a subgraph S of a graph G is the subgraph remaining in G when the elements of S are removed.

Definition 6. Path. A path between two distinct vertices (called terminal vertices) is a subgraph such that its n elements can be sequentially labeled $e_1, e_2, \ldots, e_n$ with corresponding vertex labeling $v_1^1 - v_2^1, \ldots, v_1^2 - v_2^2, \ldots, v_1^n - v_2^n$ where each of the non-terminal verticles has exactly two labels v_2^k and v_1^{k+1}, $k = 1, 2, \ldots, n-1$. Each of the terminal vertices has exactly one label.

Definition 7. Connected graph. A graph is connected if and only if at least one path exists between every pair of distinct vertices of the graph.

Definition 8. Circuit. A circuit C is a subgraph of a connected graph such that there are exactly two distinct paths between any two vertices of C.

Definition 9. Tree. A tree T of a graph G is a connected subgraph which contains all the vertices of G and no circuits.

Theorem 1. If G is a connected graph and T is a tree of G, then there is exactly one path in T between any two vertices of G.

Theorem 2. If a connected graph G contains N_b elements and N_v vertices, any tree T of G contains $N_v - 1$ branches.

Definition 10. Chord Set. The complement of a tree T is said to be the chord set of T. Each element of the chord set is called a chord of T.

Theorem 3. If G is a connected graph and T is a tree of G, then each chord of T together with T defines a unique circuit.

Theorem 4. The number of chords in a connected graph G for any tree is $N_b - N_v + 1$.

Theorem 5. If G is a connected graph and S is some subset of elements which contains no circuits, then S can be made a part of a tree of G.

Definition 11. Cut set. If the vertices of a graph G are segregated into two disjoint sets, S_1 and S_{29}, the set of elements incident to one vertex in S_1 and one vertex in S_2 is designated a cut set.

Theorem 6. Each set of elements incident to one vertex is a cut set.

Definition 12. Cutset matrix, $\underline{Q}_a$. For each possible segregation of vertices of a connected graph into sets S_1 and S_2, the typical entry of $\underline{Q}_a$ is

$$q_{ij} = \begin{cases} +1 \text{ if } j^{th} \text{ element is in } i^{th} \\ \quad \text{cutset and oriented } S_1 \to S_2 \\ -1 \text{ if the } j^{th} \text{ element is in } i^{th} \\ \quad \text{cutset and oriented } S_2 \to S_1 \\ 0 \text{ if } j^{th} \text{ element is not in } i^{th} \\ \quad \text{cutset.} \end{cases}$$

Definition 13. Incidence (Node) Matrix. The matrix $\underline{A}_a$ of a connected graph is the matrix for which the typical entry is

$$a_{ij} = \begin{cases} +1 \text{ if the } j^{th} \text{ element is incident} \\ \quad \text{to the } i^{th} \text{ node and oriented} \\ \quad \text{away from it} \\ -1 \text{ if the } j^{th} \text{ element is incident} \\ \quad \text{to the } i^{th} \text{ node and oriented} \\ \quad \text{toward it} \\ 0 \text{ if the } j^{th} \text{ element is not in-} \\ \quad \text{cident to the } i^{th} \text{ node for} \\ \quad \text{each node of the graph.} \end{cases}$$

Theorem 7. The rows of $\underline{A}_a$ are included in the rows of $\underline{Q}_a$ if each vertex segregated set is always taken as a S_1 set.

Theorem 8. The rows of $\underline{Q}_a$ are linear combinations of the rows of $\underline{A}_a$.

Theorem 9. The rank of $\underline{A}_a$ for a connected graph of N_v vertices is not greater than $(N_v - 1)$.

Proof: The sum of all rows of $\underline{A}_a$ is a row of zeros, and elementary row operations cannot change rank.

Theorem 10. If T is a tree of a connected graph G of N_v vertices, removal of one branch of T creates a graph of two parts.

Definition 14. Fundamental Cutset. For a given tree T of a graph G, the cutset of G defined by a segregation of vertices by removal of a branch of T is called a fundamental cutset corresponding to T. S_1 and S_2 are defined such that branch orientation is from S_1 to S_2.

Theorem 11. For a connected graph of N_v vertices, there are $(N_v - 1)$ fundamental cutsets corresponding to a given tree.

Theorem 12. For a connected graph of N_v vertices, the rank of $\underline{Q}_a$ is at least $(N_v - 1)$.

Theorem 13. The rank of $\underline{Q}_a$ is exactly $(N_v - 1)$ for a connected graph of N_v vertices.

Proof: (a) Rank cannot be greater than $(N_v - 1)$, since the rows are linear combinations of the rows of $\underline{A}_a$. (b) Rank must be at least $(N_v - 1)$, since a subset of the rows of $\underline{Q}_a$ has rank $(N_v - 1)$.

Definition 15. Circuit matrix. The circuit matrix $\underline{B}_a$ for a connected graph is a matrix for which the typical entry is

$$b_{ij} = \begin{cases} +1 & \text{if } j^{th} \text{ element is contained in the } i^{th} \text{ circuit and orientation is same as circuit sensing} \\ -1 & \text{if } j^{th} \text{ element is contained in the } i^{th} \text{ circuit and orientation is opposite to circuit sensing} \\ 0 & \text{if } j^{th} \text{ element is not in the } i^{th} \text{ circuit for all circuits of the graph.} \end{cases}$$

Definition 16. Fundamental circuit. For some tree T each circuit made up of one chord and its unique tree path is called a fundamental circuit corresponding to T. The circuit sensing agrees with the chord orientation.

Theorem 14. For a connected graph of N_b branches and N_v vertices there are $(N_b - N_v + 1)$ fundamental circuits for any given tree.

Proof: There is one fundamental circuit for each chord.

Theorem 15. $\underline{Q}_a \underline{B}_a^T = 0$ if $\underline{Q}_a$ and $\underline{B}_a$ are the cutset and circuit matrices for the same connected graph G.

Proof: Consider the i^{th} row of $\underline{Q}_a$ and the j^{th} row of $\underline{B}_a$. Let $\underline{Q}_a \underline{B}_a^T = \underline{D}$ for which the typical entry is

$$d_{ij} = \sum_{k=1}^{n} q_{ik} b_{jk}$$

where col. order of $\underline{Q}_a = n$, q_{ik} and b_{jk} may be 0, 1, or -1.

Case 1. j^{th} circuit does not include elements for i^{th} cutset. Either q_{ik} or $b_{jk} = 0$ for each k.

Case 2. j^{th} circuit includes elements from i^{th} cutset. Elements from cutset must be included in circuit equation in pairs. If signs are same in cutset, they are opposite in circuit equation, etc.

Theorem 16. If $\underline{A}$ is order $m \times n$ and of rank n and $\underline{B}$ is order $n \times p$ and $\underline{AB} = 0$, then the maximum rank of $\underline{B}$ is $(r - m)$.

Proof: Let $\underline{D}_1 \underline{AD}_2$ be normal form of $\underline{A}$ where $\underline{D}_1^{-1}$ and D_2^{-2} exist $(\underline{D}_1 \underline{AD}_2)\, D_2^{-1}\underline{B} =$ $\begin{matrix} \underline{U} & 0 \\ \overline{0} & 0 \end{matrix} \begin{matrix} B_{11}^1 \\ B_{21}^1 \end{matrix} = \begin{matrix} B_{11}^1 \\ 0 \end{matrix} = 0$ where $\underline{D}_2^{-1}\underline{B} = \underline{B}^1$. $B_{11}^1 = 0$ and since premultiplying by D_2^{-1} cannot change the rank of $\underline{B}$, the rank is at most $(n - m)$.

Throem 17. For the rank of $\underline{B}$, the fundamental circuit matrix for some tree is $(N_b - N_v + 1)$ for a connected graph of N_b elements and N_v vertices.

Theorem 18. For a connected graph, the rank of $\underline{B}_a$ is exactly $N_b - (N_v - 1)$.

Proof: (a) Rank of $\underline{B}_a$ must be at least $(N_b = N_v + 1)$. (b) Since $\underline{Q}_a \underline{B}_a^T = 0$ cannot be greater than $N_b = (N_v - 1)$.

VITA

Prabhakar Mahadeo Ghare

Candidate for the Degree of

Doctor of Philosophy

Thesis: OPTIMAL RESOURCE ALLOCATION IN ACTIVITY NETWORKS

Major Field: Engineering

Biographical:

Personal Data: Born September 9, 1929 at Baramati, Maharashtra State, India, the son of Mahadeo Govind and Mrs. Janhavibai Ghare of Poona, Maharashtra State, India.

Education: Attended high school at Baramati; after graduation in 1944 entered the University of Bombay and studied at Fergusson College (Poona) and the College of Engineering (Poona). Received the Bachelor of Engineering degree in Electrical and Mechanical Engineering in December 1949. Entered Oklahoma State University in September 1961 and received the Degree of Master of Science, with a major in Industrial Engineering and Management, in May 1963. Completed the requirements for the Degree of Doctor of Philosophy in August, 1965.

Professional Experience: Employed by the Public Works Department of the State Government of Bombay, India as a construction engineer from 1950 to 1957. Served as a Senior Lecturer in Mechanical Engineering at S.V.V. University, Gujerat, India from 1958 to 1961. Served as a Research Assistant in the School of Industrial Engineering and Management at Oklahoma State University from September 1962 to August 1964. Assistant Professor of Industrial Engineering at Texas Technological College, Lubbock, Texas, since September 1964.

Professional Membership: American Institute of Industrial Engineers, Alpha Pi Mu, Operations Research Society of America, Society of the Sigma Xi.