OCCASION

This publication has been made available to the public on the occasion of the 50th anniversary of the United Nations Industrial Development Organisation.

DISCLAIMER

FAIR USE POLICY

CONTACT

Please contact publications@unido.org for further information concerning UNIDO publications.

For more information about UNIDO, please visit us at www.unido.org

UNITED NATIONS INDUSTRIAL DEVELOPMENT ORGANIZATION
Vienna International Centre, P.O. Box 300, 1400 Vienna, Austria
Tel: (+43-1) 26026-0 · www.unido.org · unido@unido.org

PROCEDURES FOR PROGRAMMING AND CONTROL OF IMPLEMENTATION OF

INDUSTRIAL PROJECTS IN DEVELOPING COUNTRIES

Page

PREFACE

CHAPTER 3. (Cont.)

Page

PAGE

Page

[illegible] planning for development [illegible] However, many times planning has [illegible] developing countries have [illegible] tise to the [illegible] schemes [illegible] and was [illegible] formulating [illegible] industrial development programmes without similar effort and expertise [illegible] voted to the course of action necessary to realize these programmes. [illegible] most countries failure to attain the intended development has frequently been failure to achieve successful or even satisfactory implementation. Experience has shown that developing countries encounter different shortcomings in their effort to implement industrial projects. [illegible] factors, which vary greatly, are of economic, technical, financial, administrative and managerial natures, while some of them are beyond the control of a country.

2. However, it became apparent at recent international meetings and from reports of experts of the United Nations Development Programme and other international advisers, that among the various factors contributing to these shortcomings, that of programming and control of implementation of industrial projects is of paramount importance. No formal approach and procedures have been available whereby implementation and programming for implementation are considered as a dynamic process. In the absence of these, no successful plan of operation nor effective implementation scheduling could be prepared and hence, projects will be hampered by delays and overrrun of costs and project implementation will fall short of expectations.

[illegible]

[illegible]

[illegible]

[illegible] programming for implementation [illegible] projects in the environment [illegible] if they are not dependent on highly [illegible] that is not readily [illegible] personnel [illegible] from computer [illegible]

[illegible]

[illegible] the process of subdivid[illegible] [illegible] their sequentia[illegible] [illegible] the estimating of the require[illegible] [illegible] most effective techni[illegible] [illegible] applied. Advanced [illegible]

[illegible] formulated it must be communicated [illegible] must be carried out [illegible] the project undertaking from the conceptual stage to [illegible]

[illegible] implementation plan must be [illegible] of each component acti[illegible] the plan as a whole must be practi[illegible] with respect to resource requirements [illegible] demands the utilization of various [illegible] classifications of labour skills as well [illegible] different types and sizes of equipment. The total requirement for any [illegible] at any instant must not exceed the level of its availability. [illegible] demands result in an implementation plan that is not realistic [illegible] therefore, that cannot be accomplished.

[illegible] changing regulatory requirements [illegible] prevent performance in accordance with original ideas. Therefore, the plans for programming and control of implementation should be dynamic so that [illegible] modifications [illegible] become necessary or when they become advantageous. If an implementation plan is not altered to reflect changes, it ceases to be valid. This is the fate of many, many implementation plans which have resulted from competent, and often expensive efforts and that originally have offered an ex[illegible] solution to the problem involved. Soon after the undertaking has commenced, changes have occurred but the plan has not been updated. Subsequent work then, is performed in the same manner as if it had not been programmed or, what may be still more hazardous, in accordance with a plan that has ceased to be valid.

It is essential, as indicated above, that an implementation plan be realistic and that it be kept updated. In addition, for truly effective implementation, there should be continued programming in greater detail as the job progresses and continual replanning of the existing strategy. No matter how excellent is the original programming there are practical limits for the amount of detail that should be developed at this stage. It must be recognised that changes are inevitable as the project proceeds and that extremely detailed overall programming of implementation is not justified. However, work in the immediate future can profitably be programmed in great detail. Therefore, this level of implementation programming for a limited period ahead should be carried on as the work progresses and should supplement the original, or updated, master implementation programming. There also should be a constant effort to improve job performance through replanning. It was pointed out in the preceding paragraph that unforeseen conditions may often [illegible] encountered that present problems and require updating. It is also likely that these unforeseen conditions may often offer opportunities for improvements. Sometimes, with a better knowledge of job conditions and of actual productivity [illegible] make it possible to make advantageous changes. While it is commendable to faithfully follow a well-conceived plan of implementation in order to reap the benefits of the skill and thought that has gone into its development, it is even better to continue to seek better solutions and when they are found, to be willing to change the implementation plan accordingly. It is, of course, important that the change receive very careful analysis in order to insure that it is in fact an improvement. Then it is important that the existing plan be updated, not abandoned.

11. Also, in order for an implementation plan to be executed successfully, those who manage the work should know the relative importance of the elements of the plan. This will allow them to concentrate efforts where they are needed most. For example, the effects of deviations in the timing or the sequencing of project activities can range from negligible to extremely serious. The information developed by the implementation programming procedure should indicate the nature of these effects to management. Intelligent control also requires appropriate corrective action after changes occur. First, it is necessary to determine whether a problem exists. Schedule updating can indicate the effects on other activities, on the project completion date, and on the timing of important intermediate events. Resource updating can indicate whether these requirements have been thrown out of balance or have become excessive. Having discovered the extent of the problem, it is then necessary to reestablish a realistic implementation plan. Reprogramming techniques should provide the means for accomplishing this objective. Moreover, they should allow the costs of these corrective actions to be determined. Then it becomes management's control function to use this information to make sound decisions.

12. In summary, then, implementation of an industrial programme or project needs to be intelligently programmed, and the implementation plan needs to be effectively executed. These two principal requirements involve:

[illegible] Programming of Project Implementation

- Definition of objectives.
- Breakdown of work into component activities.
- Statement of sequential relationships.
- Determination of methods, resource requirements and costs.
- Time estimates
- Calculation of resulting time schedule.
- Calculation of resulting resource schedules.
- Improvements by consideration of alternative strategies.
- Improvements by time-cost trade-offs.
- Improvement by resource allocation methods.

[illegible] Control of Project Implementation

- Formulation of a realistic implementation plan.
- Updating of plan as changes occur.
- Expansion of basic implementation plan in greater detail.
- Continual attempts to improve through reprogramming.
- Intelligent control based on implementation programming information.

B. The Scope of this Report

13. It is the purpose of this report to present operational techniques for identifying component activities of projects, determining their sequential relationships and represent them in a network diagram, making time-cost trade-off decisions and allocating resources. The time-cost trade-off problem arises from the fact that most of the activities into which the overall project is subdivided can be performed by alternative approaches requiring different amounts of time, resources and hence, expense. Generally it is true that those methods of performance which decrease the time requirements tend to increase direct, or variable costs. These direct costs will rise more rapidly in some cases than in others as work is expedited. If a project completion time is arbitrarily specified or is set by external controls, the time-cost trade-off procedure attempts to develop that combination of activity scheduling which meets the completion deadline with the lowest total direct cost. A more general problem would result if the procedure were applied to also determine the most favourable completion date. Here the additional fact that reductions in project duration result in lower indirect, or

fixed, costs must also be recognized. Then the time-cost trade-offs would be made with the objective of finding the schedule which gives the lowest combination of direct and indirect costs and, hence, the lowest total costs.

14. The resource allocation problem involves the determination of scheduling that satisfies resource restraints in as favourable a manner as possible. Most activities in a project require the use of one or more resources. If these requirements are stated and an initial schedule is developed, the number of units of each separate resource needed during each time period can be determined. If the resource demands at any time exceed the availability of any resource some activities must be rescheduled to reduce the concurrent requirements for that resource. Sometimes this rescheduling necessitates extending project duration. In such cases a principal objective is to minimize such extensions. Frequently a time-cost trade-off approach can be useful in these cases also. Excessive resource requirements can usually be satisfied by means other than mere rescheduling, but which involve activity replanning and higher costs. Although the topics of time-cost trade-offs and resource allocation have generally been considered separately, greater attention is needed to develop procedures that allow their interrelationships to be recognized. A secondary resource allocation problem is that resource requirements should be as constant as possible. Peaks and valleys in resource schedules invariably indicate uneconomic performance. Improvements can be achieved to some degree by rescheduling. The utilization of idle resources can also offer important opportunities for favourable time-cost trade-offs and is another instance of the interrelationships between these two topics.

15. A more complex resource allocation problem is created by the necessity to schedule several projects concurrently which draw resources from the same resource pools. This multi-project problem presents added difficulty because it involves a simultaneous consideration of a larger amount of data than demanded by individual analysis of the projects separately. It also requires proper consideration of the respective priorities of the various projects and of the mobility of resources from project to project.

16. Both the time-cost trade-off and the resource allocation problems involve a considerable amount of data. In both cases procedures developed to solve these problems have generally been mathematically complex. For these reasons it is not surprising that these techniques have usually been computer oriented. However, in developing countries the computer capability may not exist or may not be readily available at the level of project activity. Therefore, this report will present procedures that can be applied without computer processing. While this establishes real restrictions on methods that are practical to use, it does not mean that the results obtained will necessarily be inferior to results obtained by more sophisticated approaches. It appears fair to state that computer techniques for solving the time-cost trade-off and resource allocation problems, even where there is an ample supply of processing equipment available, have met with rather limited success. They have the inherent shortfall of not being able to recognize the interaction between activity costs when changes take place, for example, the time-cost trade-off relationships for an activity may depend on the time duration of one or more activities that may apportion some resources together. Good solutions require a considerable exercise of judgment. This judgment is needed not only in preparing data at the beginning of the calculation phase and analyzing results at the end of it, but judgment can also be very effectively applied during the intermediate steps of the solution process. However, the exercise of such judgment during the course of calculations is very difficult to successfully program in mathematical form. Its absence has severely limited the value of computer results in solving these types of problems.

C. The Approach of this Report

17. As stated before, this report is concerned with implementation of projects. "Project implementation" is an undertaking which has a definite beginning point and a definite end point, as contrasted with the cyclic type of operations characteristic of manufacturing work. In other words, by its very nature, the work involved in project implementation is non-repetitive. Very effective methods, the networking techniques, have been developed during

the past decade for the planning, scheduling, and control of project-type work. The best known of these networking techniques are the Critical Path Method (CPM) and the Programme Evaluation and Review Technique (PERT). Procedures to be proposed in this report will be based on networking methods.

18. As a prerequisite for the advanced networking procedures for time-cost trade-offs and resource allocation, a thorough understanding of the basic networking procedures is imperative. Therefore, Chapter 2 briefly discusses related methods for the development of a project implementation plan and its reduction to a paper model. This involves the presentation of a project implementation plan in a network diagram which is in fact a graphical portrayal of the precedence relationships between the various activities or tasks of a project from the beginning to the end of the plan. Furthermore, the methods included in Chapter 2 involve the estimation of time duration of project activities and provide computational procedures for establishing basic scheduling data whereby the relative importance of each activity in the overall project network can be determined. This is of essential importance since it draws the attention of project management to those activities which by their very nature limit or control the duration of a project. In addition, methods for the communication of the implementation plan and schedule to those who will implement them as well as methods for the application of data to project control are also included in Chapter 2.

19. In Chapter 3 the mechanics of the network diagram will be analyzed. Based on an understanding of the means by which changes are transmitted through the network, methods for updating project data will be presented. These methods, of course, are useful for the normal updating required as unforeseen changes occur in the performance of project activities. But they are also a key factor in the development of time-cost trade-off procedures where the ability to update certain project data as intentional changes are made in activity durations is essential. In Chapter 4 the time-cost trade-off problem will be presented in detail. Procedures for solving it will be given. Chapter 5 will discuss the resource allocation problem and offer an approach to its solution. Chapter 6 will discuss the multi-project resource

allocation problem and the necessary modifications of the single project method of solution. Finally, Chapter 7 will consider the interrelationships between the time-cost trade-off and the resource allocation problems. Methods for taking into account the effects of these interrelationships will be discussed.

20. It is not the intention in this report to stress cleverness in mathematical manipulations or to attempt to achieve mathematically optimum solutions. The stress is rather on simplicity and practicality of application at the project level by manual methods. Solutions that are short of theoretical perfection will be acceptable as long as they produce improvements in project performance which are realistic to achieve.

CHAPTER 2 - BASIC NETWORKING TECHNIQUES

A. Developing a Paper Model

21. After the objectives of a project have been defined it is necessary to programme the manner in which these objectives can be achieved. For this purpose it is advantageous to consider the overall job as consisting of a number of related but separate activities. It is simply not practical to attempt to work with the entire project as a single entity. Subdivision is not only necessary for implementation programming but also for other purposes such as time estimating, cost accounting, and project control. The resulting component activities should consist of logical subdivisions which involve work that, for some reason, is different from that which precedes or follows it. Factors governing subdivision of a project into component activities are discussed later in this chapter.

22. These component activities normally have very definite sequential relationships with one another which must be properly considered in programming of project implementation. If the project has any degree of complexity, the programmer should not attempt to carry these relationships solely in his head. Moreover, at some point his implementation plan must be communicated to others. A strictly verbal description will result in many different interpretations as well as the loss of much of the detail that has been developed through careful analysis It is practically essential for a programmer to reduce his ideas to a paper model in order to keep track of what he is doing and the restraints involved. It is also practically essential to use a paper model to transmit the results of his implementation programming efforts to others in a form that can be visually comprehended and referred to whenever necessary.

23. The type of paper model most commonly used to convey project implementation plans has been the bar, or Gantt, chart. This chart shows the programmer's breakdown of the project into its component activities, and it shows

the scheduling developed by him for each of these activities. While the bar chart conveys the project scheduling data quite effectively, its usefulness as a programming tool is quite limited. It does not clearly show the sequential relationships which the programmer must constantly keep in mind. It does not force the programmer to consider all the restraints that may be involved in scheduling, since it does not require that he show all the activities that must be completed before another can begin. It does not convey to others whether he has or has not given consideration to the various prerequisite activities. While a bar chart may technically satisfy a specified requirement to furnish a documented implementation plan, careful examination frequently shows that the plan is subdivided to an insufficient extent, that it totally omits many restraining activities, that it does not indicate whether careful analysis has been made or not, and that it will need further interpretation to be fully understood. An implementation plan presented by means of a bar chart may be based on careful and masterful programming, or it may be the result of sloppy and incompetent programming. Those who review the paper model find it difficult to judge which is the case because insufficient detail is shown.

24. A very simple bar chart having only four activities is shown in Figure 1. In this example it might be conjectured that Activity B is dependent on Activity A and that Activity D is dependent on Activity B, since in both cases one starts at the time when the other is completed. It might also be suspected that Activity C's performance depends on the partial completion of Activity B. Activity descriptions may also provide hints in actual cases. However, in charts involving many bars such deductions would be much more difficult to make and much more likely to prove incorrect. There is no evidence that other relationships exist between the activities shown nor is there any reason to believe that every activity offering a potential restraint to those shown has also been included on the chart. For example, if the reviewer thinks of another activity that might affect the starting time of Activity C, he has no way to know whether the programmer also considered this possibility or, if so, what was the basis for the conclusion that he

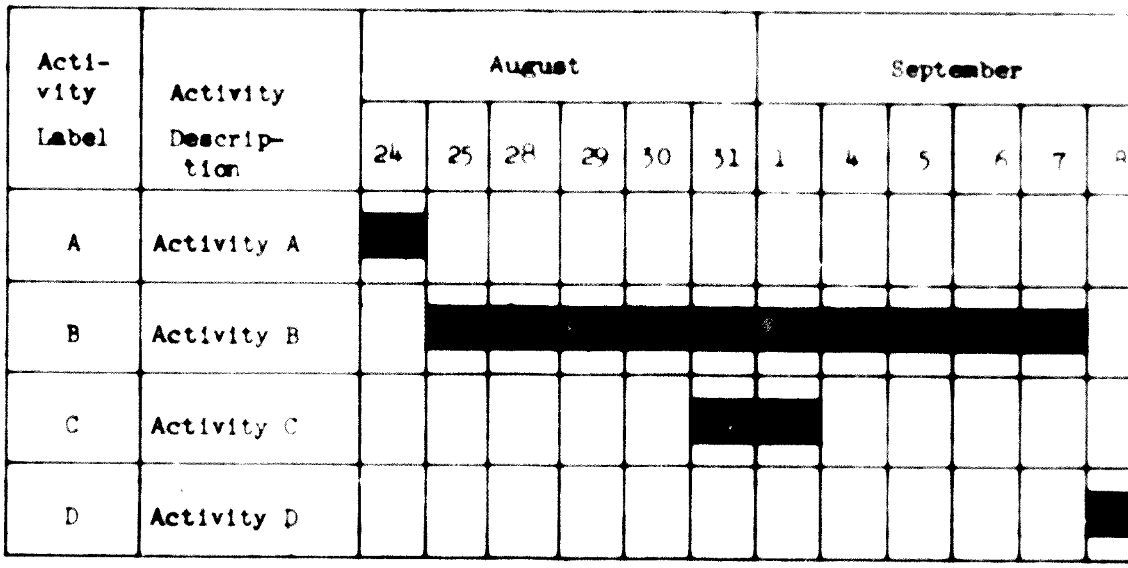


Figure 1 - Simple Bar Chart

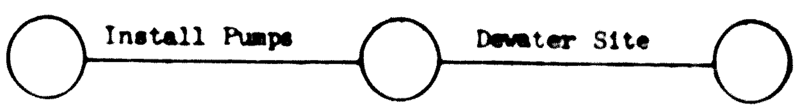


Figure 2

reached. So it may be concluded that the bar chart has shortcomings as a programming aid both to programmers and to others who must review and understand the results.

25. The network diagram has been introduced to overcome the defects of the bar chart and, thereby, to furnish a more effective form of paper model. It is basically a programming and controlling tool and is developed prior to the determination of scheduling data. However, it is subsequently used to provide the relationships necessary for calculating the schedule and is sometimes plotted to a time scale to show the schedule.

26. The network diagram requires that project activities be well identified and their sequential relationships be shown. There are several variations of network diagramming methods. However, a network may be represented mainly by two methods: arrow diagramming and precedence diagramming.

A.1 Diagramming Methods

A.1a Arrow diagramming:

27. It is the most widely used approach which was employed in both the original Critical Path Method (CPM) and Project Evaluation and Review Technique (PERT). In this type of diagramming, a project activity is represented by an arrow. An activity is a part of a project and is sometimes called a task. It requires resources such as manpower, equipment and time for its performance. It has starting as well as ending points. The terminal points of the arrow representing an activity are node points in the resulting network and they represent events, e.g. the start or finish of the activity. If one activity follows another, they have a common node where the head of the arrow representing the preceding activity is connected to the tail of the arrow representing the following activity. For example, in implementing a project and after the land required to build factory buildings has been acquired, it may need certain preparation before construction work can start. In this case, the two activities shown in Figure 2 may take place. If more than one activity precedes a following activity, the heads of the arrows representing the preceding activities merge at a node which is also the tail of the following activity. This is illustrated in Figure 3. In a similar way, if there is more than one following activity for a given activity, the arrows representing each of the following activities have their tails at a

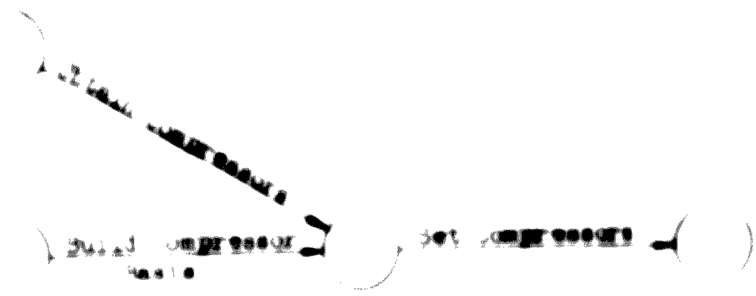


Figure 3

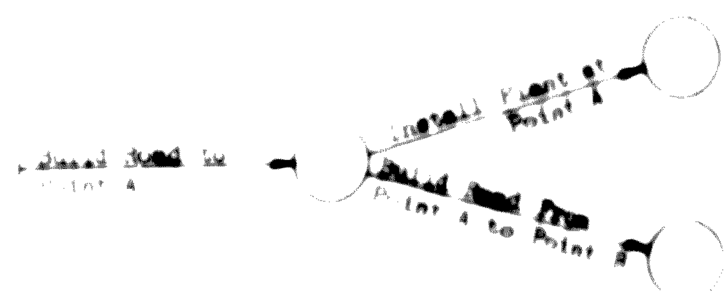


Figure 4

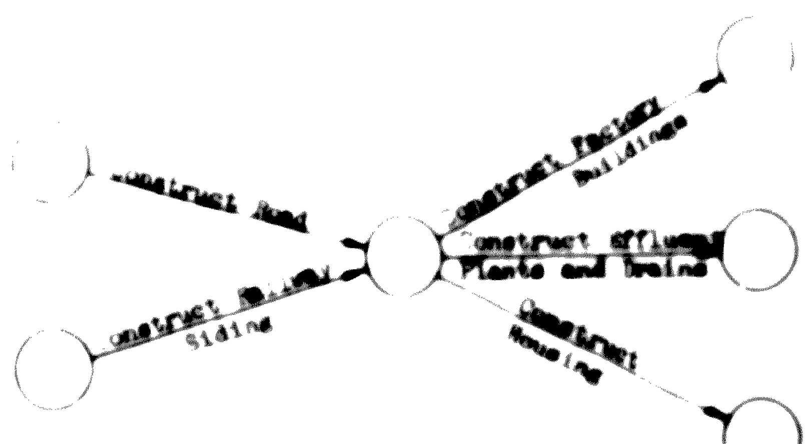


Figure 5

[illegible]

[illegible] arrow diagramming [illegible] arrow diagramming [illegible]

[illegible] is shown in Figure [illegible]

Arrow diagramming [illegible] dummy activities or dummies [illegible] time durations [illegible] reason for requiring [illegible] correct sequencing relationships [illegible] by bringing the terminals of the arrows [illegible] In other words [illegible] a number of preceding [illegible] ding activity but other [illegible] ities. Suppose for example [illegible] be completed before acti[illegible] to add an activity [illegible] but which is not dependent on activity [illegible] this activity cannot commence at the [illegible] cate a dependence on activity A [illegible] duce an artificial or dummy arrow representing a restraint of zero time duration as illustrated in Figure 2.

Dummy activities only define precedence relationships [illegible] tion occurs. A dummy activity is sometimes called a dependency arrow and is

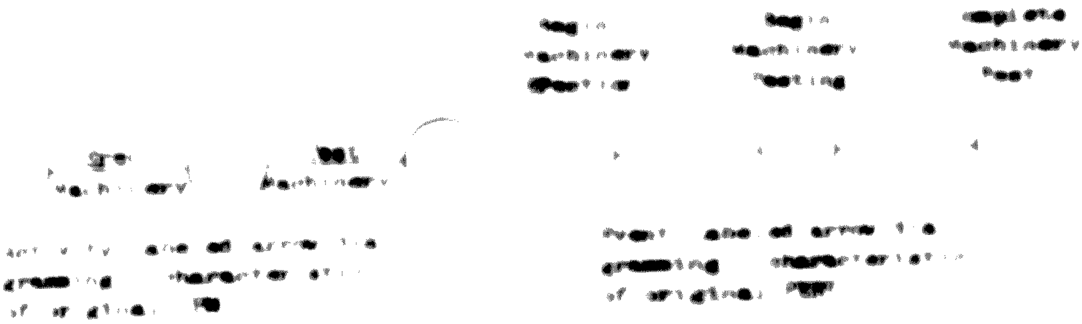


Figure 1

Figure 2

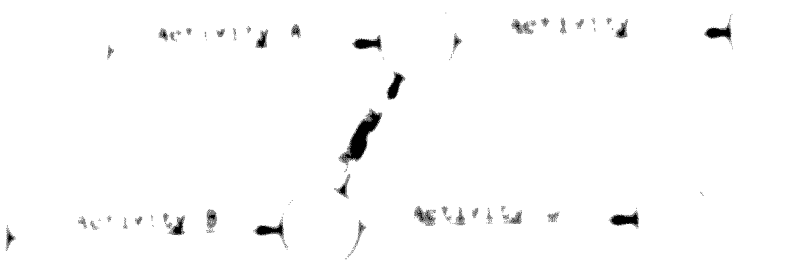


Figure 3

[illegible]

Arrow diagramming [illegible] arrow [illegible] the tail of the arrow [illegible] the head of the arrow representing [illegible] must always have a number greater than that of its tail [illegible] is therefore identified by two numbered events [illegible] of the arrow representing [illegible] diagramming [illegible] indicates activity sequential relationships. In case of diagramming to time-scale [illegible] from left to right [illegible] arrow representing it. The length of its projection on a horizontal time scale indicates its duration.

4.1[illegible] Precedence Diagramming

[illegible] Another approach to network diagramming utilizes the nodes of the network to represent the activities rather than the events. The lines, then, indicate the sequential relationships between activities. It is an activity oriented method. This type of diagramming has been referred to as "circle" or "circle and connecting line" diagramming[4] and as "Activity-on-Node" diagramming.[5] However, more recently it has become known as "Precedence Diagramming" and has received increasing favourable attention.[6]

[illegible] In precedence diagramming the relationships between activities [illegible] and [illegible] of the previous example would be shown as in Figure [illegible]. The arrow heads on the sequence lines are unnecessary if the diagram is drawn with the flow of work always from left to right.

---

4. John W. Fondahl, "A Non-computer Approach to the Critical Path Method for the Construction Industry", Technical Report No. 9, Dept. of Civil Engineering, Stanford Univ., 1961.

5. Joseph J. Moder and Cecil R. Phillips, "Project Management with CPM and PERT", Appendix 2-1, Reinhold Publishing Corp., 1964.

6. <u>Engineering News Record</u>, "Contractors Shift from Arrow to Precedence Diagrams for CPM", 6 May 1965, pp. 52-53.

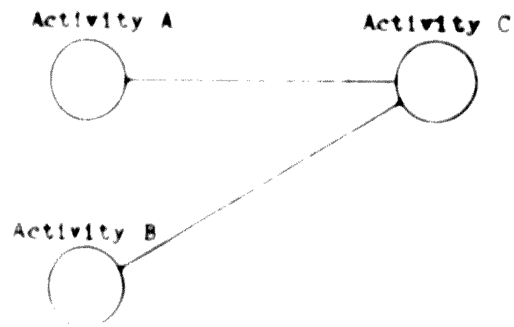


Figure 9

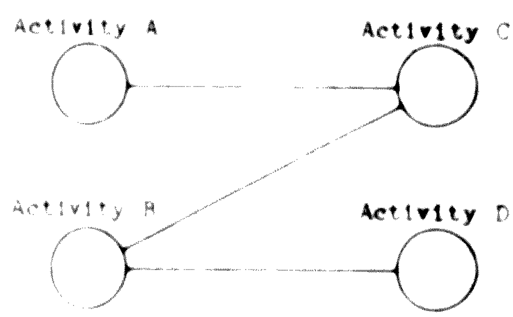
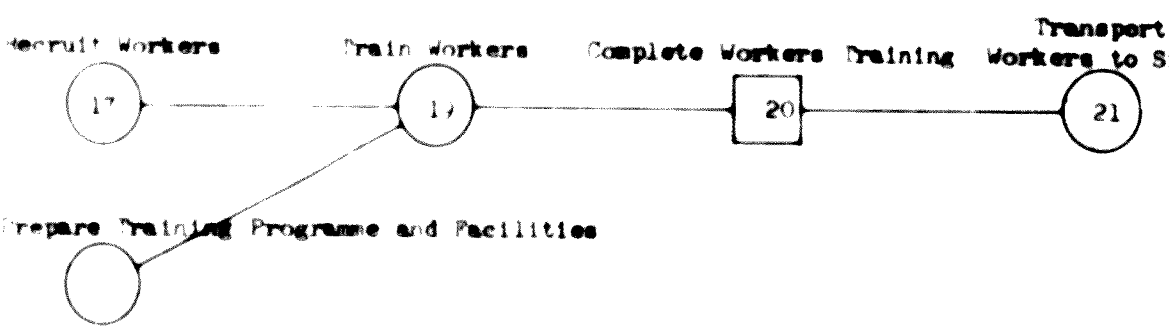


Figure 10



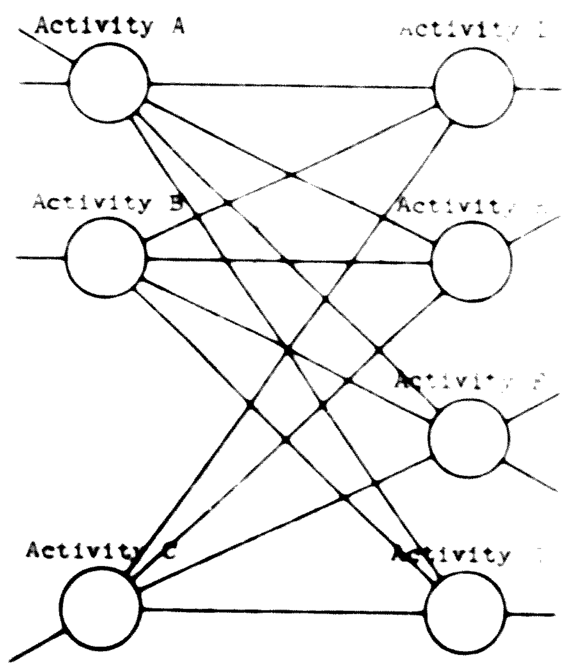
Precedence Diagramming with a Key Event

Figure 11

When the Activity I is added, it merely requires inserting the corresponding node symbol and drawing the appropriate sequence line as in Figure 10. This is a very simple and direct approach that requires no special diagramming skill. Activities are represented by a symbol, e.g. a circle located at any convenient position in the diagram, labeled, and lines are drawn to the preceding and following activities. There is no reason for being concerned with dummy activities. It is true that, in effect, the sequence lines are actually dummy activities, but the diagrammer needs to give no conscious attention to this fact.

54. Due to the nature of the work involved in project implementation one may need to represent one or more events on the network diagram as significant milestones. These key events may indicate the commencement or completion of certain important activities or stages of a project. This is illustrated in Figure 11. Representing an event on the diagram can be accomplished by considering an event to be an activity of zero time duration and may be labeled. A different symbol may be used for events, for example a square, if desired as shown in Figure 11. Only those events having special significance need be shown.

55. Nevertheless, the inclusion of some events in the network diagram may facilitate and simplify the construction of the precedence diagram even if their importance does not warrant their representation. This is particularly true when a number of activities cannot be commenced unless another number of activities are completed. For example, in Figure 12-a any activity of the Activity Group II (Activities D, E, F and G) cannot be commenced before all the activities in the Activity Group I (Activities A, B and C) are completed, and consequently there are twelve sequence lines indicating these sequential relationships. However, in order to simplify and improve diagramming the network the event signalling this instant - the completion of Activity Group I and the commencement of Activity Group II - justifies its representation as shown in Figure 12-b, even if it has no significance as a milestone.



Activity Group I
(Activities A, B and C)

Activity Group II
(Activities D, E, F
and G)

Figure 12-a

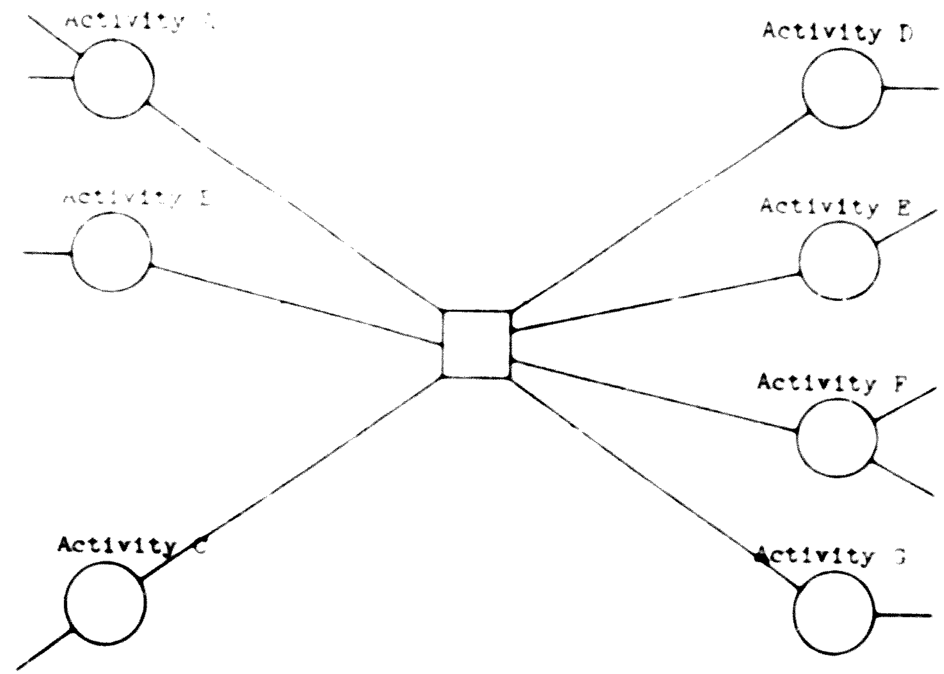


Figure 12-b

Figure 12 - Precedence Diagramming with an Event for Simplification

Furthermore, the representation of events also provides interface connecting points in network breakdown or subnetworking as will be discussed in Chapter 3 below.

36. Precedence diagramming is much simpler to apply and to teach than arrow diagramming. While there are other advantages also, such as those inherent in its labeling system,[1/] it is primarily due to its simplicity that precedence diagramming has been chosen for use in this report. By minimizing the required skill in the mechanics of diagram construction, those who possess the other necessary and more important skill, i.e. a thorough knowledge of the work to be accomplished, will be able to develop the diagram themselves.

37. It takes shorter time and less effort to diagram a project implementation plan by using precedence diagramming rather than arrow diagramming since the latter at least requires care in locating activity arrows at the event nodes where they merge and depart. Also, in the case of diagram revision to cope with changing conditions, diagrams constructed by precedence diagramming lend themselves more readily to revision than those constructed by arrow diagramming. Adding an arrow to an arrow diagram implies certain changes in the position of some arrows already drawn on the diagram. However, with precedence diagramming, addition of an activity to the diagram is merely locating a node in a suitable position on the diagram and connecting it with the preceding and following activity nodes by sequence lines as has been mentioned before and shown in Figures 9 and 10 respectively. This promotes frequent updating as conditions necessitate.

38. The diagram is the basis for the application of all the remaining network techniques, and it is essential that it be a realistic representation of the work to be performed. Otherwise even the most sophisticated procedures and processing equipment are useless in obtaining worthwhile results. Therefore, it becomes very important that those with the best knowledge of the work be the ones to develop the diagram and to revise it as necessary.

1/ For a more detailed analysis and comparison of the two diagramming systems see John W. Fondahl, "Methods for Extending the Range of Non-Computer Critical Path Applications", Technical Report No. 47, Dept. of Civil Engineering, Stanford University, 1964.

Keeping diagramming mechanics simple makes this practical. For this reason simplicity becomes an extremely important characteristic.

39. One point which might be of interest in cases of computer utilisation is that computers use the dual numbering system used by arrow diagramming. Although precedence diagramming uses the single numbering system, as mentione before, the dual numbering of sequence lines can sufficiently be used with computers.

40. As a simple illustration of network diagramming by the precedence method, suppose that the project previously shown in the bar chart of Figure is to be represented by this approach instead. Activity B must be subdivided further since overlapping is not permitted in network diagramming. (Some computer programmes appear to permit overlapping through the use of "lag factors" but actually the subdivision is still performed internally by the computer.) Activity B1 is a portion of Activity B that can be described separately and that must be completed before Activity C may commence. Activity B2 is the remaining portion of Activity B which may be performed concurrently with Activit C. A new Activity X is also added. This is work, perhaps, that is not done on the job site but that is an external activity such as the furnishing by others of an item of equipment to be installed, or the checking and approval of plant drawings. Since this is work that is not directly performed by the programmer's own organization, he will frequently omit showing it on the bar chart even though he may have considered the resulting restraint. However it is an activity which must be shown on the network diagram since it requires time to accomplish and must be completed before Activity C can commence Its inclusion in the diagram results in a better paper model because it clearly communicates to others who review or use the implementation plan the fact that this activity has been considered. Although at the time that project implementation is programmed this activity may not be a controlling restr due to later delays it may become so and may affect the scheduling of other project activities or even of project completion.

41. The activities into which the project is subdivided and their sequential relationships, then, are as follows:

Activity	Activity Duration Time in Days	Must Precede Activity
A	1	B1, X
B1	4	B2, C
B2	6	D
X	2	C
C	2	D
D	1	- -

A precedence diagram might be drawn in the form shown in Figure 13. This would serve as the paper model of this project and would provide the basis for the application of other networking techniques. It conveys the project implementation plan more effectively than the bar chart of Figure 1, but it does not convey the project schedule. However, for the sake of comparison, Figure 14 shows three methods of presenting the same simple project shown in Figure 1 and Figure 13. Figure 14-a reproduces the bar chart of Figure 1 and does not need any further explanation. Figure 14-b and Figure 14-c present the network of the project implementation.plan as an arrow diagram and a precedence diagram respectively. To make the comparison more effective, both diagrams are drawn to a time scale which is shown at the bottom of Figure 14. The time scale, however, shows both calendar dates and working days. In Figure 14-b each activity is represented by an arrow, its horizontal projection on the time scale denotes the activity duration time. Every arrow starts from and terminates at a node or an event and as previously mentioned, each event has a number and hence each arrow is identified by two numbers. For example, in Figure 14-b, Activity A is represented by the arrow 0-1, Activity B1 by 1-2, ... etc. To maintain the precedence relationships between activities B1 and C illustrated in Figure 13, a dummy activity is needed to show this relationship. This is represented by the dashed line arrow 2-3 which indicates that Activity C (represented by the arrow 3-4) cannot be commenced until Activity B1 (1-2) is completed. Also, in Figure 14-b the last portions of Activities A (1-3) and C (3-4) are dashed. This means that these two activities are expected to be completed prior to the occurence of the two succeeding events; events 4 and 5 respectively and, therefore, the dashed portion denotes the leeway or slack which each of these activities has. This is due to the fact that while Activity B1 (1-2) has a duration time of 4 days as shown in

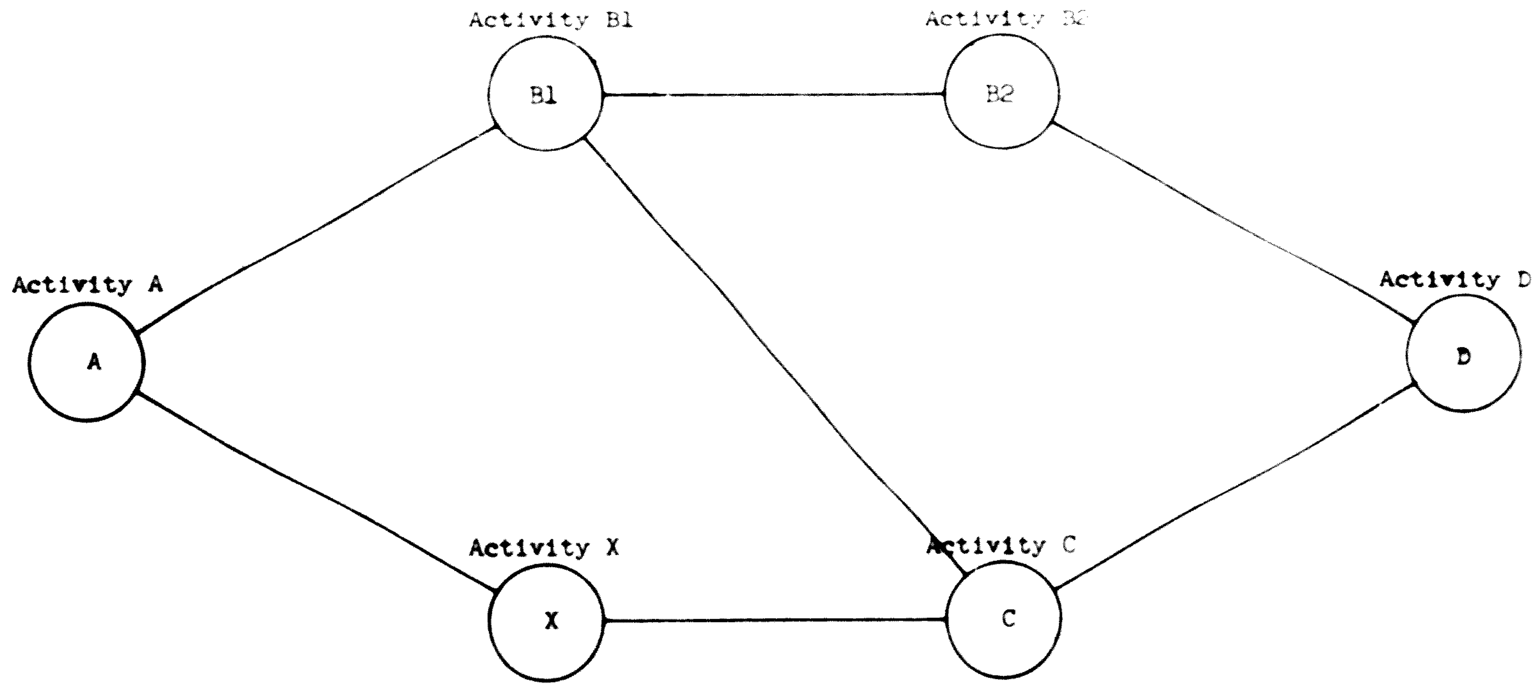


Figure 13 - Precedence Diagram

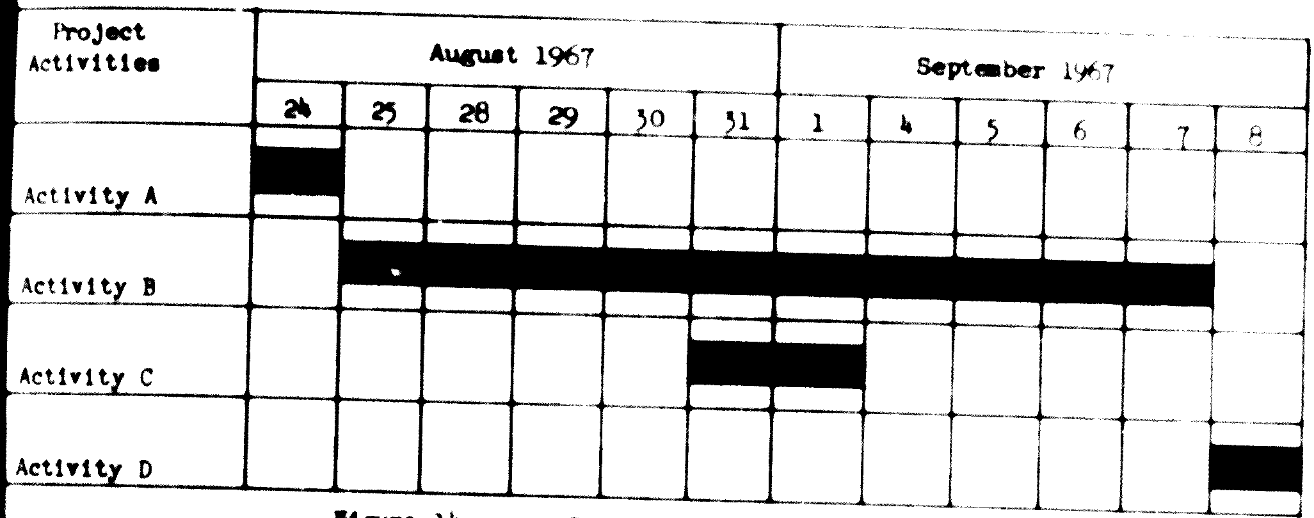


Figure 14-a - Bar Chart

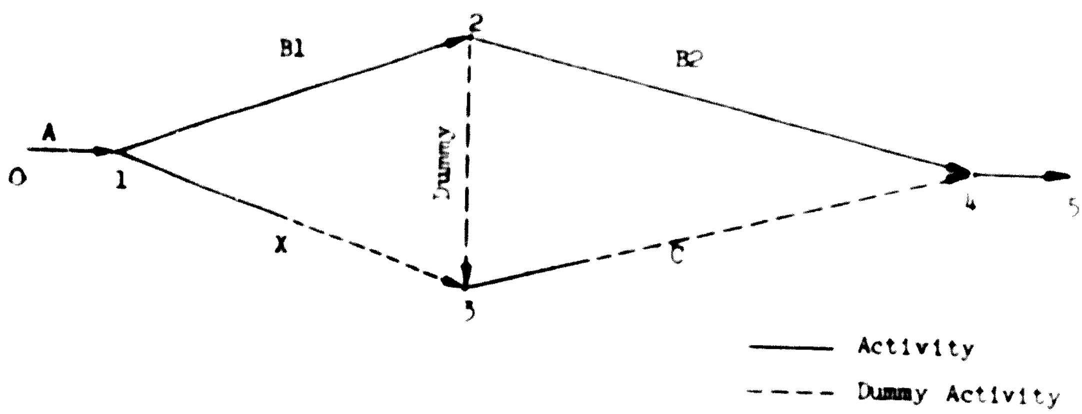


Figure 14-b - Time-Scaled Arrow Diagram

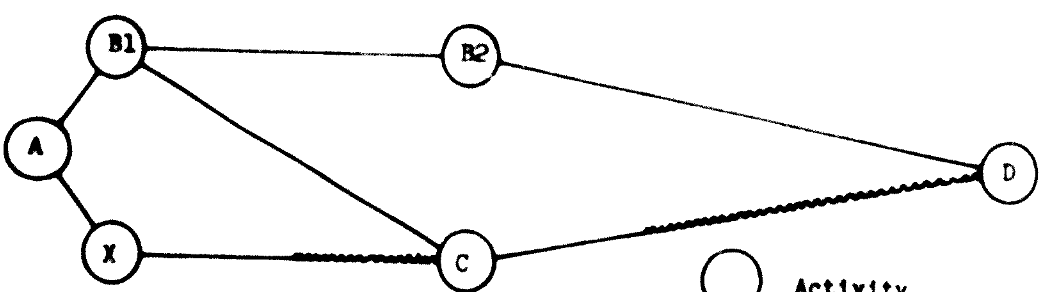
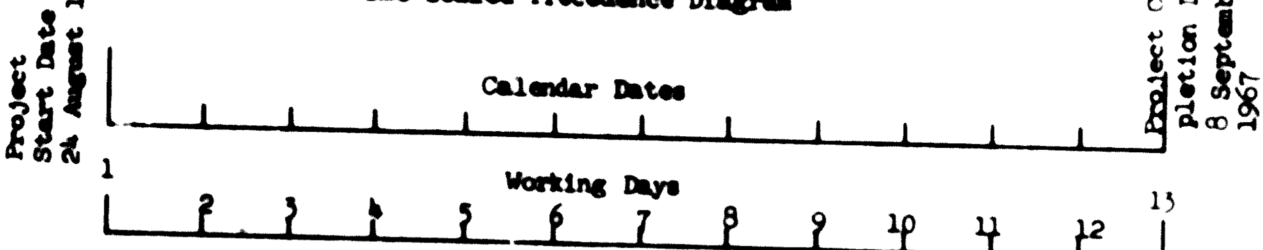


Figure 14-c
Time-Scaled Precedence Diagram



14 - Comparison of Presentation Methods for a Project Implementation Plan

the table on page 25, Activity X (1-3) has a duration time of only 2 days, and since event 3 cannot occur before the end of the 5th working day, i.e. the beginning of the 6th working day as can be shown in Figure 14-b, Activity X (1-3) has a leeway or slack of 2 days. This means that within this amount of time the finish time of Activity X (1-3) can be delayed without affecting the occurence time of event 3. In the same fashion, it can be shown that Activity C (3-4) has a leeway or slack of 4 days and within this period of time its finish time can exceed its expected finish time without affecting the occurence time of event 4. One more point is that these activities lie on the path 1-3-4 and the expected time to accomplish them is 2+2 = 4 days, but with reference to the time scale at the bottom of Figure 14, it can be seen that the time period between the occurence of event 1 and event 4 (the starting and terminating events of the path 1-3-4 respectively) is 10 days which indicates that this path has 6 days of slack. In such a case the path 1-3-4 may be called "slack or non-critical path". On the other hand, Activities B1 (1-2) and B2 (2-4) have no slack. They are represented by solid arrows between events 1 and 2 and between events 2 and 4 respectively; where the expected time to accomplish them is 4+6 = 10 days, the same as the time period between the occurence of events 1 and 4. Therefore, the path 1-2-6 has no slack and may be called "critical path".

42. The precedence diagram in Figure 14-c illustrates the same relationships. Instead of arrows, nodes are used to represent project activities and sequence lines are used to show precedence relationships. The zigzag portion of the sequence lines between Activities X and C and Activities C and D denotes lag times of 2 and 4 days for these two sequence lines respectively. In this particular case, these lag times equal the amounts of slack of the preceding Activities X and C respectively and the path X-C-D may be called "slack or non-critical path" since it has again 6 days of slack. The other sequence lines B1-B2 and B2-D are solid denoting that the path B1-B2-D has no slack and hence it may be referred to as "critical path". All these terms are discussed in detail in Chapter 3.

A.2 Development of the Network Diagram for a Project Implementation Plan

43. A basic step for programming the implementation of a project is to start constructing an original network diagram for project activities and their interrelationships. As programming of implementation goes on the original network diagram can be developed. This will assure that all project activities and their sequential relationships are considered and represented.

44. To start this phase of implementation programming, the following steps may be considered:

A.2a **Preparation of a list of project activities**

All project activities may be listed as they come to mind. Although this may be helpful for beginners, the list might not be necessary after they acquire experience in diagramming. However, for complex projects having a great number of activities, an activity list might be of importance. At this stage, however, a decision must always be taken as to the level of breakdown of project activities. This depends on the following factors

i. **Nature of work involved**

Some project activities or activity groups contain different work contents and need for their accomplishment different types of resources (labour, equipment, etc.). For example, aggregate activities such as "Finalisation of Project Plans", "Construction of Buildings" and "Installation of Machines and Equipment" have different types of labour and machinery to perform them, and hence they can be considered separately.

ii. **Place and time of work**

Work undertaken at different locations or at different times may be considered as separate activities.

iii. **Supervision and responsibility for work**

This may be taken as a factor for determining the level of project breakdown. If supervisors or key personnel who can assume responsibility for some parts of a project are scarce, activities may be aggregated so that each [illegible] accurate [illegible] the nature of work [illegible] interdependent [illegible] and the like. Also [illegible] different departments [illegible] separated.

[illegible] of work

[illegible] sometimes [illegible] where [illegible] by a number of agencies [illegible] development work [illegible] of these [illegible] stages of [illegible]

[illegible] Construction of an original project network diagram

[illegible] paper [illegible] programmers can now develop the original project network diagram. [illegible] project activities are determined and each is represented by a node at the [illegible] left hand side of the sheet of paper. These nodes are then [illegible] each activity is represented in the diagram, the questions "What activities must be completed before this activity can commence" and "What activities may commence when this activity is completed" must be asked. As these questions are answered, the corresponding activities are added to the diagram, labelled and connected with other activity nodes with sequence lines so that the appropriate sequencing relationships can be shown. Then the questions and the same procedure are repeated for each of these activities and this continues until the last project activity has entered the network diagram. The rules for network diagramming which require that each activity shown be completed before the following activity can commence necessitate a further breakdown of work that overlaps other work, for instance, when one activity can commence before a preceding activity [illegible] pleted before [illegible] which represents [illegible] more [illegible] and the [illegible] before another [illegible] a more [illegible] tasks that [illegible]. This means more effort must be expended. This extra effort [illegible] and forces the programmers [illegible] conveys the implementation plan more [illegible] to others.

[illegible] There is another procedure [illegible] programmers may start by entering the [illegible] of the sheet of paper (panel [illegible]) and work backwards adding the activities that must be completed before the commencement of the activity [illegible] entered [illegible] the sheet of paper. This procedure [illegible] activities are entered here again appropriate sequence [illegible] as the network diagram is developed. During the construction of this network diagram arrow heads are put on sequence lines to show precedence relationships.

[illegible] Although the first procedure is more common, the programmer may choose the one that is more suitable for him. Programmers should be aware of how to present a project implementation plan network in a way that may assist others engaged in the project to understand it, to see clearly activity sequential relationships, and to be able to review it.

[illegible] network diagram [illegible] that the work breakdown [illegible] the nodes of [illegible] undertaken under the [illegible] programming and control [illegible] broken down into a [illegible] The original network diagram [illegible] presentation of the project [illegible] drawn free-hand and [illegible] disciplined or logical grouping [illegible] that a few of the sequence [illegible] nodes from right to left instead [illegible] direction of the flow of work. For [illegible] sequence lines during the construction of [illegible] network diagram in order to show precedence relationships [illegible] cancelled and thus they might be crossed [illegible] figure [illegible]. However, the original network [illegible] work sheet for programmers and for those interested [illegible] necessary activity sequential relationship [illegible] original network scheduling computation.

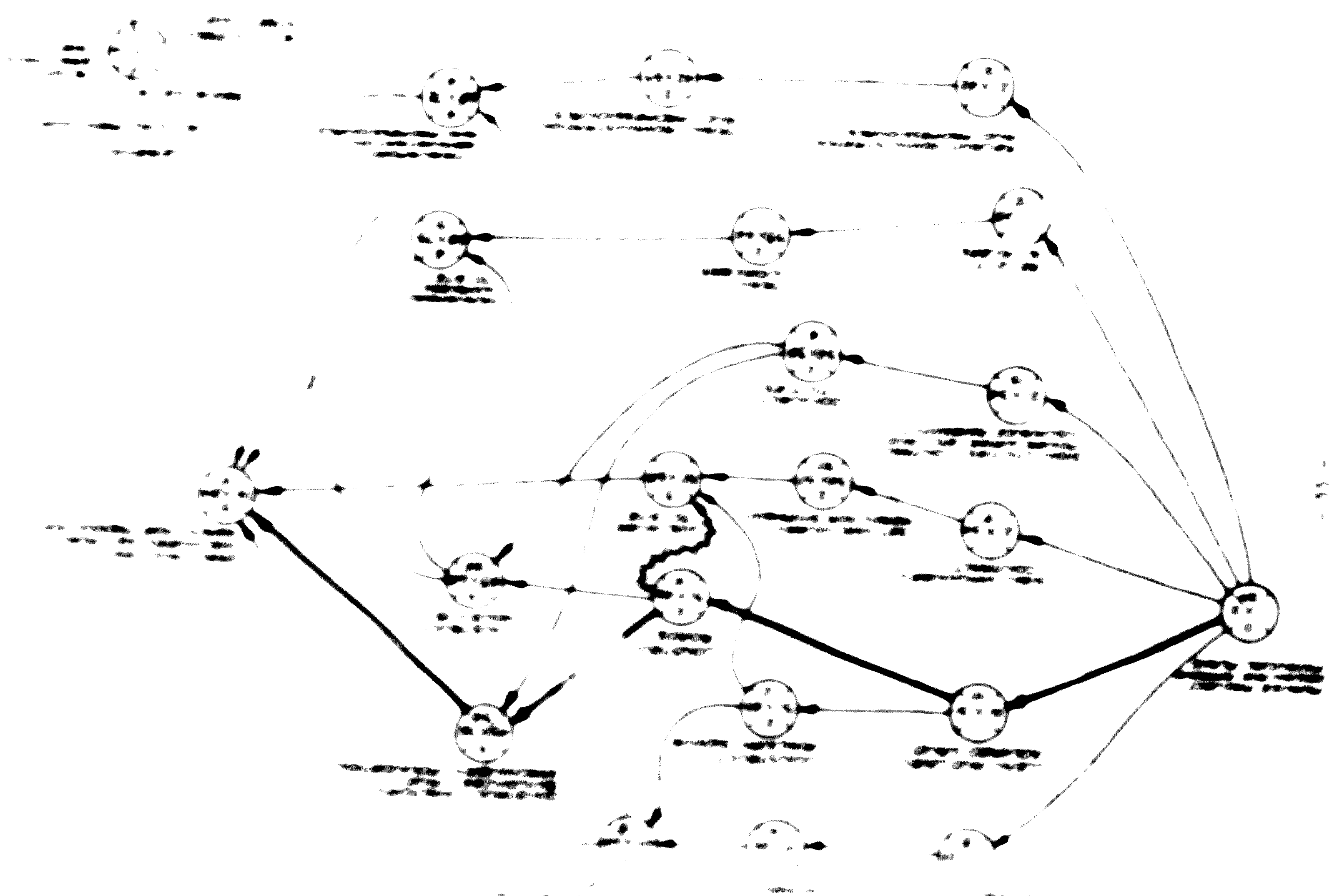
3e.1.2. Recompilation of the network diagram of a project implementation plan

[illegible] presentation of a project implementation plan [illegible] network diagram is required in order to:

[illegible] evaluation of the plan and in making the necessary adjustments;

[illegible] obtaining more easily, data and information [illegible] the network diagram for further computations or periodic revision;

[illegible] in making necessary computations on the network diagram.



lend itself more readily for updating

serve as a common language to those engaged in project implementation.

4. However, it is advantageous to postpone the redrawing of the original network diagram until it is proven to be satisfactory. For this purpose, using activity time duration estimates, the initial basic time computations may be made by hand on the original diagram to find out whether the plan of implementation meets requirements. Here programmers need no more than to make the "forward pass" of basic time computations starting from the beginning to the end of the project where the early start and early finish times of project activities are calculated and hence the early project completion date. The early start time of an activity is the latest or biggest of the earliest finish times of all activities preceding it and the early finish time of an activity is its early start time plus the activity duration time. Then in order to determine the critical activities and hence critical path(s) without computing activity late finish and late start times and total floats, we can start from the last project activity and go backward to the initial activities along the sequence lines connecting those activity nodes of equal early start and early finish times. For example, in Figure 15, starting from the last project activity and going backward to the initial activity(s) as mentioned above, it can be seen that the activities "Erect Machinery, Get Plant Ready for Production", "Construct Factory Buildings and Machinery Foundation", "Construct Main and Access Roads", "Clean and Level Land" and "Finalize Project Designs and Approve Financial Plans" are critical and hence the path along which they lie.[5] Now if the project duration is found to meet requirements, and if the activity sequential relationships included in the implementation plan (mainly those concerning the critical activities) are found acceptable, then the original project network diagram may be redrawn in a better form to serve all the functions previously stated. On the other hand, if the project duration does not meet requirements and is

---

[5] Basic scheduling computations are discussed in detail in Section B below.

unsatisfactory, the project implementation plan needs to be reconsidered. Re-programming of project implementation may be indispensable and hence certain changes in the network diagram may be incorporated. It is worth mentioning that consideration of the critical activities may suggest some changes in activity breakdown, i.e. level of detail of project activities presented on the diagram, so that some activities, if possible, may be overlapped. After these changes have been made and the project duration as well as the implementation plan have been reviewed and accepted, the redrawing of the network diagram can then take place. However, for better arrangement of project activities and sequence lines, the sequence step or time-scale diagramming methods can be used in redrawing the network diagram.

Sequence-step Procedure

49. A suitable scale of sequence steps can be chosen and vertical lines can be drawn at each sequence step on this scale. The size or length of a sequence step is arbitrary. It should be selected to provide a clear graphical portrayal of the project network diagram. After numbering the sequence steps and hence these vertical lines, nodes representing activities (and perhaps key events) can be located on the vertical lines according to their sequence step numbers. Each node is placed horizontally to the right of activities preceding it. Next to each node a brief description of the activity it represents is included. As shown in Figure 16, the sequence step of each activity is determined by giving initial activities a sequence step number of zero, plotted on the corresponding vertical line. Then, each other activity is given a sequence step number which is one greater than the highest sequence step number of activities directly preceding it, and so on. The vertical position of the nodes on the network diagram should be chosen so as to achieve disciplined or logical groupings, and to enable sequence lines to be clearly drawn. The nodes are then numbered. Numbering starts from the top of the first sequence vertical line by giving the node in the upper left hand corner of the diagram a number of one, then proceeding downward along that

vertical line giving the second node a number of two, and so on. Then, proceeding to the top of the second line of activities and going again down that same line continuing this way until the farthest node to the right of the network diagram is reached. This provides that no activity will precede another activity of a lower number. The sequence step diagram provides an effective step toward a final diagram or time-scale diagram after having accomplished all necessary computations and revisions. However, the sequence step diagram may sometimes be considered as the final diagram and activity duration time as well as other basic scheduling data may be included next to or inscribed into corresponding nodes as illustrated in Figure 16. This depends on the need of the diagrammer and the people who will use the diagram, and the degree of accuracy with which activity relationships need to be represented.

Plotting the network diagram on a time-scale

50. The network diagram can be plotted on a time-scale from the original or the sequence-step network diagram. The need for a time-scale diagram stems from the fact that neither the original rough diagram nor the sequence-step diagram represents project activities in their appropriate time relationships. The sequence-step network may show two or more activities with the same sequence number (located on the same vertical sequence line) which in reality are performed in different time phases. If a network diagram is plotted to a time-scale it shows the real activity interdependencies, at least at the beginning of project implementation. The sequence-step and the time-scale diagrams can be used together. In such a case, a sequence-step diagram is constructed for the entire implementation plan. Then, at each time interval (one month or two months) perhaps a detailed time-scale diagram can be prepared for the portion of the project that is going to be implemented during the next time interval. The selection of the time interval depends on the nature of the project, the degree of detail required, and the complexity of the work. When conditions change network diagrams constructed to time-scale need a great deal of effort and time to be updated. Therefore, in the case of large and complex projects, especially

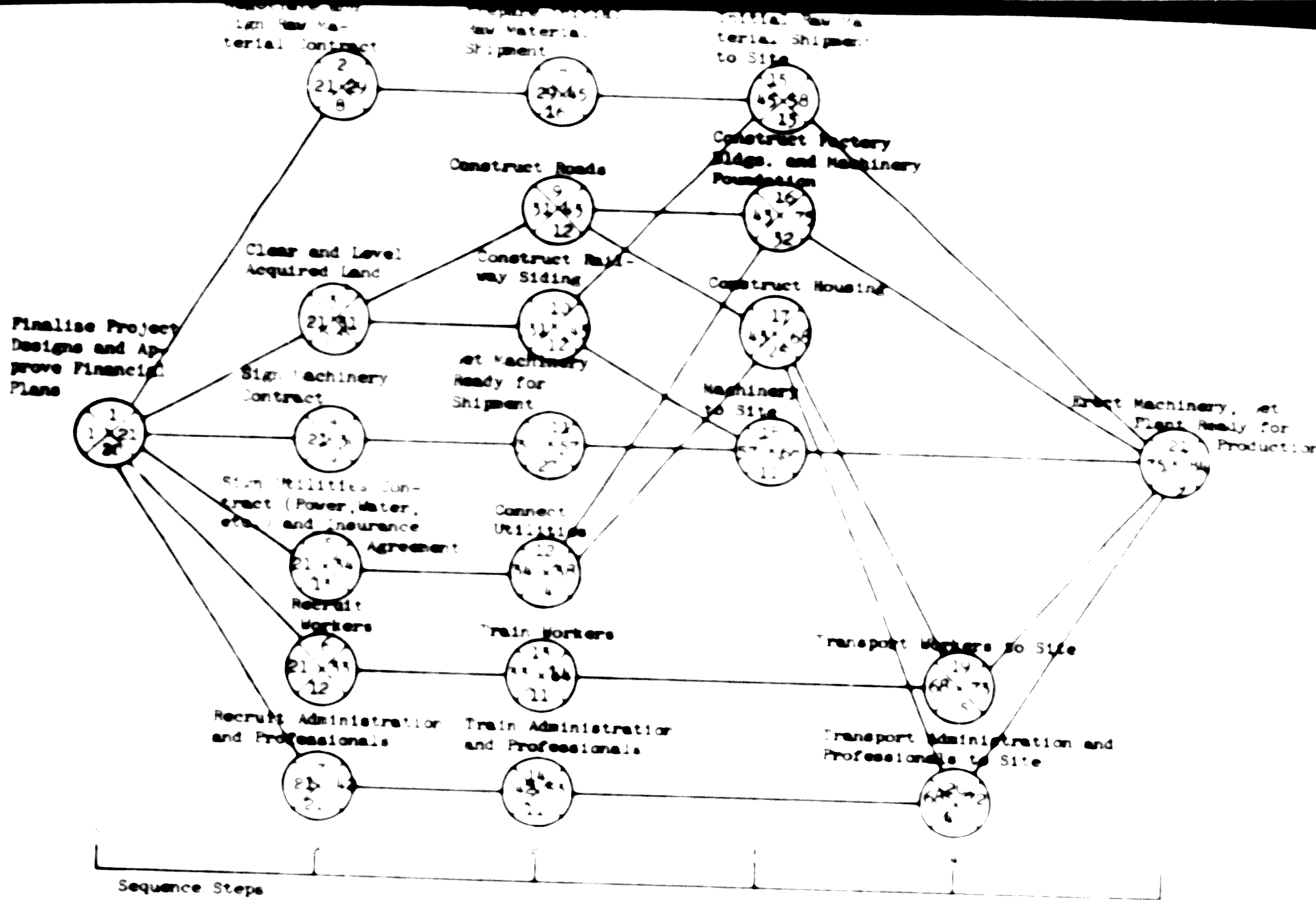


Figure - Sequence Step Precedence Diagram when non-computer methods are used, it is recommended not to update the entire network diagram. The whole diagram might be revised once and as explained earlier, at each time interval (which might again be one or two months) during project implementation a time-scale network diagram of that portion of the project network that will be undertaken in the following time interval could be revised and updated. Revision of diagrams plotted to time-scales may be restricted to cases where some activity sequential relationships change, and some activity breakdown is modified.

51. In plotting the project network diagram as shown in Figure 17, a time-scale is drawn. Then, at each activity early (or scheduled) start time, a vertical line can be drawn on the time-scale on which the node representing this activity can be located. If a sequence line has a lag time (see Chapter 3) it can be shown on the sequence line by a corresponding zigzag portion.

52. One of the advantages of constructing a network diagram to a time-scale is that if a number of vertical lines are drawn on the diagram at various time intervals, one may determine at any point in time, what activities should have been performed, what activities are in process and what activities are to be performed next. Since the drawing of vertical lines on the diagram can show what activities are concurrently being undertaken, time periods with excessive demand on one or more key resources can be pointed out. Even before applying more effective resource allocation techniques this could suggest some quick improvements in resource scheduling as for example by shifting certain activities along the diagram so that more leveling of resource utilization can be achieved.

53. It should be emphasized that this phase of networking procedures concerning the construction of the network diagram is of prime importance for programming and control of project implementation and is the basis for applying the advanced networking procedures for time-cost trade-offs and resource allocation.

Developing Project Scheduling Data

54. Once a realistic network diagram has been constructed for the project, the next step is to introduce time data by making time estimates of the dura-

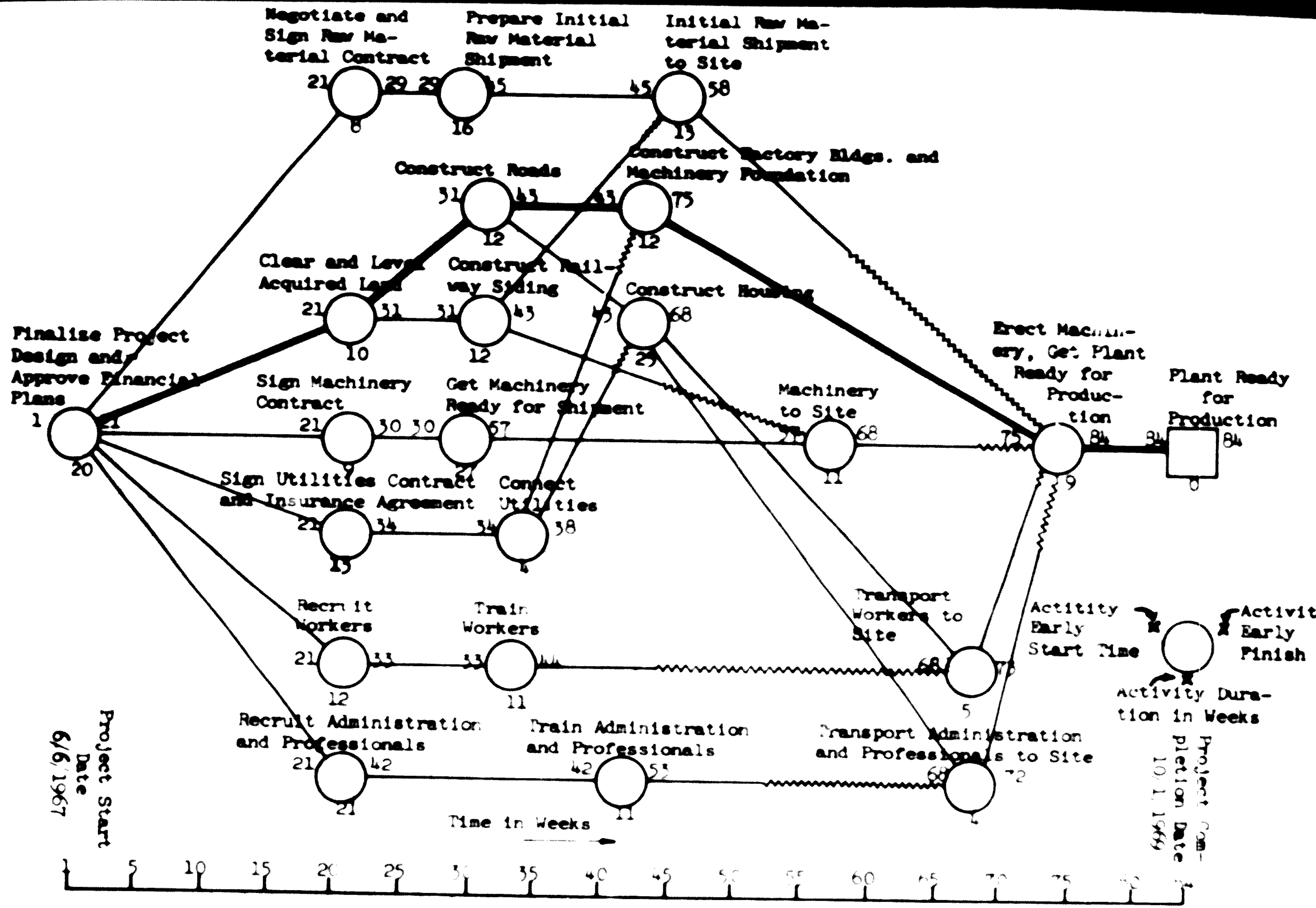


Figure 17 - Time Scaled Precedence Diagram

tion of each activity. Activity duration is the time required to accomplish an activity. These estimates may be made in any appropriate units, such as months, weeks, days, hours, etc. Any time unit can be used so long as the same unit is used for all activities. When days are used there must be a decision whether these are calendar or working days. The latter, based on the number of days in which work is actually performed, is more common. It requires the conversion of durations that are generally quoted in calendar days, such as delivery times, concrete curing periods, etc., to a corresponding number of working days. In general, due to non-working days, weekends, holidays, climatic interference such as storms, rains, etc. time could be lost during which work could not be undertaken and hence the elapsed calendar days are more than the number of actual working days. Construction work is an example where rain seasons and storms could effect delays and interruption of work. For illustration, if a construction activity is scheduled in a rainy season where its performance may be interrupted by 4 weeks and its duration time has originally been estimated at 12 weeks without weather consideration, it should be adjusted to 16 weeks (12+4). However, other activities such as delivery of machines and equipment, is not affected by these factors.

55. Activity durations are the basis of effective scheduling; all scheduling calculations depend on them. Therefore, estimating activity durations should be based on thorough understanding and knowledge of the work to be performed and the techniques that may be used, because in network diagramming good results are obtained only through the exercise of competent judgment.

56. During the course of project implementation the need may arise to change activity durations. Changing the technique or amount of resources used to carry out an activity always changes activity duration, as for instance when more resources are allocated to an activity or, on undertaking an activity when the estimated duration is found to be inaccurate due to inadequate understanding of the work involved and other influencing factors at the time the estimation was made. All these require re-estimation of activity durations so that they could always be kept updated.

57. If the type of work is one with which there is a reasonable amount of previous experience, single time estimates for each activity are practical and simplest. However, the approach followed by the Project Evaluation and Review Technique (PERT) historically permitted three time estimates for each activity: a most likely duration, an optimistic forecast, and a pessimistic forecast. Much of the missile programme work which PERT was originally designed to control involved research and development work and the manufacture of components never built before. The personnel involved were understandably reluctant to provide a single time estimate for the performance of their activities. Therefore, a weighing formula was developed to convert the three time estimates, that generally could be obtained, to a single, statistically equivalent time. The activity duration, then, was applied in the same manner as if a single time estimate had been made. This report will assume single time estimates, a method followed by the Critical Path Method (CPM). There may be cases involving engineering design work for example, where similar reluctance to provide such estimates is encountered. If so, a standard text on PERT will provide precise definitions for each of the multiple time estimates and will give the weighing formula.

58. After activity durations have been furnished, whether they result from single time estimates or have been calculted from weighted multiple estimates, certain routine scheduling computations can be made. [illegible] generally furnish six items of data for each activity: earliest or early "expected" start time, earliest or early "expected" finish time, latest or late "allowable" start time, latest or late "allowable" finish time, total float, and free float. The word "expected" is used here in connexion with earliest or early start and finish times, as all these activity duration times are just estimates and not the actual activity duration times. The latter can be only known on performing the activities. For economy in words, both the words "expected" and "allowable" could be deleted.

59. Rules governing the computation of the above mentioned scheduling data are based on network logic and may be summarized as follows:

a. Activity earliest start time (ES) - is equal to the latest or largest of earliest finish times of the activities preceding it;

b. Activity earliest finish time (EF) - is equal to the earliest start time of an activity plus its duration time;

c. Activity late finish time (LF) - is equal to the earliest or smallest of the latest start times of the activities following it;

d. Activity latest start time (LS) - is equal to the latest finish time of an activity less its duration time;

e. Activity total float (TF) - is the difference between the earliest finish time and the latest finish time of an activity (or the difference between the earliest start time and the latest start time of an activity). Activity total float time gives the amount of time (number of days, say) by which the finish time of an activity can exceed its early finish time without affecting the overall project duration. In other words, it is a measure of the extra time or leeway available for the performance of an activity without causing the project duration to be extended.

f. Activity free float (FF) - is the difference between the early finish time of an activity and the earliest or the smallest of the earliest start times of the activities following it. Activity free float gives the amount of time (number of days, say) by which the finish time of an activity can exceed its early finish time without affecting the early start time of any other activity. In other words, it is a measure of the extra time or leeway available for the performance of an activity without causing the delay of any other activity.

60. Based on the network diagram shown in Figure 13, and by referring to the activity durations already estimated and included again below, the scheduling computations can be made.

Activity	Duration in days
[illegible]	[illegible]
B1	[illegible]
B2	[illegible]
X	[illegible]
[illegible]	[illegible]
[illegible]	[illegible]

61. The first calculations are for the purpose of determining the earliest start and finish dates for each activity and are often referred to as the "forward pass". These calculations can be performed on a separate tabulation with reference to the network diagram for the sequential relationships, or they can be performed directly on the diagram as shown in Figure . Starting at the beginning of the project, the earliest start time for Activity A is 1, the beginning of the first day. Adding its duration of 1 day, its earliest finish time is 1+1=2, the beginning of the second day (or the end of the first day). Since the diagram indicates that Activities B1 and can commence after Activity A is completed, their earliest start dates are , the beginning of the second day. This process is continued through the network until the final activity is completed. Where an activity follows more than one preceding activity, its start time is determined by the finish date of the preceding activity which is completed the latest. This is the case with Activities C and D. For example, Activity C follows both Activities B1 and X. The earliest finish times or dates of these two activities are 6 (the beginning of the sixth day) and 4 (the beginning of the fourth day) respectively. Consequently the earliest start date of Activity C is 6. For Activity D, the preceding activities are Activities B2 and C with earliest finish dates of 12 and 8 respectively. Thus, the earliest start date of Activity D is 12, the beginning of the twelfth day. Besides providing the earliest dates at which each activity can be started and completed, the forward pass calculations also provide the project duration. It is set to be the earliest completion time of the final activity and in this case is 12 days, since the completion date of Activity D is the beginning of the 13th day (or the end of the 12th day).

62. Holding this project duration fixed, the latest finish time for the final activity is set equal to its earliest finish time. Then the "backward

[illegible]


Activity
Duration

[illegible]

[illegible]

activities [illegible] even [illegible] network [illegible] are [illegible]

[illegible] extend [illegible] the [illegible] network [illegible]

project duration [illegible]

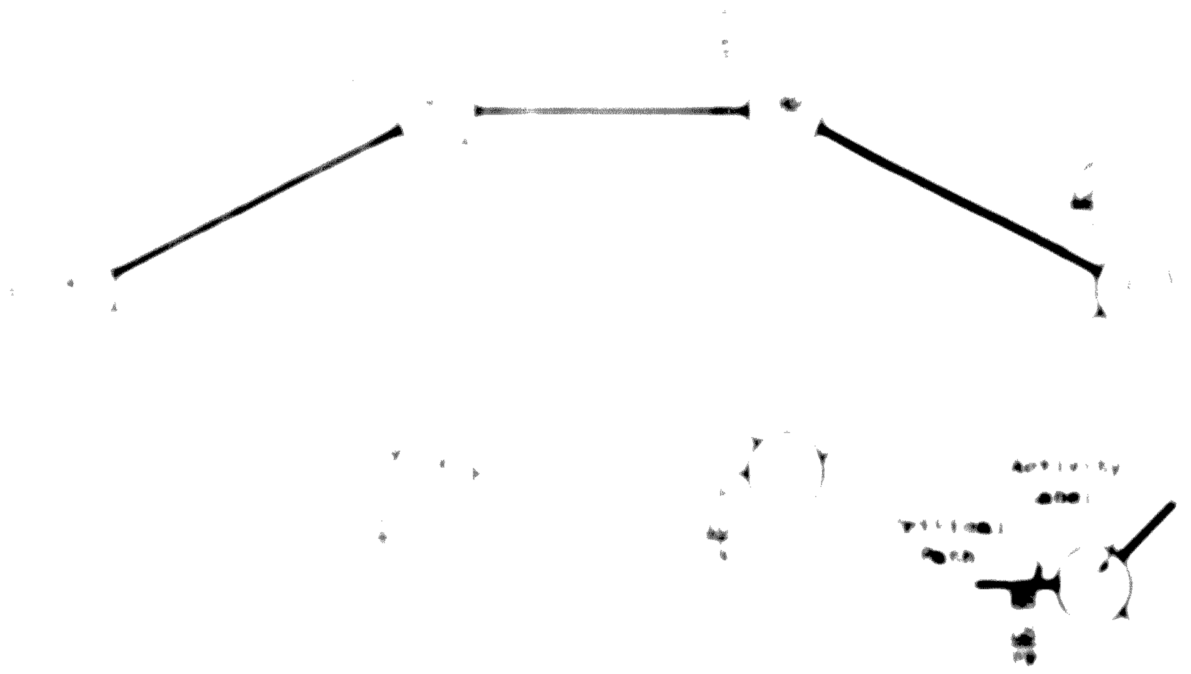
example [illegible] consists of activities [illegible]

[illegible] of the total activities are [illegible]

networks [illegible] that [illegible] a small [illegible] perhaps [illegible]

activities are critical.

4.4 Free Float calculations are shown in Figure [illegible] example [illegible] activity [illegible] has a Free Float of [illegible] days. This indicates that [illegible] completion [illegible] be delayed as much as [illegible] days without affecting any other activity [illegible]



[illegible]

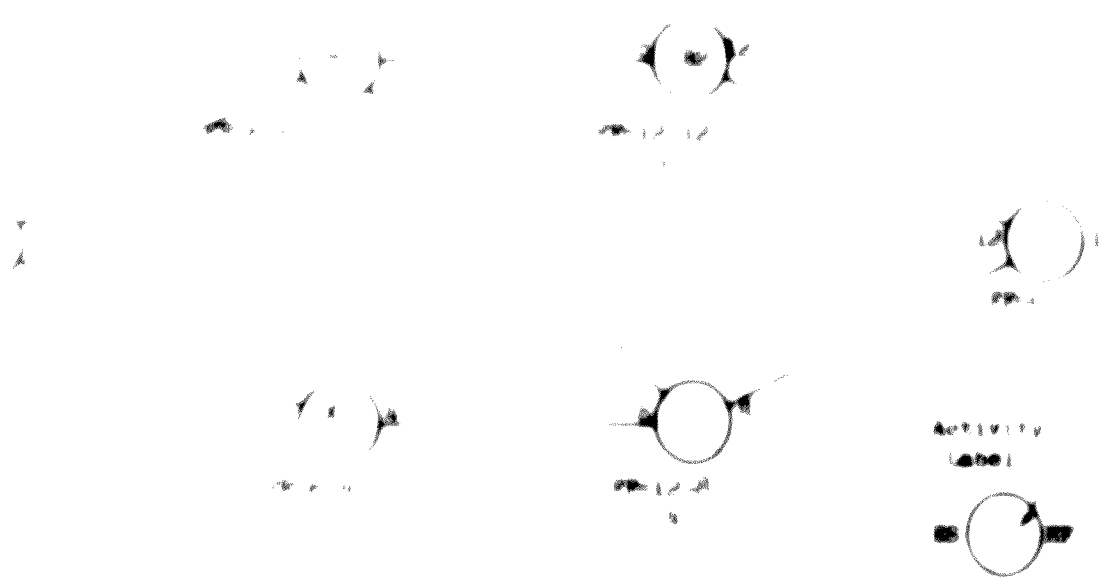


Figure [illegible] Activity Free Float Assumptions

[illegible]

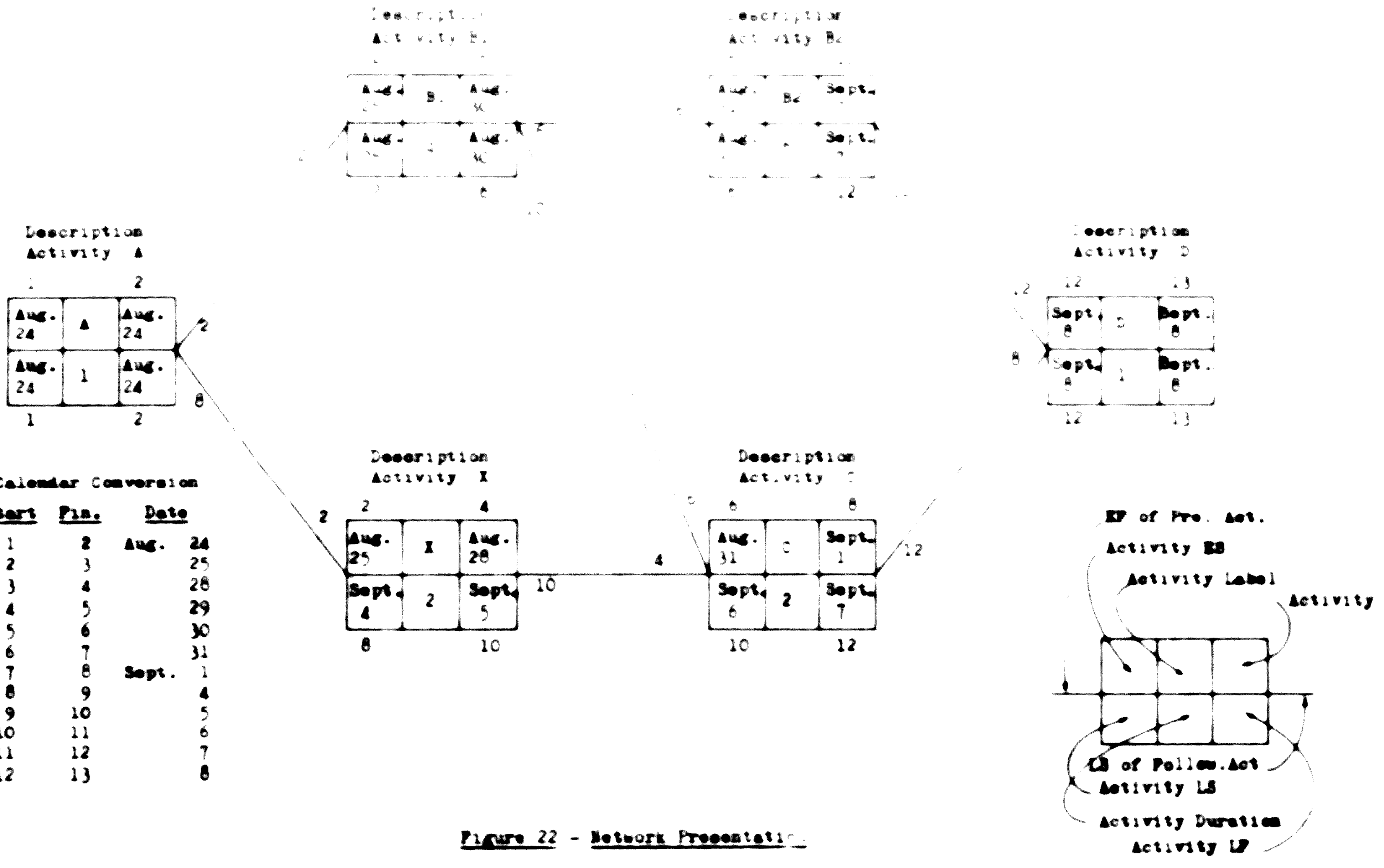
[illegible]

[illegible]

[illegible] there are [illegible] three methods [illegible]

[illegible] shown [illegible] the diagram [illegible] start time, earliest finish time [illegible]

The earliest finish time of each activity is transferred [illegible] end of its following sequence [illegible] The earliest start time of an activity is the largest of the numbers on the incoming terminals of the preceding sequence [illegible] Figure 4. [illegible] and [illegible] are written above [illegible]



Calendar Conversion

Start	Fin.	Date	
1	2	Aug.	24
2	3		25
3	4		28
4	5		29
5	6		30
6	7		31
7	8	Sept.	1
8	9		4
9	10		5
10	11		6
11	12		7
12	13		8

Figure 22 - Network Presentatio

of the sequence lines connecting Activity [illegible] to Activities [illegible] and X respectively. Six is the earliest finish time of Activity B1 whereas 4 is the earliest finish time of Activity X. Hence, the earliest start time of Activity [illegible] is 6. Earliest start and finish times of an activity are temporarily entered above their respective compartments. On the backward pass, the latest start times are transferred to the far end of the preceding sequence lines and put below the lines. The latest finish time of an activity is equal to the smallest of the numbers on the adjacent terminals of the following sequence lines. Again in Figure 22, the numbers 6 and 10 are put below the following sequence lines connecting Activity B1 to Activities B2 and C. Six and ten are the latest start times of Activities B2 and C respectively, therefore, the latest finish time of Activity B1 is 6. Latest start and finish times are temporarily entered below their respective compartments. Finally, using a conversion table shown on the diagram, undated start and finish times are converted to calendar dates and entered into the appropriate node compartments.[6] This single diagram then shows both the project implementation plan and the scheduling data. The total float or free float of any activity could be readily obtained by subtracting the appropriate undated entries from one another. An additional possibility is to plot the diagram to a time-scale in order that the activities will be located with respect to their time relationships as well as their logic relationships. This can be effectively accomplished by plotting each activity node at a position corresponding to its earliest start time with respect to a horizontal time-scale as previously mentioned.

---

[6] Each calendar date indicates the beginning of the corresponding undated time date in case of activity start time and the beginning of the next undated time date in case of activity finish date. Also, calendar dates allow for weekends. In this example five working days per week have been assumed.

[illegible] presentation is shown in Figure [illegible]. Here all [illegible] information is shown in bar chart format. The programmer has [illegible] network [illegible], and, therefore, the bar chart breakdown is the same as that required by the network. Sequencing relationships are given [illegible] of preceding activities at the [illegible] each bar [illegible] following activities at the [illegible], free and interfering float periods are shown as extensions [illegible] shows the duration [illegible] programming and scheduling information [illegible] float data, is available from a single drawing. [illegible] important advantage that it presents this information [illegible] familiar [illegible] most personnel and, therefore, requires the least adjustment to new concepts.

68. 4. The third method is to provide the programming information and the scheduling information separately. A network diagram, in its simplest form such as shown in Figure [illegible], shows the project implementation plan and scheduling data can be presented in the form of a printed tabulation. This is the common approach when computer processing is used, but it is equally applicable with non-computer processing and listing. Figure [illegible] provides a tabulation which could be used in conjunction with Figure 15. It would be updated following variations in time data and the diagram would be updated following the less frequent variations in sequencing data. The scheduling information might also be conveyed in the form of a bar chart, omitting the sequencing information described in the second method. In either case, both implementation programming and scheduling information might be transmitted to some personnel while only the scheduling information would be required and useful to others. It is often useful to carry this a step further and only transmit certain scheduling information to certain parties. For example, many parties may participate in only certain activities and may merely require scheduling information concerning them. Moreover,

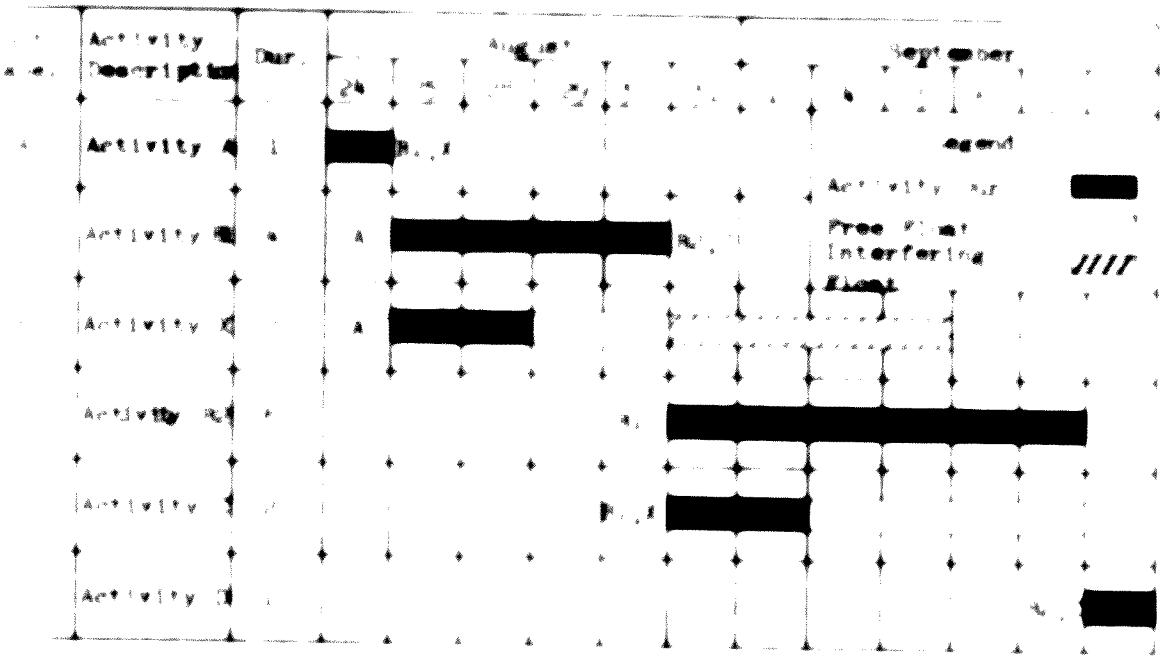


Figure [illegible] - Bar Chart Presentation

Activity Description	Dur.	Critical Activities	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
Activity A	1	•	Aug [illegible]	Aug [illegible]	Aug [illegible]	Aug [illegible]		
Activity B	[illegible]	•	Aug [illegible]	Aug [illegible]	Aug [illegible]	Aug [illegible]		
Activity C	[illegible]		Aug [illegible]	Aug [illegible]	Sept [illegible]	Sept [illegible]		
Activity D	[illegible]	•	Aug [illegible]	Sept [illegible]	Aug [illegible]	Sept [illegible]		
Activity E	[illegible]		Aug [illegible]	Sept [illegible]	Sept [illegible]	Sept [illegible]	[illegible]	[illegible]
Activity F	1	•	Sept [illegible]	Sept [illegible]	Sept [illegible]	Sept [illegible]		

Figure 24 - Schedule Tabulation

... scheduling data concerning those activities ... apply. For example, a subcontractor ... with the master programming of a project ... whether he can utilize interfering ... Therefore, while it may be helpful to provide ... that he will have as much schedu- ... not be wise to provide total ... start and finish dates. This third method of ... scheduling data provides the maximum ...

... good implementation plan and schedule for ... there must be a strong effort to make this plan ... then must be effective corrective ... effects of the deviations. The networking ... information than any other methods developed ... type undertakings.

... required information in three different areas. ... to concentrate management effort to make the ... showing the degree of seriousness of changes ... proposed. The third involves determining ... most effective after a change occurs.

... indicate where to concentrate management effort ... activities with zero total float are critical ... the project completion date is to be realised. ... having very little total float time, need ... since they will effect project duration after a ... Activities having a comfortable margin of free ... are at the other extreme and require the least amount of attention. If ... resource restrictions have made it advisable to schedule activities at certain dates, regardless of the amount of float that is indicated these activities must be watched. The resource schedules should provide management with an awareness of the importance of maintaining such scheduling.

72. The networking techniques give a clear indication of the seriousness of changes that do occur or that may be proposed. If completion of a critical activity is delayed or if completion of a non-critical activity is extended beyond its latest finish date, it is known that project completion will be delayed by the same amount unless corrective action is taken. If completion of a non-critical activity is delayed within its free float range, it is known that neither project completion nor the scheduling of other activities is adversely affected (unless resource problems are created). If completion of a non-critical activity is delayed to a point within its interfering float range, a problem may or may not exist. Further investigation is required. Project completion is not directly changed, but other activities must be postponed. Since these activities are delayed within their float periods, the effect is often unimportant. However, in certain cases such postponements may be just as serious as delays of critical activities. For example, if an activity to be performed by a subcontractor is not permitted to commence on a scheduled date, additional delays may result due to previous commitments of the subcontractor at the time that the activity can commence. The total effect may be a considerably longer delay extending well beyond float ranges and causing an increase in project duration. Since the networking techniques provide the means for determining which other activities are affected when one is delayed within its interfering float range, such effects as that just described can be anticipated and investigated in advance by management. Also, the effects on resource schedules may be forecasted if such schedules have been developed.

73. Finally, the networking techniques provide a means for determining the appropriate corrective action to take. They allow the scheduling of all activities to be properly updated. If project duration has been extended and it is desired to reestablish the previous completion date, the current critical activities have been determined. Therefore, project management knows which activities will shorten project duration if they are expedited. If cost data is

available, the time-cost trade-off technique will indicate the most economical means to accomplish this expediting. If project management wishes to consider new implementation programming approaches, the network can be revised and the corresponding schedule developed. If resource data is available, resource allocation and leveling procedures may be used to achieve a new schedule that again satisfies resource restraints.

CHAPTER 3 - NETWORK MECHANICS ... UPDATING AND DEVEL...

A. Introduction

74. The representation of a project imple... cating the component activities and their s... fundamental concept on which all other netw... these techniques, such as schedule updating... dynamic ones. They involve making changes ... effects of these changes on the remainder o... of basic networking mechanics will furnish ... for these dynamic applications. This is th... chapter.

75. One of the limitations on manual appl... ble ... h to successfully working with la... overall, or master network into smaller one... order that the important relationships and ... are properly represented by the subnetworks... work mechanics will be helpful in accomplis...

B. Lag Values

76. In the precedence diagram the sequenc... quential relationships between the correspo... line connects one preceding activity to one... concept that has been proposed[1/] is the asso... a "lag" value with each sequence line. In ... may commence as soon as the preceding activ...

1/ John W. Fondahl, "A Non-Computer Appr... for the Construction Industry", Techn... Civil Engineering, Stanford Universit...

S WITH APPLICATIONS TO
ELOPMENT OF SUBNETWORKS

lementation plan by a network, indi-
sequential relationships, is a
tworking techniques depend. Some of
ng and time-cost trade-offs, are
s in the data and determining the
of the network. An understanding
h a basis for developing procedures
the principal objective of this

plications is network size. A possi-
larger networks is to subdivide the
nes. This must be done with care in
d functioning of the master network
ks. Again, an understanding of net-
ishing this.

nce lines between nodes show se-
ponding activities. Each sequence
ne following activity. A useful
sociation of a time quantity called
n many cases the following activity
ivity is completed. In these cases

proach to the Critical Path Method
hnical Report No. 9, Department of
ity, 1961.
-

the lag value is zero. In the remaining cases the commencement of the following activity is delayed for a period after the completion of the preceding activity. The usual reason for this delay is that the commencement of the following activity is determined by some other preceding activity which has a later completion date. Sometimes the delay may be one that has been intentionally scheduled to satisfy some restraint, e.g. a wait for resource availability. In these cases there is a positive lag value. Its magnitude is determined by subtracting the earliest finish time or scheduled finish time, whichever is later, of the preceding activity from the earliest start time or scheduled start time, whichever is later, of the following activity. A lag value cannot be negative since according to the rules of network diagramming a following activity cannot commence until all of its preceding activities are completed.

B.1 Transmission of changes

77. The lag value indicates an important characteristic of the sequence line. Those sequence lines with zero lag values serve to transmit the effects of changes to the remainder of the network. Those sequence lines with positive lag values are inoperative in communicating changes. As an example, consider the simple network of Figure 25. Earliest start and finish times have been calculated for each activity and used to determine the lag values for each sequence line, as shown. Suppose that Activity B takes a day longer to perform than its estimated duration. This causes its earliest finish time to increase by one day as shown on Figure 26. This change is conveyed to all other activities ahead that are connected to Activity B by zero-lag sequence lines. The earliest start and finish times of Activities D, E, and G are all shifted one day later. Note that the scheduling of Activity F is unaffected since the sequence line connecting it to Activity B has a positive lag value. If the duration of Activity B were increased another day, the change would be transmitted in the same manner. If the duration of Activity B had been decreased, rather than increased, the same activities would have been affected and each would have been shifted to an earlier date, as shown in Figure 27.

Figure [illegible] - Sample Network

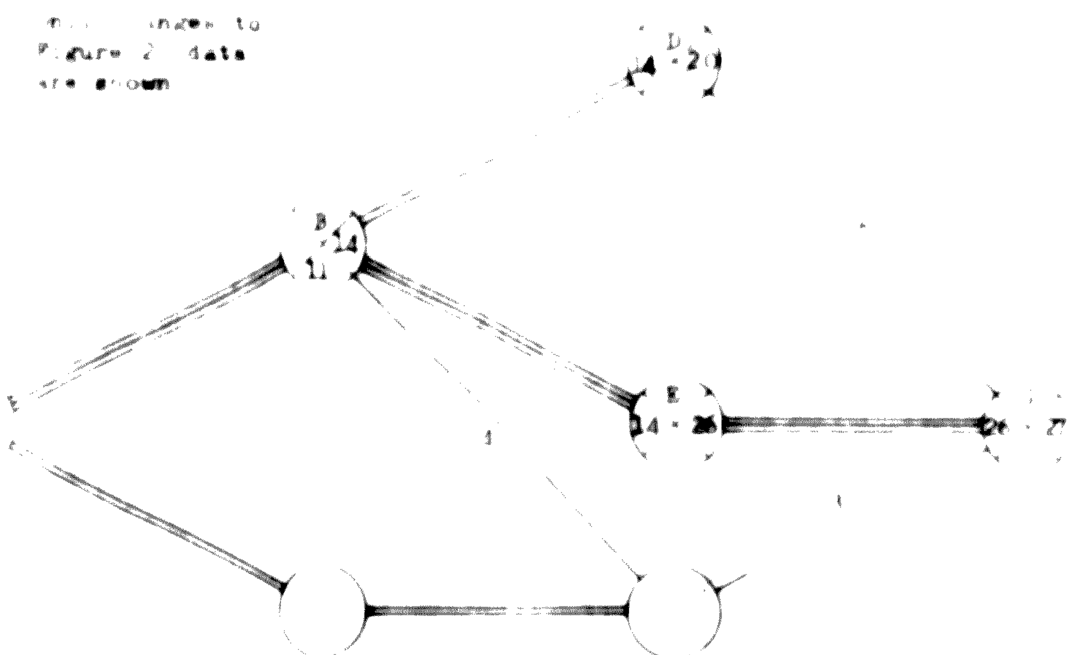


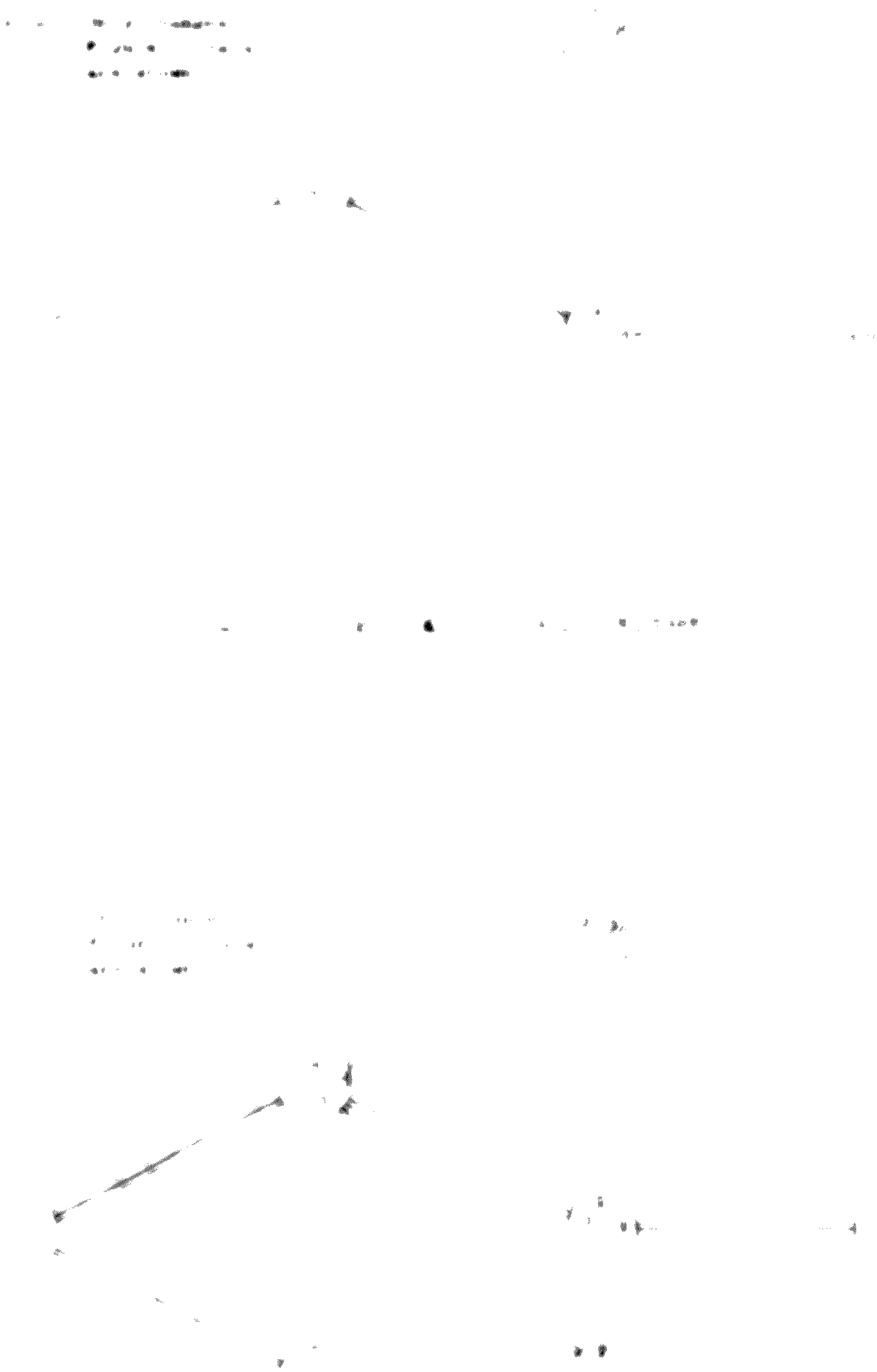
Figure 26 - Updating for Duration Increase

[illegible]

[illegible] with [illegible] increase [illegible] the same result [illegible] Sequence Line [illegible] sequence [illegible] and [illegible] the preceding activities [illegible] have [illegible] together [illegible] and [illegible] are exchanged [illegible] the start time of the [illegible] similar results [illegible] activity [illegible] except that the [illegible] rather than [illegible] the [illegible] decreases rather than increases. The former was [illegible] the latter [illegible] increased [illegible]

[illegible] Alteration [illegible]

Since the [illegible] values of sequence lines can change, they may be converted from positive to zero values or from zero to positive values. It has already been [illegible] that [illegible] are [illegible] within the network by zeroing



[illegible] therefore the [illegible] ranges.

[illegible] Figure [illegible] reaches zer[o] [illegible] the [illegible] sequence [illegible] sequence [illegible] positive and negat[ive] [illegible]

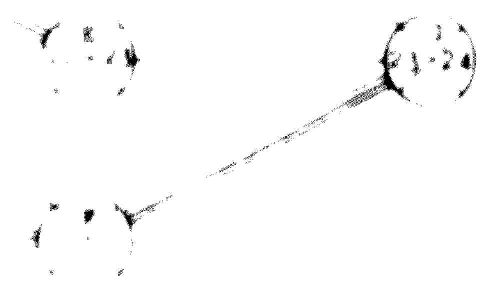
[illegible] network [illegible] a time period [illegible] the completion date of a [illegible] number of days of each change that [illegible] a sequence [illegible] zero lag [illegible] For [illegible] shown in Figure [illegible] that the [illegible] of the [illegible] is [illegible] network [illegible] for [illegible] days.

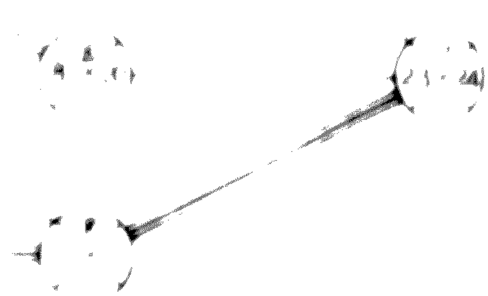
[illegible] states the number of time units [illegible] a speci[fic] [illegible] the network [illegible] takes place [illegible] the [illegible] necessary to [illegible] for the [illegible] of the previous step. At the [illegible] sequence [illegible] a zero-lag [illegible] order that subsequent [illegible]

[illegible]

[illegible] sequence line [illegible] However [illegible] has already [illegible] of the network [illegible] stay fixed. Therefore the zero-lag line [illegible] to that following activity must [illegible] converted [illegible] to transmit the effects of the change. [illegible] that sequence line [illegible] of Figure [illegible] Activity [illegible] becomes less than [illegible] days. [illegible] shows the updated network when the duration of Activity [illegible] is reduced [illegible] and [illegible] have become non-critical [illegible] one that is different than [illegible] Figure [illegible]

[illegible] It will be noted that the definition of the network [illegible] is based on a positive-lag value becoming a zero-lag value [illegible] clude the case where a zero-lag value is converted [illegible] cases result in a modification of the network that causes [illegible] ferently than before. However, the second case is automatically taken care of at the beginning or end of an updating computational cycle. The reason that this is true may be understood by analyzing the two ways in which a zero-lag line can be converted to a positive-lag line [illegible] following activity to remain fixed while the preceding activity shifts to an earlier date. This only occurs when there is at least one other zero-lag line merging into the following activity, and it results in the apparent





negative lag situation described in the previous paragraph. This situation is recognized at the very start of a computational cycle and is immediately remedied as already described. The second way for a zero-lag line to become positive is for the following activity to move to a later date while the preceding activity remains fixed. While the deletion of a zero-lag line should be taken into account for any subsequent updating cycles, it is not necessary to interrupt the cycle once commenced. The nature of the case indicates that affected activities are shifted to later dates and that the change involved does not affect the preceding activity in the sequence line under consideration. Therefore, whether the line is zero-lag or positive-lag has no effect on the transmission of changes in the updating cycle in process.

Updating of Essential Data

Frequent updating of the schedules produced by networking techniques is very important in obtaining the potential benefits from these procedures. With computer processing the simplest approach to updating is to change the items of data affected and then recompute the entire set of data for the whole network. It has generally been assumed that this is the simplest approach for manual processing also. This has placed practical limitations both on the frequency of updating and on the size of network that can be handled by non-computer methods.

The network mechanics that have been described in this chapter can be applied to accomplish updating by a different procedure. Only the activities affected by a change need be updated. If, in addition, greater selectivity is exercised in determining the data for each activity that must be updated, manual procedures will have a greater range of application. In computer processing a requirement for twice as much data for each activity may involve a negligible increase in processing time. In manual processing, twice as much data will probably require twice as much time.

In the initial determination of the project schedule it is desirable to calculate a full set of data. This set would include earliest start and finish times, latest start and finish times, and total and free floats of [illegible] the progress of the work. It is usually desirable to repeat these calculations to produce a completely updated set of data. However, the most valuable updating is not that which is performed periodically, but rather that which is performed immediately whenever changes occur. This more frequent updating can be very effective without involving a complete updating of all data. The most important item of scheduling data for each activity is its earliest start time, or its scheduled start time if for some reason this is later. If a record is continually maintained of the earliest or scheduled start times and of the current duration estimates for each activity, and if the critical activities can always be correctly identified, this is the essential information that is the most helpful. It permits keeping those who are to perform in the future aware of changes in their scheduled start times, and it allows project management to always concentrate their attention on the proper activities. For the application of the procedures to be described it is also necessary that a current record of lag values be maintained for each sequence line in the network. To make manual processing even more practical, updating may be limited to those activities in the immediate future, e.g. the next [illegible] days, rather than over the entire project duration. The use of subnetworks to be described later in this chapter will make this possible. So while the entire network should be updated periodically, this is not essential on a daily basis.

In summary, the essential data requirements to be maintained current on a continuous basis are:

a. activity duration estimates
b. earliest or scheduled start times
c. lag values of sequence lines

However, if a subnetwork approach is adopted, this data need be maintained only for that portion of the project representing a limited time period ahead. All remaining data can be obtained if desired. Earliest or scheduled finish times are equal to earliest or scheduled start times plus durations. Free float can be determined from the lag values since it is equal to the minimum lag value of those sequence lines leaving the activity. Of course, if only one sequence line bursts from the activity, activity free float will be equal to the lag value of the sequence line [illegible] by a procedure to be described in a later section of this chapter that does not necessitate the computation of these late [illegible] late start and finish times, can be determined from the corresponding early start and finish times plus total float. Critical activities can be identified by observing or noting those activities connected to the final project activity by zero-lag sequence line chains.

Methods for Updating

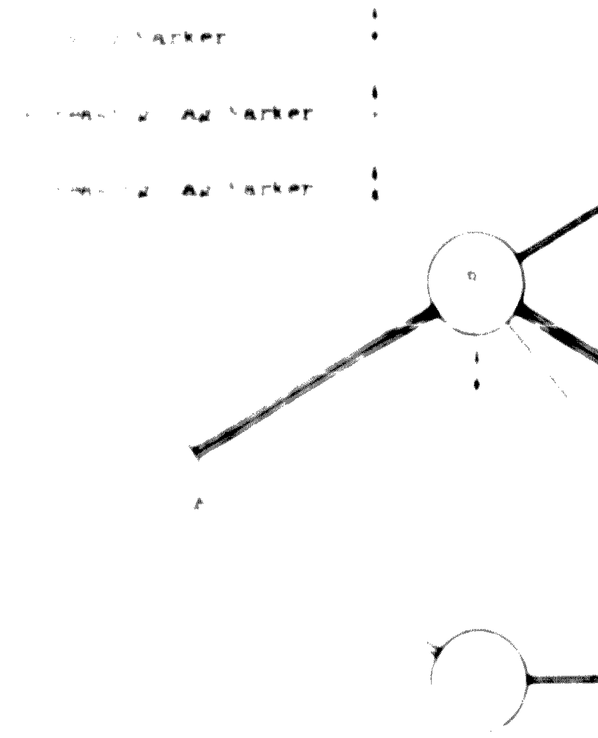
Methods for manual updating of scheduling data may be developed using the principles of network mechanics that have been presented. A change in one activity is conveyed to other activities by zero-lag sequence lines. By identifying these zero-lag lines on the diagram, the activities affected by a given change can be quickly determined. It is essential in applying this procedure that the lag values be kept updated at all times. When the affected activities have been identified, the sequence lines that connect these activities with unaffected activities can also be identified. These lines are the ones with changing lag values. Depending on whether scheduling data is being shifted to earlier or later dates and also depending on whether the preceding or following activity of the sequence line is the one affected, it is possible to determine whether these lag values are decreasing or increasing. This permits the lag values to be updated, addition or deletion of zero-lag sequence lines to be recognised and made, and network interaction limits to be determined.

Changes requiring updating involve changes in activity duration, changes in activity scheduling, deletion of existing activities, addition of new activities, or some combination of these. If the procedure for updating for duration change is understood, the procedures for the other types of changes will follow easily since they involve only slight modifications. Therefore this basic case will be described in some detail. It is assumed that a network diagram exists and that the zero-lag sequence lines are distinguished [illegible] is also assigned [illegible] start and maintained. One of these [illegible] earliest or scheduled start date [illegible] the current lag value for each se- [illegible] and consists of a documentation [illegible] nature of the change, the reason [illegible] the change and a listing of [illegible]

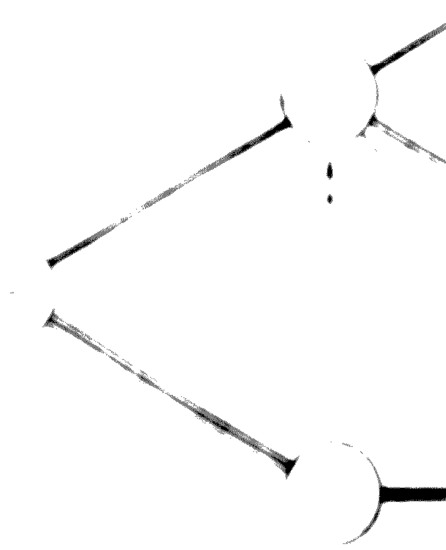
[illegible] marked on the diagram [illegible] of the diagram or sheet metal backing and to use [illegible] for marking activities. Then, proceeding in [illegible] activities following the changed [illegible] lag sequence lines are marked in a simi[illegible] markers [illegible] the activities whose start dates need to be [illegible] except [illegible] changed duration whose start time remains [illegible] decreasing lag values are marked [illegible] For example, the change [illegible] sequence lines extending from [illegible] have decreasing lags. The next [illegible] increasing lags using magnets with [illegible] the case of an increased duration [illegible] from earlier unmarked acti[illegible] the activity whose duration is changed) [illegible] network interaction limit is determined [illegible] current lag values of all sequence lines [illegible] least of these values. Finally, the [illegible] change in activity duration is less than the NIL, the start times of all affected activities and the lag values of all affected sequence lines are changed by an amount equal to the duration change. As changes are entered in the records, corresponding magnets may be removed from the diagram. When all have been removed the updating is completed. For the case where the change in duration is greater than the NIL, the affected activities and sequence lines are updated by an amount equal to the NIL. The new zero-lag sequence line relationship is entered on the diagram. Then additional cycles of updating are performed until the increment to complete the change is less than the NII for the final cycle. More than two cycles are seldom required.

[illegible] As an example involving most of the complications that will be encountered, assume that the duration of activity [illegible] of Figure 25 is increased to [illegible] days. Markers would be placed on the diagram as shown in Figure [illegible]. A check of the remaining lag lines (only [illegible] here) would indicate a NII of 2 days. The duration of Activity [illegible] and the earliest start times of Activities [illegible], E, and [illegible] would be reduced by two days on the records. The lag values of Sequence Lines [illegible] and F- would be changed to [illegible] and [illegible] respectively and the diagram altered to show the new zero-lag relationship. An attempt to perform a second cycle of updating would indicate a decreasing lag for Sequence Line F- which has already become zero. This requires recognizing that Sequence Line E- must be converted to a positive-lag line instead. Having made this alteration and once more marked the activities and sequence lines with changing values, the diagram would appear as shown in Figure [illegible]. Since there are no decreasing lags, the NII is infinite and the remaining two days of updating can be completed. The duration of Activity [illegible] and earliest start times of Activities [illegible] and E are reduced by the final two days. The lag values of Sequence Lines [illegible], [illegible], and [illegible] are increased by two days. Markers are removed as changes are recorded and the updating is completed.

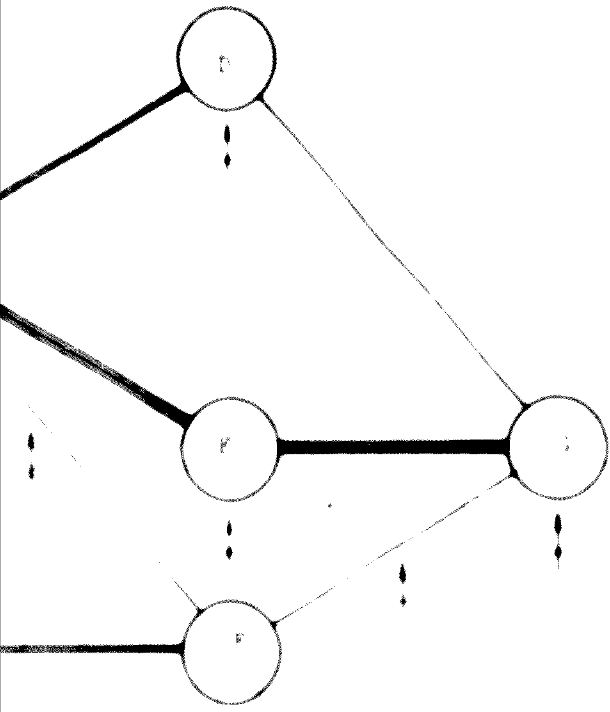
[illegible] Updating involving changes in more than one activity can be accomplished in a single operation as long as the activities requiring updating are not connected to one another by chains of zero-lag sequence lines and as long as the changes are in the same direction timewise. If the changes are different in magnitude, the change of least magnitude is accomplished concurrently for all the activities. Additional cycles of similar concurrent updating are performed for the remaining activities until the network has been updated for the activity having the change of greatest magnitude. However, if the activities whose durations are changed affect one another or if changes of both increased and decreased durations are involved, the updating for such changes must be performed in separate operations.



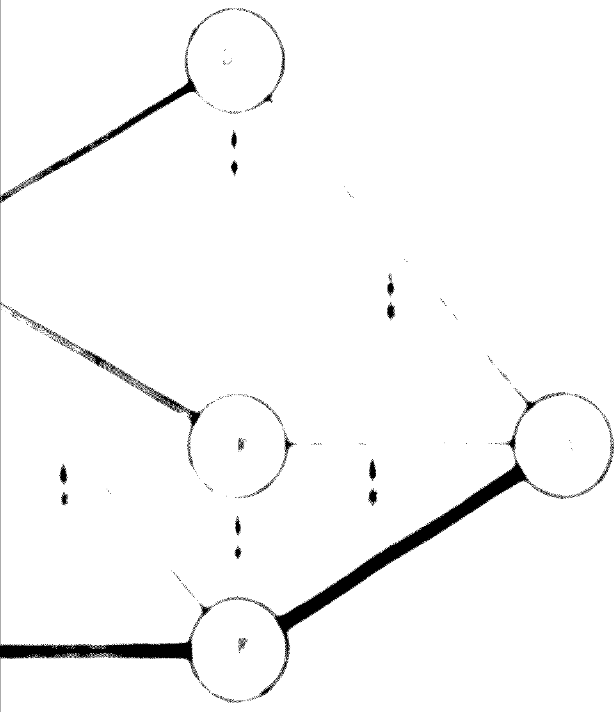
[illegible]



[illegible]



a. ...cle for Duration Change



a. ...cle for Duration Change

74. A change involving rescheduling an activity to a different date is handled in a very similar fashion to a change involving variations in duration. The only difference is that the start time of the activity being rescheduled is shifted to a different date as well as its finish time and, therefore, must also be changed. If sequence lines come into the rescheduled activity from earlier unmarked activities, they will have changing lag values in this case.

75. A change involving the deletion of an activity from the network may be accomplished by reducing the duration of that activity either until it reaches zero or until the activity commences to develop free float, whichever occurs first. The activity will begin to have free float when all sequence lines departing from it to later activities have been converted to positive-lag lines. When either of these conditions have occurred, the activity can be deleted from the network. It is necessary to examine the sequential relationships between the preceding and following activities of the deleted activity to determine whether new sequence lines between them are required or not.

76. A change involving the addition of an activity may be accomplished by inserting the new activity and its necessary sequence lines into the diagram, having temporarily set the duration of the new activity to zero. Then the duration of the new activity is increased from zero to its estimated value using the same procedures as for changes in activity duration.

77. Other changes in project implementation programming and scheduling can be treated as combinations of those already discussed. For example, a change in sequencing of two activities can be accomplished by the deletion of one of the activities and then by its addition in its new position.

F. Total Float Computation

78. The initial scheduling calculations for the entire network provide a complete set of data including latest start and finish dates and total floats. The latter provide a knowledge of the relative degree of criticality of all non-critical activities, and this is helpful in intelligently controlling the work. The updating procedures just described do not update this data. For

[illegible] is not a serious disadvantage. As long as critical activities are identified at all times and as long as the complete set of data is updated at periodic intervals to reestablish the awareness of near-critical activities, it is not essential to continually calculate current data on latest start and finish times and total float. However, there are occasions when, in making a decision on a rescheduling problem, where it is desirable to know the current value of total float of an activity. This information can be obtained for any given activity by the updating procedure already described for increasing the duration of an activity. The activity whose total float is to be determined is temporarily increased in duration until a zero-slack path is formed between it and the final project event. In other words, the number of days of delay to cause it to become critical are determined. This is its total float. This procedure has the advantage of also clearly showing the other activities that will be affected by the use of this float time. This information is generally required in reaching a sound decision regarding the proposed use of total float time, and it is not provided by the customary tabulations of updated values of total floats of all activities.

[illegible] with the capability of determining total floats comes the ability to obtain updated latest start and finish dates by simple addition of data already available. Therefore, the procedures discussed in the foregoing paragraphs allow updating of all data that is needed without a complete recomputation of data for the entire network.

Subnetworks

[illegible] another important procedure that might be classified as an application of network mechanics is the development and use of subnetworks. There are different approaches to the formation of subnetworks. The simplest case is the isolation of a portion of the larger network where that portion is only tied to the remainder at two points. The subnetwork can be removed for separate analysis and can be replaced in the main network by a single equivalent activity of a duration determined by the critical path of the subnetwork.

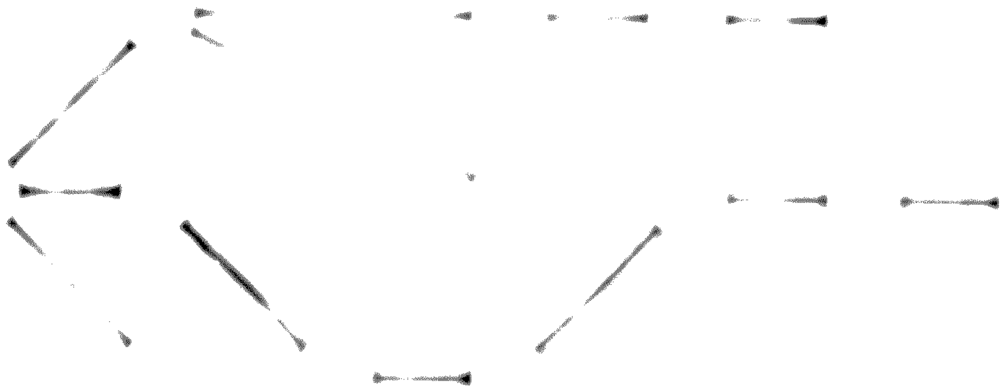
Unfortunately, this has [illegible] frequently [illegible] having a similar result [illegible] the master [illegible] is expanded into a network of [illegible] for detailed [illegible]

[illegible] There is another approach called the [illegible] method. [illegible] this approach the master network [illegible] subnetworks [illegible] periods. Ordinarily only the subnetwork for the current time period [illegible] month period) would be developed. As work progresses [illegible] time period the subnetwork for the next period is developed [illegible] next period the preceding subnetwork is [illegible] until the final subnetwork for the [illegible]

[illegible] This method has two important advantages. First [illegible] much smaller network which makes manual methods practical where otherwise they would not be. For example, consider a project network [illegible] vities and that has a duration of two years. If subnetworks [illegible] periods were developed, the average network might be expected [illegible] only about 25 activities. Assuming the largest network [illegible] twice as many activities as the average [illegible] Manual procedures are easily applied [illegible] network [illegible] second advantage is that attention [illegible] concentrated [illegible] work which occurs in the immediate future. [illegible] implementation [illegible] gramming and scheduling for the entire project is very important [illegible] implementation programming and scheduling for the entire project [illegible] It is a waste of time to perform detailed updating [illegible] leveling for activities that are [illegible] performed [illegible] inevitable that there will be many more changes before those activities can be commenced. However, it is certainly not a waste [illegible] tailed attention to the implementation plan and schedule for the next [illegible] days, for example. In fact, it is generally [illegible] desirable that the implementation plan for this period be expanded [illegible] further [illegible] is practical to show on the master diagram for the entire project.

[illegible]

1. Ibid.



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

76. 02. 06

2 OF 2

01098

[illegible] heet

[illegible]

values and space for updating them.

The procedure for each cycle of computation necessary to develop the project time-direct cost curve may be outlined by the following steps.

(a) Using the Selection Sheet (Figure 41), choose the activity to be expedited. It will be the first activity which has an entry in column 1 and none in column 4.

(b) Enter this activity in column 2 of the Summary Sheet (Figure 42). Determine its possible shortening from the Selection Sheet and enter this figure in column 5 of the Summary Sheet. Determine its cost slope from column 2 of the Selection Sheet and enter this figure in column 6 of the Summary Sheet.

[illegible]

[illegible] Summary Sheet [illegible]

[illegible]

[illegible] Summary Sheet [illegible] sequence lines marked [illegible]

[illegible] from [illegible] to zero (as will be the case when the [illegible] zero the amount [illegible] cycle shortening) or from zero to positive [illegible] the project network (Figure 4 [illegible]) [illegible] ordinarily [illegible] shown the new relationships for the lines involved.

([illegible]) Update the Selection Sheet (Figure [illegible]) by changing the Possible Shortening figures for the activities expedited. If an activity has reached its crash limit, make an entry in column [illegible] to indicate the cycle in which this occurred. If a new zero-lag relationship has produced a new critical path on the project network, make entries in column [illegible] of the Selection Sheet indicating these new critical activities and the cycle in which they became critical.

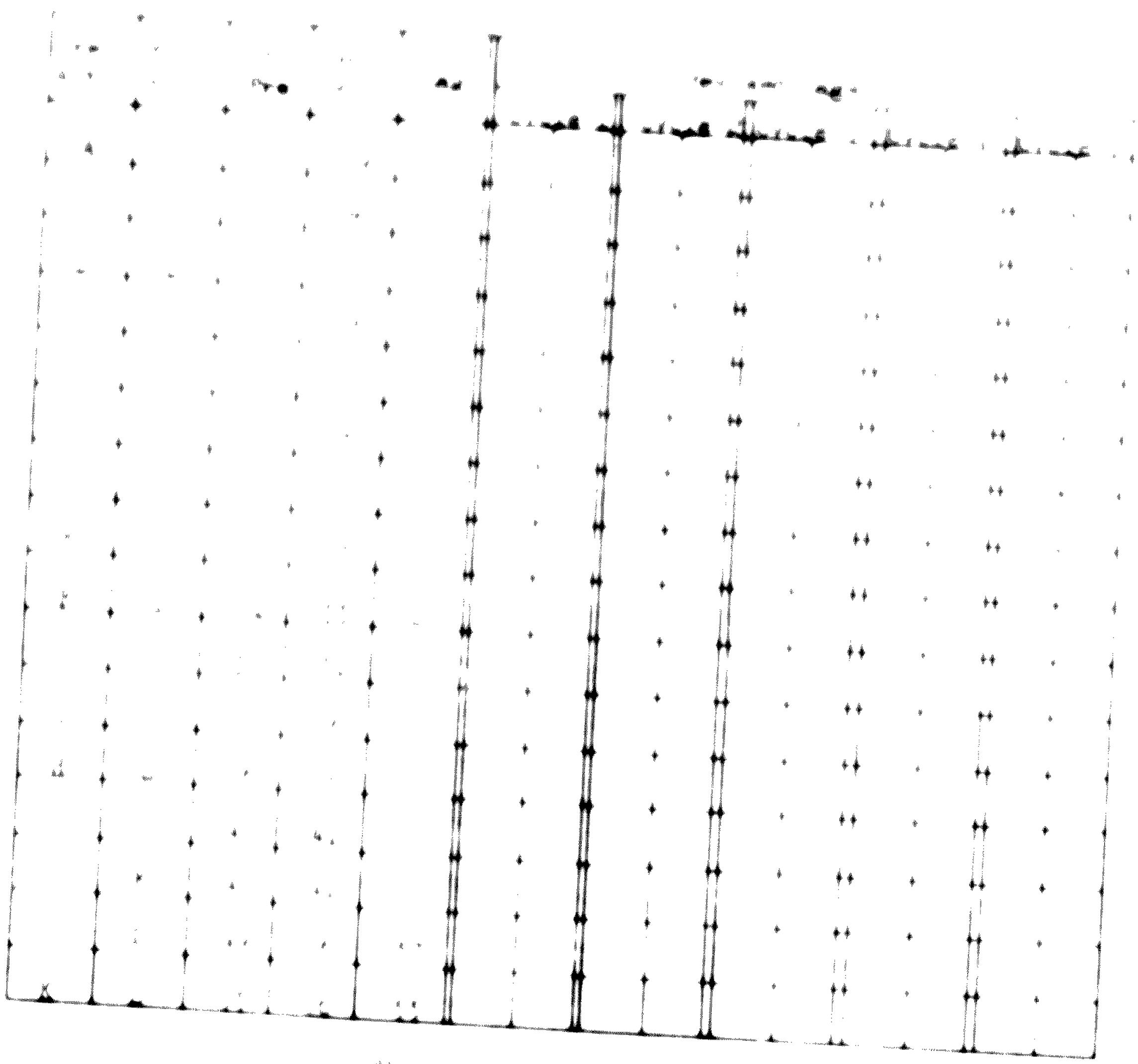


Figure [illegible] - [illegible] Sheet

[illegible] critical path is developed [illegible] activities on the critical path [illegible] to correspond to project time [illegible] advance, as an independent check, [illegible] using activity crash durations [illegible] Sheet (Figure 41).

[illegible] cost per time unit for expediting project performance, as shown [illegible] Summary Sheet, [illegible] exceeds the estimated indirect cost per time unit that is saved by earlier completion.

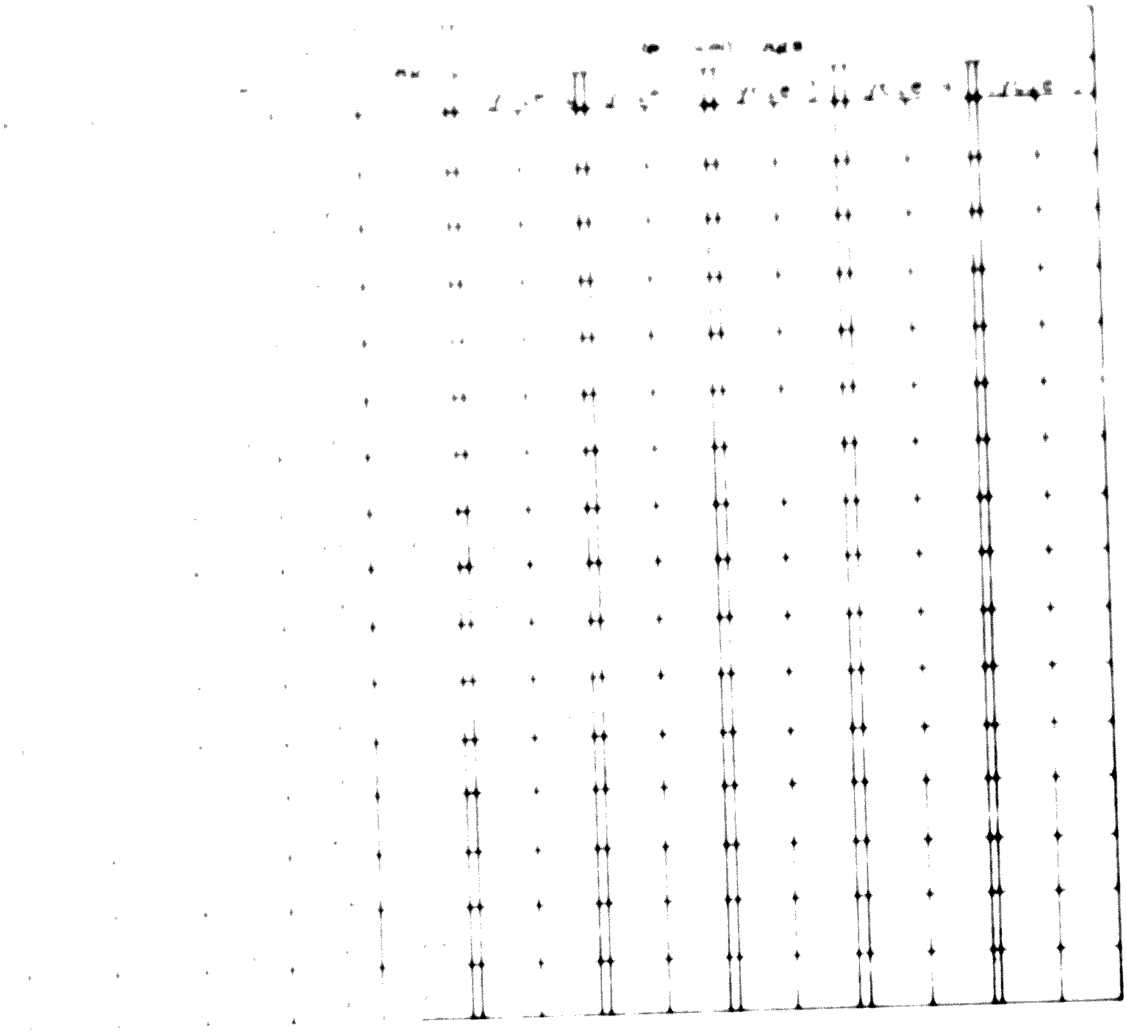
The project duration has been reduced to correspond to a specified completion date even though the cost of expediting to that date is greater than the saving in indirect costs.

104. In general, criteria (ii) would appear to be the most logical basis for termination, and would usually only require development of part of the total curve. Figures [illegible] to [illegible] illustrate this procedure step by step. Figures 54, 55, and 56, however, show also the Summary, Selection and Log Sheets respectively, at the conclusion of the development of the entire project time-direct cost curve for the example. Had the indirect cost per week been estimated at $1,500, for example, then the calculations could have been terminated following Cycle 2. Note that for project crash performance the total direct cost is $804,000. This amount corresponds to the cost for Point 3 of Figure 57 and is substantially below the $807,000 amount for all-crash performance which was developed in Figure 41. Note also that at the end of Cycle 2 a second critical path was produced. This necessitated a slightly more involved selection process for activities to be expedited in subsequent cycles. Either activities from both critical paths had to be expedited concurrently or an activity common to both paths had to be selected. In these cases it helps to list the two paths side by side, showing the portion of each path included between their common points. The most economical activity in each such sub-path can be circled. For the example at the beginning of Cycle 3, the listing would be as follows: (activity L is excluded from the list since it is common to both paths and moreover cannot be shortened).

B	A
F	(C)
(H)	C

A	[illegible]		
E	[illegible]		
B	NONE	0	
[illegible]	NONE	0	

Figure 45 - Selection Sheet (Cycle I)



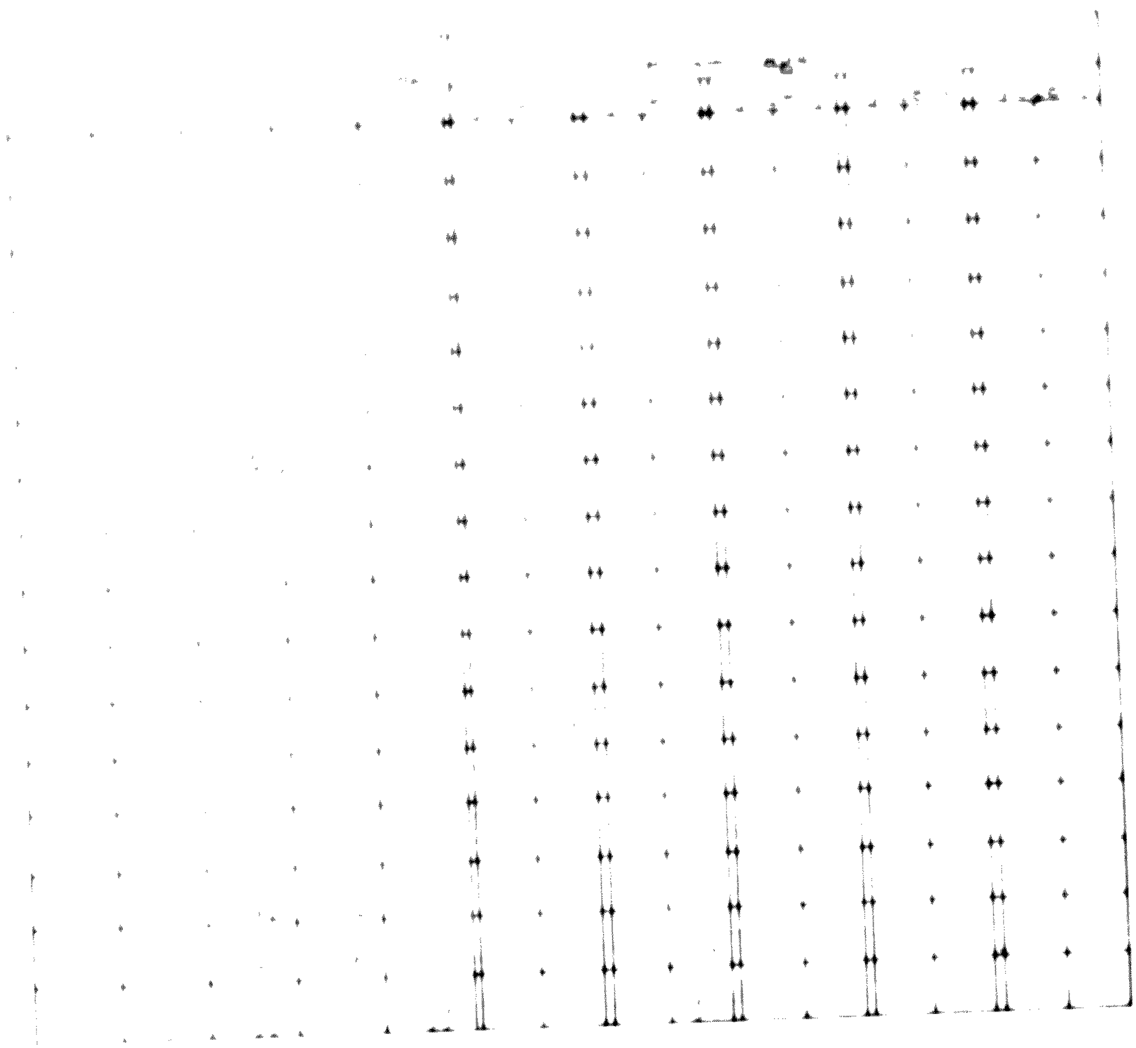
[illegible]

[illegible]

[illegible]

[illegible]	[illegible] Dollars	[illegible]	[illegible]	[illegible]						
	[illegible]	2		[illegible]						
	[illegible]			[illegible]						
	[illegible]			[illegible]						
[illegible]	1,2[illegible]			[illegible]						
[illegible]	1,[illegible]			✓						
[illegible]	2,[illegible]			4						
	[illegible]	2		[illegible]						
F	4,[illegible]			1						
A	5,[illegible]	2		[illegible]						
E	[illegible]			2						
B	NONE			[illegible]						
L	NONE			[illegible]						

Figure 49 - Selection Sheet (Cycle 2)



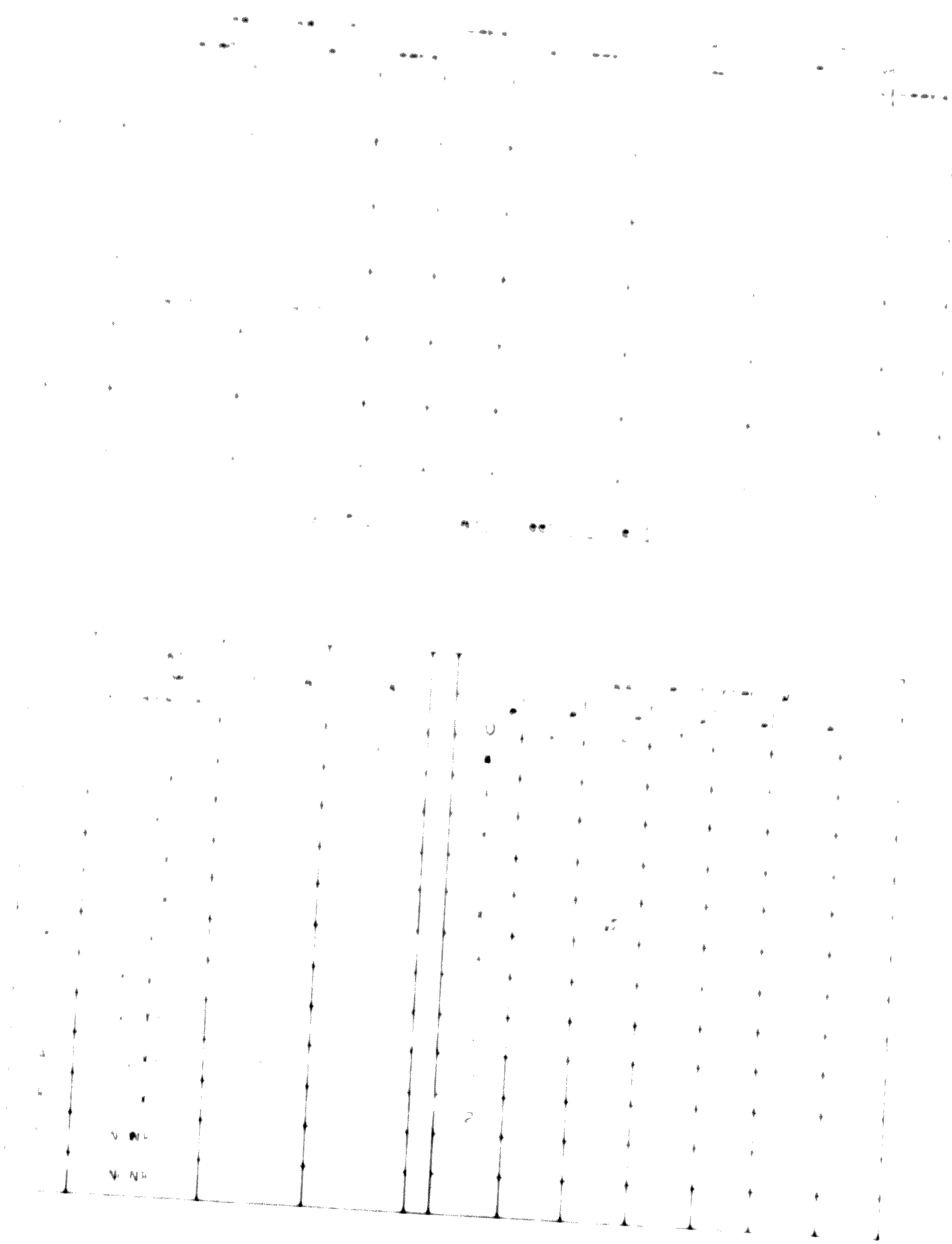


Figure 12 - Selection Sheet, [illegible]

[illegible]

[illegible] there are three or more critical paths [illegible] In its simplest form there would be three [illegible] paths A, B and [illegible] critical activity [illegible] would have been previously expedited [illegible] there would exist a combination of one activity [illegible] paths A and [illegible] and one activity [illegible] shortened concurrently [illegible] of shortening for [illegible] each one for paths A and [illegible] This would permit extending the duration of the previously shortened activity [illegible] by the time unit for each unit of shortening of the two selected activities. The combination of the [illegible] expediting the two selected activities less the saving realized by [illegible] time for the activity whose duration could be extended might be less [illegible] expedition [illegible]

[illegible] programming techniques [illegible] mathematical [illegible]

[illegible] however, [illegible] under infrequently, the

[illegible] programmer who is aware

[illegible] solution unless there

[illegible] complex interrelationships.

[illegible] manual approach proposed here.

[illegible] Trade-offs

[illegible] preceding section may be considerably

[illegible] The manual approach also allows a degree

[illegible] use of realistic data that computer procedures

[illegible] mathematical techniques do not practically permit.

[illegible] gives the possibility of obtaining a sub-

[illegible] rather rare circumstances discussed in the last

section. [illegible] will explain these claims.

[illegible] procedure discussed so far in this report and mathematical solutions [illegible] general are based on the assumption that the activities in [illegible] are independent of one another except for the sequential relationships shown by the network diagram. From a practical standpoint this is often not the case. The measures taken to expedite one activity may affect others and in turn produce secondary cost effects that should be considered. Sometimes these cost effects need to be considered in the time-cost trade-off cycle being performed and other times they should be considered in performing future cycles. In either case an analysis of the interdependencies is necessary during the intermediate steps of developing the time-cost curve. The manual, cycle-by-cycle calculation has a very real advantage because it permits a continual application of judgment and resulting modifications of data throughout the process of developing a solution.

For example, a decision to work overtime or shift work to expedite a particular activity may have an effect on concurrent activities. It may become necessary from a practical labour relations viewpoint to put other crews on the same schedule. The resulting

[illegible]

... attract[illegible] ... of the [illegible] ... after [illegible] ... specialized equip[illegible] ...

[illegible] crashing the [illegible] ... different [illegible] these two [illegible]. Yet the [illegible] ... to furnish the crash [illegible] ... which crashing is to take place. Such a demand [illegible] unrealistic [illegible] the data resulting is necessarily unrealistic. [illegible] set of data, as discussed further under the next [illegible], and [illegible] permitting modifications after each expediting cycle the procedures [illegible] this report do not demand or depend on unrealistic data.

(b) One of the major problems from a practical standpoint [illegible] applying existing time-cost trade-off procedures is that the amount of data required is excessive. At least two time and cost estimates for each activity in the project have been considered a minimum requirement. This was the case with the data presented in Figure 41 and in general, is the requirement for all computer procedures. However, a major advantage of the manual procedure is that a complete set of data is not required. The only cost slope data needed at a specific stage of the procedure is that for the activities that are critical at that point. These generally include a small proportion of the total activities. Moreover, a competent estimator can usually eliminate the majority of these critical activities from detailed consideration by a rapid examination and then concentrate on a more detailed analysis of the few remaining ones. For example, in the case of the network of Figure 40 there were initially only four critical activities, Activities B, D, H, and L. Two of these, B and L, were of such a nature that the estimator could eliminate them from consideration at a glance. Therefore, it would only be necessary to examine and develop data for two activities at the beginning of the time-cost trade-off procedure for the previous example. When later in the procedure Activities A, C and G became critical, these activities would need to be considered. However, again, the estimator would probably postpone a detailed analysis of Activities G and A since he could tell very quickly that Activity C would offer the best possibilities for expediting. The importance of requiring only a minimum amount of data for effective application of time-cost trade-offs cannot be emphasized too much. It is a tremendous advantage in implementing these potentially powerful procedures. It is an inherent advantage of the manual cycle-by-cycle method of calculation.

(c) It was mentioned earlier in this chapter that the activity time-cost curve is often of the same general shape as the project time-cost curve. This is logical since many activities can be expanded into networks of their own and be considered as projects. Such a curve was shown in Figure 39b. For each activity with this type of curve there is a period of possible expediting at a cost slope considerably less than the cost slope determined by the two terminal points of the curve. If many activities are not expedited during the application of time-cost trade-off procedures [illegible] determined for normal and crash performance appear [illegible] many favourable opportunities for project cost improvement will have been overlooked. It is very important to achieve the greatest economies that expediting decisions be based on the low cost slope ranges of the critical activities rather than on the customary normal-crash cost slopes. This is seldom practical in computer procedures since the data requirements for multi-segment activity time-cost curves would be prohibitive. As discussed under point (b), the requirement for only two time-cost points per activity is often a serious obstacle to practical implementation. However, with manual, cycle-by-cycle calculations there is no problem in taking into consideration the low-slope initial portions of activity time-cost curves. Generally a much more favourable solution will result from limited expediting of a number of activities instead of expediting a few all the way to their crash limits.

(d) Because of the almost infinite number of scheduling combinations the project time-cost coordinates may actually approach an almost smooth, continuous curve. However, this is not as true for activity curves. They are often a series of discrete points that represent a limited number of alternatives. In some cycles of the development of the project time-cost curve the amount of activity shortening will be determined by the network interaction limit. If from a practical standpoint the activity must be shortened by an amount of time greater than the NIL in order to achieve project shortening equal to the NIL, the true cost slope will be higher than that based on the assumption of a continuous curve. There may prove to be more attractive solutions based on expediting some other activity or based on the performance of that same activity by an alternative method producing an amount of shortening more nearly matching the NIL. This alternative method might not ordinarily be considered because it appears to have a steeper cost slope. To illustrate, consider the conventional two-point activity time-cost curve NC of Figure 57. The cost slope is apparently $50 per day. However, assume that the NIL for shortening this activity is 2 days. Therefore, expediting to the crash limit would only produce two days of project shortening if the new zero-lag sequence line created a new critical path that would cause this activity to become non-critical. However, the cost increase would be $400. The effective cost slope is $200 per day. There may be other activities that offer more economical solutions for two days of project shortening. Or, there may be other discrete scheduling solutions for this activity, such as that corresponding to Point A. Point A represents a method of performance which, though its cost slope is twice as great as that of Point C, provides two days of project shortening for $100 less (an effective cost slope of $150 per day).

If curves are not truly continuous, cost-slopes offer a rather poor criteria for selection of activities to be expedited. The manual, cycle-by-cycle method permits the programmer to calculate the NIL for a proposed change and to consider specific alternatives with relation to their respective NIL's. This will permit more intelligent time-cost trade-off decisions.

F. <u>Time-Cost Trade-Offs for Subnetworks</u>

129. Time-cost trade-off analysis is ideally performed using the entire project network. Although it is possible to take advantage of the fact that only limited data for selected critical activities is required for application of the manual procedure, there are practical limitations on the size of network that can be analyzed. The use of the dateline cutoff subnetwork, discussed in Chapter 3, offers an imperfect but helpful approach to larger networks.

130. This subnetwork approach is imperfect because all of the opportunities for project expediting do not occur within the subnetwork and therefore are not considered in making decisions. It is both possible and probable that a more economical method of expediting is offered by activities outside the subnetwork since the subnetwork represents a small portion of the overall

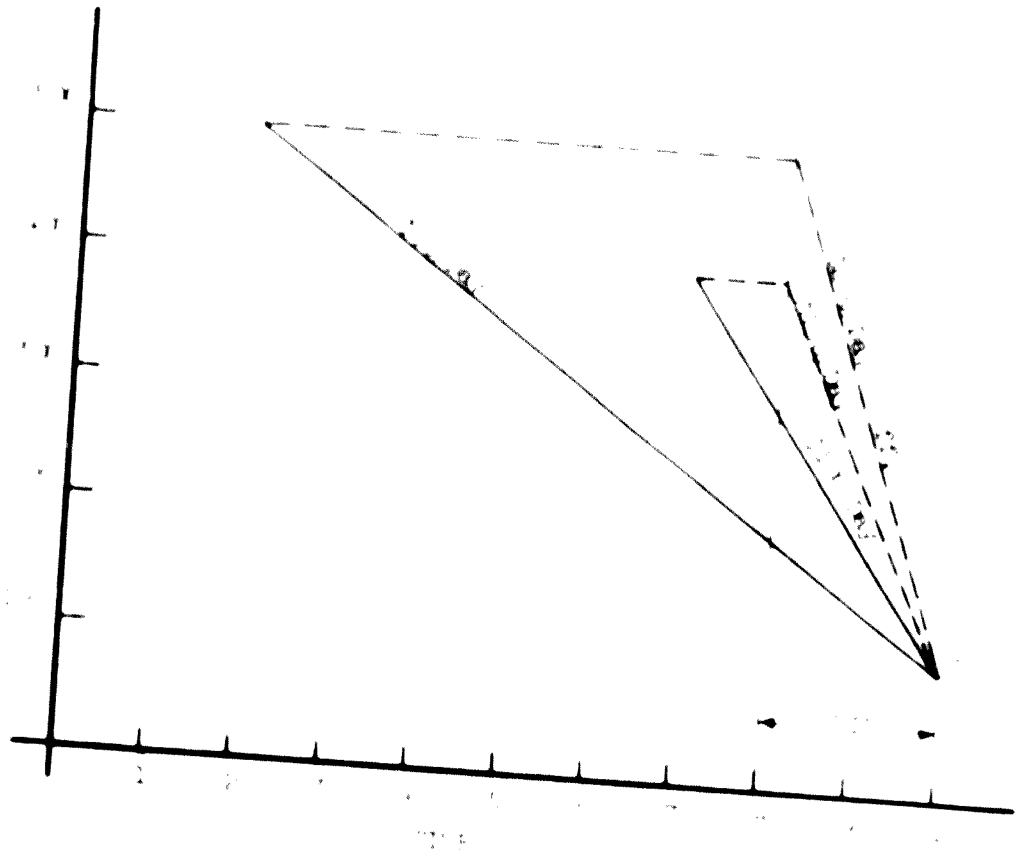


Figure [illegible] - Effect of [illegible] on Expediting project. If there are multiple critical paths, the optimum expediting solution may involve the joint expediting of an activity within the subnetwork and another in a later portion of the project. However, by confining consideration only to the work within the period of the subnetwork, these optimum solutions are not obtained.

(4) Even imperfect solutions which result in improved, more economical schedules are better than no solutions at all. By establishing a value per time unit (e.g. $ per day, per week or per month) for earlier project completion, a target is available for judging whether any proposed change will result in an improved schedule. If the change shortens the project at a lower cost than the amount saved by the earlier completion, even though it may not be the optimum improvement possible, it is advantageous The chances of achieving near-optimum solutions are greatly increased by the ability to consider the low cost-slopes of the initial segments of each critical activity time-cost curves. This increases the number of opportunities for expediting within each subnetwork and tends to reduce the differences between expediting opportunities within and without the subnetwork. It seems certain that an analysis based on a consideration of these low-slope segments but limited to activities within the current subnetwork will produce better results than an analysis based on a consideration of the entire network at one time but utilizing the assumption of two-point, single-slope activity time-cost curves.

CHAPTER [illegible] - SINGLE PROJECT RESOURCE ALLOCATION

A. Introduction

152. The execution of most project activities requires the utilization of one or more types of resources. These resources may consist of classifications of labour, such as management personnel, professionals and workers. Each category could be further classified as for example, workers could be classified into carpenters, qualified welders, or equipment operators. These resources may consist of types of equipment, such as power shovels, tractors, cranes or trucks. In some instances they may be types of materials, such as concrete, or structural iron and steel. The money required to pay for work completed might even be treated as a resource in the procedures to be discussed. In most developing countries most of these resources that are required in the performance of project activities are available only in a limited supply. In some cases these limitations are not serious, but often they impose real restrictions that need to be considered if a realistic and intelligent project schedule is to be developed.

153. So far in this report no attention has been given to the matter of resource requirements. A project implementation plan has been developed based only on satisfying physical restraints. Sequential relationships have been based on defining those activities that must be completed before others can be started. Activity scheduling has been based on these sequential relationships and routine calculations that also require estimates of activity durations. At no point has the question been asked "Are the resources required for the performance of this activity available?" and "If the required resources are not available, what action should be taken?" The answers to these questions are just as important in producing a realistic schedule as is adherence to physical sequencing restraints. It is true that certain very clearly defined resource restrictions may be taken into account while developing a project implementation plan even though these questions are not formally asked. The programmer may recognize, for example, that he has only a single pile-driving rig available and take this into account as he sequences the various activities that require the use of this equipment. But most resource problems are more subtle ones. They may involve quite a number of units of a resource

[illegible] of resource that can be scheduled or [illegible] for these activities which will not cause resource availability limits to be exceeded is frequently a difficult task. The development of procedures for these purposes is desirable.

[illegible] as in other cases of problem [illegible] complete definition of the problem is a major step towards its solution. The development of resource schedules for each important resource will clearly define the resource allocation conflicts that exist. A resource schedule is a listing of the number of units of that resource that is required during each time unit for the span of the project duration. To obtain resource schedules it is necessary for the estimator to furnish data giving the resource requirements of each activity. These requirements should be based on the method of performance that corresponds to the duration that has been estimated for the activity. It is also necessary to have developed a schedule for the timing of the performance of each activity. A schedule based on starting each activity at its earliest start date offers a logical initial schedule for purposes of problem definition. Having resource requirements of each activity and a scheduling for each activity, the development of resource schedules involves a time-unit by time-unit examination of those activities in progress, the summation of the number of units of each resource required during each time unit, and the entry of this total figure in the proper resource schedule. For purposes of a sample problem, consider the project network shown in Figure 58. Assume that there are three resources required for the performance of this project which are available in very limited supply and therefore may result in scheduling restrictions on project activities. These resources are designated as Resources X, Y and Z, and the requirements for each resource by each activity are shown in the diagram under the node symbols. Scheduling data for this network has been calculated, and the results are shown in the tabulation in Figure 59. Activity duration without consideration of resource restrictions is 34 weeks. A schedule based on starting each activity at its earliest start time is shown in Figure 60 and the resulting resource schedules for the three resources are [illegible]

summarized [illegible]

The resulting schedules are not [illegible]

ents and to [illegible] frequent [illegible]

ially true for resource [illegible]

weeks [illegible] the requirements [illegible]

1[illegible]. Two principal [illegible]

occurs when [illegible] demand for resources [illegible]

Since such a demand cannot be satisfied, [illegible]

sible for the demand will have [illegible]

requirements have to be reduced [illegible]

The principal objective [illegible]

duration is not increased [illegible]

as is absolute necessary. The second [illegible]

of the variations in resource requirements [illegible]

money, and the larger they are the more they cost. [illegible] These variations [illegible]

requirements followed by an increase represents [illegible] expense [illegible]

labour, it involves laying off men with the risk [illegible]

and later incurring the expense of [illegible]

alternative is to retain them on the payroll without [illegible]

their time. In the case of equipment, there is the [illegible]

project site and then moving it back when it is needed [illegible] the alternative

here is to absorb the ownership or rental costs while the equipment stands

idle. All of these measures tend to increase costs. The ideal resource sche-

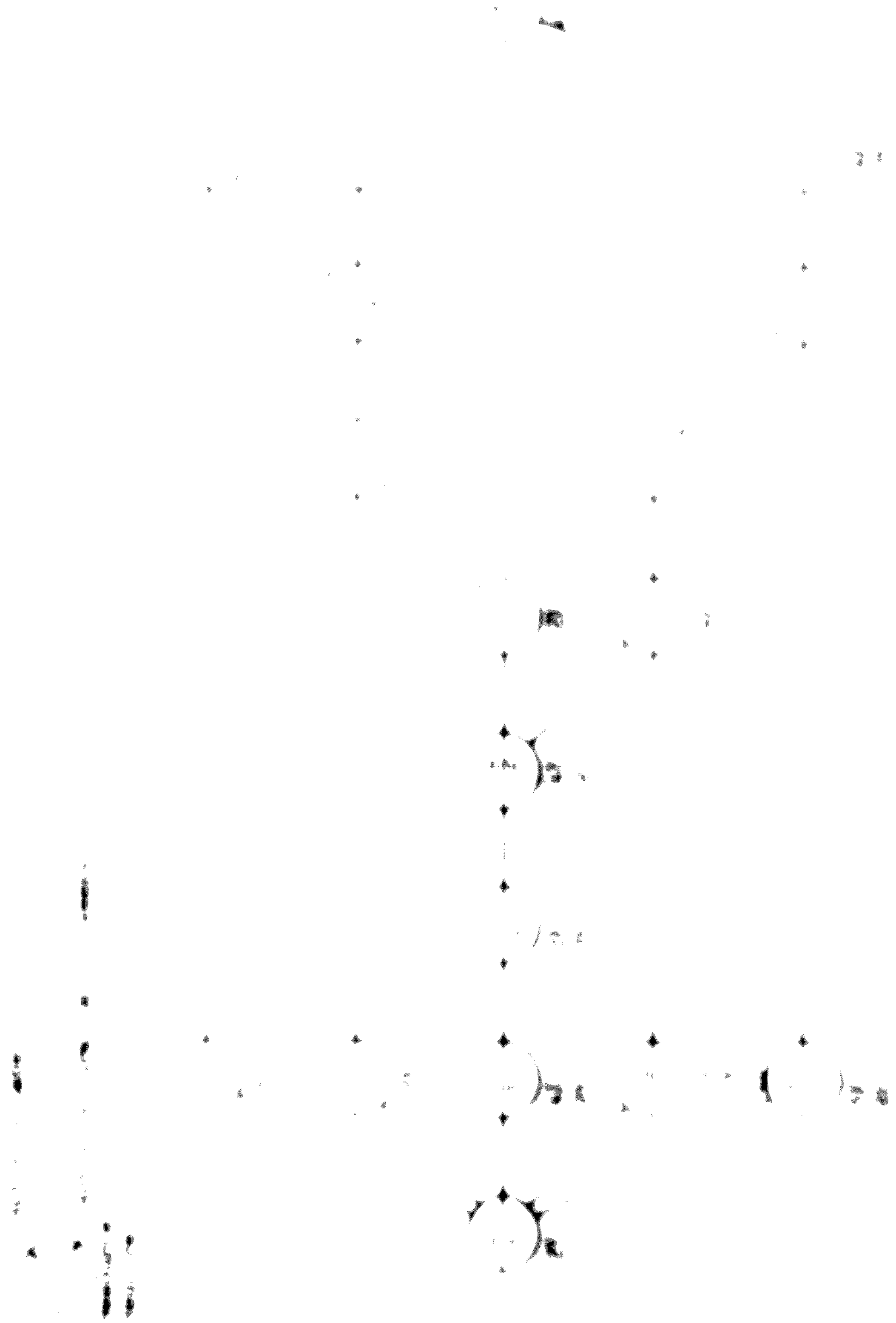
dule would be represented by a curve that built up and tapered off gradually,

was constant over its intermediate range, and never exceeded availability

limits.

B. General Approaches to Resource Allocation

136. Developing a schedule to satisfy resource restrictions can be a very complex problem. Even for small networks and for processing by large capacity computers, there are so many scheduling combinations possible that it is not practical to apply procedures that theoretically would guarantee an "optimum"



solution. There are a number of very sophisticated resource allocation programmes that require large computers and considerable running time on this equipment and which hopefully provide near-optimum solutions. Many of these programmes are proprietary ones. The problem is a complicated one because of the amount of data involved, the probability that a considerable amount of this data will have to be updated, the interactions that occur when activities are rescheduled, and the almost limitless number of rescheduling combinations that are possible. Therefore, it is generally claimed that computer approaches to this problem are essential due to the magnitude of the data and operations required.

Consider again the resource schedules developed in Figure 60 for the sample network. The development of such schedules is not a new concept nor is it dependent on networking techniques. They can be developed from bar chart schedules as long as resource requirements are provided by the estimator. Careful programmers have developed such schedules for important projects for many years. However, this has not been a common practice for the majority of projects. The reasons apparently are that a lot of data must be furnished and a considerable amount of time and effort must be expended. These are not particularly good reasons for not developing resource schedules, since the time and effort is generally profitably spent. But the fact remains that project management often chooses to proceed quite blindly with respect to resource problems due to the trouble involved in developing even a single set of resource schedules. If the initial resource schedules were developed and then if scheduling changes were proposed, each such change might require extensive modifications. If a critical activity were rescheduled or if a non-critical activity were rescheduled in its interfering float range, other activities would also have to be rescheduled. These activities may involve a number of different resources. The original change which was aimed at a specific improvement in one portion of one resource schedule may produce many changes in other portions of that resource schedule as well as changes in several other resource schedules. Some of these changes may be unfavourable and may even nullify improvements that have been previously achieved by considerable effort. In view of these problems it seems unlikely that a manual procedure can be a practical and successful method for developing project schedules that provide good solutions for meeting resource limitations. Yet this is the purpose of this chapter.

14. The mere fact that most resource programming is [illegible] accom-plished, in some cases poorly but in other cases satisfactorily, by non-computer methods provides evidence that such methods are practical. While at one extreme there are very elaborate computer oriented resource allocation programmes, these account for an infinitesimal percentage of resource programming efforts. At the other extreme there is a complete lack of advance programming, and the problems are solved as they arise. This approach might be illustrated by the situation on an ordinary construction project where on a given work day there are certain units of equipment on the site and certain men report for work in the morning. If there are insufficient men or equipment to perform all the work that is ready to be done, some work must be postponed. Choosing which items of work to postpone is often left to the craft foreman and decisions are based on judgment and intuition rather than any precise knowledge of criticality and float times for activities. If there are more men or equipment than needed, it is the foreman who decides how to keep them reasonably busy or, in the case of the men, whether to lay them off. No matter how crude this process may be, it constitutes a resource allocation technique.

15. Both extremes of practice have certain advantages and disadvantages. The computer can handle the large amount of data and computational steps that are routinely required and can apply mathematical procedures that are completely prohibitive in manual processing. On the other hand, it can only consider implementation programming and scheduling modifications that it has been programmed to perform and for which it has been supplied input data. It is impossible, in a practical sense, to program a computer to consider all alternatives that might be possible and desirable in specific situations or to furnish it with all the data that it might possibly make use of. Another disadvantage in some locations is that resource allocation programmes require very large capacity computers which are not always readily available. The disadvantages of the other extreme of practice should be apparent from the illustration in the preceding paragraph. Decisions are made at the last moment when the opportunity for favourable solutions may already have been lost; they are made without benefit of data that could be helpful, and they are often made by a level of management that may not be the best qualified to make them. An advantage may be that they do not require

computer equipment that is not available. However, regardless of the availability of computer facilities, an even greater advantage is that they permit the exercise of judgment. There are many possible ways to resolve resource conflicts. One is to reschedule activities. This is the principal, and often the only, basis for most formal procedures. However, there are other measures that can be just as effective. Activities can be replanned to change their resource requirements. When an estimator is required to furnish resource data, he must provide a listing of resource requirements in specific numbers. However, given the opportunity, he can generally alter these requirements or even eliminate them by adopting a different method of performance. In some cases activities can be interrupted temporarily. In some cases resource units can be borrowed from other activities without completely interrupting them. Sometimes resource units can be shared by concurrent activities. The clever foreman in the construction project illustration previously used who is faced with a shortage of resource units will often consider all of these alternatives and a few more. He has an ability which cannot be programmed into a computer and which has allowed him to furnish results that have been widely accepted even though, on analysis, the method appears crude.

140. It is the purpose of this chapter to propose a middle course. It is assumed that computer facilities are not available and that, therefore, a manual procedure is required. This manual procedure must be within the limits of practical application. It should take advantage to as great an extent as possible of knowledge of basic network scheduling data and the application of network mechanics. Such an approach cannot be as simple as pushing a button and awaiting a printout of a solution. It must require a considerable amount of experimentation, calculation, and analysis, which will be hard work. However, in cases where a prolonged effort is to be expended in the implementation and management of an important project that requires a considerable period of time for its performance, this effort is justified. Perhaps if it were necessary to programme a large number of very short duration projects, only computer processing could provide practical solutions to resource allocation problems. But the fact that this report concerns single projects, or programmes involving a limited number

of projects, that extend for months or years means that project managers can afford to take the time required by the procedures to be proposed. If they do so, they will be rewarded by a much more intelligent understanding of the problem with which they are dealing than the programmer who submits his data to a "black box" and accepts whatever solution is returned. It can be expected that a more intelligent understanding of a problem should produce more intelligent solutions. Procedures for three cases, each of increasing complexity, will be discussed below.

[illegible]. Manual Resource Allocation Procedures - Preparatory Steps

141. An initial step for the procedures to be discussed is the development of resource schedules for one specific project schedule which, except for resource requirements, would be a feasible scheduling solution. This step is not essential in the Case II procedure, below, but is helpful there also. The simplest schedule to use is that based on the earliest start times of each activity, as shown in Figure 60. In the procedures to be described activities are rescheduled, or other changes are made that may require activities to be rescheduled, in attempts to eliminate resource requirements that are unacceptable and to improve resource schedules. As a result total requirements for each time unit are frequently changed.

142. If the mechanics of physically making changes in the data tabulations are simplified, the possibilities for improving schedules will be expanded. A method for accomplishing this is to use a worksheet consisting of a large piece of cross-section paper mounted on a sheet metal backing. A time scale can be shown horizontally across the sheet with each column of the grid representing one time period in the time units chosen. Then each activity can be represented by a bar cut from a strip of magnetic plastic. Such plastic is inexpensive and can be obtained in rolls of stripping in widths as narrow as 1/4 inch. It may be painted with an enamel paint. Then a wax pencil or a pen using water soluble ink, either of which are easily obtainable, can be used to mark resource requirements on each activity bar. These entries should be positioned to match the horizontal scale of the grid sheet in order that a single entry will appear in each time-unit column over the time span of the activity. It is not essential to

cut the plastic bars to match the duration of the activity. A module, such as 5 time units, can be chosen and all bars cut the same length from the rolls of stripping material. The bar for any activity can be made up by placing these basic modules end to end and leaving a uniform space on the final length blank. It is also helpful, if the number of different types of resources to be analyzed is not too great, to use a different colour bar for each resource. The magnetic strip material can be painted in a number of pastel shades up to at least 6 to 8 different colours. These will permit simple marking with pencil or pen and will make it very easy to visualize the origins of the various resource requirements. If an activity requires more than one resource, different colour bars for each resource can be placed one above another to form a composite bar for the span of the activity. This group of bars can be moved as a unit when rescheduling of the activity is performed. Strips of magnetic plastic can also be mounted across the bottom of the worksheet to receive entries of the totals for each time period. If colours are used to represent different resources, one strip of each colour spanning the entire project duration is used for the totals. Entries can be made in a similar manner to those for the activity bars, i.e. with grease pencil or water soluble ink. As activities are rescheduled these entries can be easily changed by carefully wiping out the previous figure and entering the new total. This arrangement permits making changes in a more practical manner than entries directly on the cross-section paper with subsequent extensive erasing.

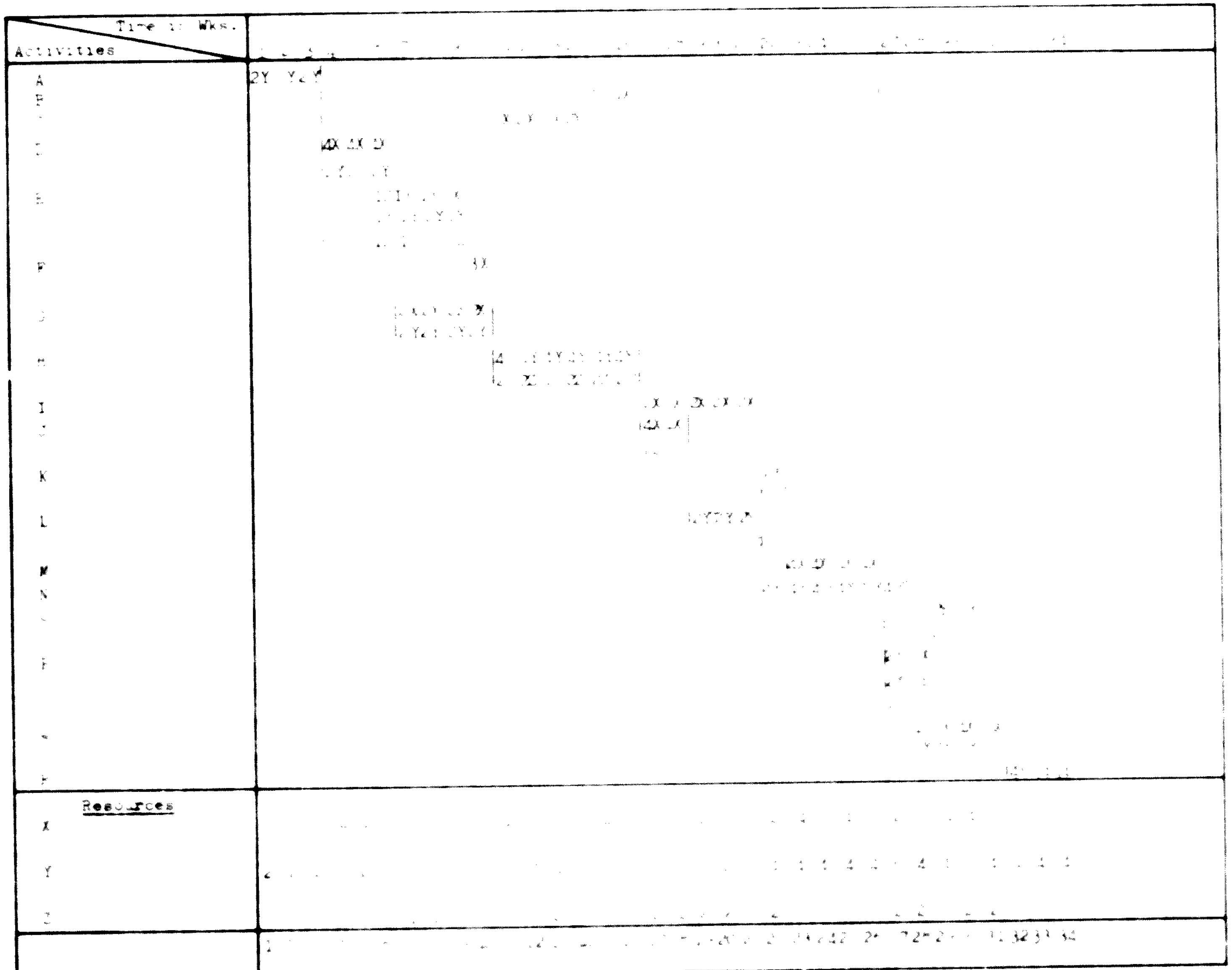
113. An additional refinement to the above procedure to allow experimentation with changes that involve shifting a number of activities is possible. Sometimes a decision concerning the desirability of a proposed change cannot be reached until rather extensive changes have been made. An extra blank strip for each resource can be provided across the bottom of the worksheet adjacent to the strip on which totals per time unit are entered. The bars for each activity involved in the proposed change can be left in their original position and merely be turned over to temporarily eliminate them from consideration. New bars can be made up and placed in the new proposed positions (pieces of magnetic strip can be laid on top of portions of the turned-over strips, so extra vertical space need not be provided). The new totals can be entered on the

extra blank strips at the bottom of the worksheet after the proposed change is completely processed and all affected activities have been scheduled in their new positions. Then these figures can be compared to the previous totals directly above them which have not been destroyed. If the change does not prove desirable, the new data can be removed and the turned-over bars restored to their former positions. If the change proves to be a worthwhile improvement the turned-over bars can be removed, the entries on the original totaling strips on the bottom of the worksheet can be changed to match those on the spare strips, and the spare strips can be blanked out for reuse.

C.1 <u>Case I - Limited Resource Restrictions</u>

144. In some cases the resource restrictions are not severe and the elimination of major peak requirements can be achieved by a relatively simple shifting of activities, mostly within free float ranges. This shifting can be accomplished by moving the magnetic bars representing the activities to a new position and changing the totals affected at the bottom of the worksheet. If an activity is shifted within its free float range, there are no chain effects. Therefore, no other activities need be shifted. This is the simplest possible type of change and yet can be quite effective in many situations. Of course activities can also be shifted in their interfering float ranges if the affected activities are also shifted and all necessary totals corrected. If the chains of affected activities are short ones, a prohibitive amount of effort is not required and such changes may also be practical to process. Critical activities can also be shifted and other activities can be shifted beyond their float limits if the improvement produced is worth extending project duration. However, unless such changes occur near the end of the network diagram, there are likely to be many activities affected and a large amount of data modification required. Therefore, scheduling changes for solving resource problems by Case I procedures are limited primarily to changes involving the use of free float time and such other changes that do not require an excessive amount of activity shifting.

145. Figure 62 shows a rescheduling of the activities of the project network of Figure 58. Only activities with free float have been rescheduled from their early start times, and rescheduling has been limited to the free float ranges.



Time in Wks.
Activities
A
B
D
E
F
G
H
I
K
L
M
N
P
R
Resources
X
Y
Z

The vertical lines opposite each activity indicate the possible time span for rescheduling by marking the earliest start time and the end of the activity's free float time. In this rescheduling, project duration has not been changed, of course. Moreover, no activities have been interrupted nor have any activities been replanned to change resource requirements. However, the improvement is impressive. The maximum requirement for Resource [illegible] has been lowered from 14 units to 6 units. Those for Resources Y and [illegible] have been reduced, respectively, from [illegible] to [illegible] and from 4 to [illegible]. The [illegible] unit requirement for Resource [illegible] could quite easily be reduced further to 4 units by simply extending project duration by one week as is apparent from a quick examination of the work sheet. The continuity of resource usage has also been improved. Once a resource is required, it is used continuously, in varying amounts, until it is no longer needed. Resource [illegible] builds up to its maximum that then remains constant except for a single 4-week break, as compared to two breaks and a less uniform requirement in the initial schedule. The schedule of Figure [illegible] is definitely an improvement over the schedule of Figure [illegible] from a resource standpoint, and yet the schedule modifications were very simple to make.

146. It does not seem necessary or desirable to develop a set of strict rules or a definite procedure for rescheduling of this type. It involves a trial and error juggling that can be accomplished easily with the physical arrangement of data on magnetic strips that has already been suggested. Whether the best possible answer is obtained depends on the skill and imagination and luck of the person in charge. But an initial schedule can be considerably improved in most instances. If this person also considers such measures as replanning activity performance, borrowing resources temporarily from concurrent activities, varying resource usage during the performance of an activity, interrupting activities, sharing resources between activities, and a host of other possibilities, he can probably improve his original schedule to a much greater extent. For example, in the schedule shown in Figure 62 it might be possible to replan Activity I for performance in 3 weeks using 4 units of Resource [illegible] per week. This would increase the resource-weeks of effort from 10 (5x2) to 12 (4x3) to allow for some inefficiency in the alternative method. But it

[illegible] requirements [illegible] [illegible] schedule instead of a [illegible] schedule [illegible] if the need for doing so is brought to the [illegible] with the method of performance [illegible]

<u>Case II - Different Resource Restrictions</u>

147. In some cases, satisfying resource restrictions is more difficult than in the previous example. Rescheduling activities within their free float [illegible] the reductions that are necessary. Other rescheduling [illegible] activities, as already discussed, and if there are [illegible] experimental, trial-and-error procedure [illegible] very practical. [illegible] changes that involve a number of activities is difficult with a manual procedure and imposes practical limitations on the number of scheduling improvements that may be attempted, a real problem is posed to improve project schedules that are unacceptable and that are subject to severe resource restrictions. A different method of approach from Case I will be helpful in meeting this problem. Instead of starting with an initial schedule and attempting to reschedule activities within it, the programmer can start at the beginning of the project work and gradually build an acceptable project schedule activity-by activity. Rescheduling of activities will be confined, at any moment, to the activity then being scheduled and possibly a very limited number of activities that are in conflict with it for available resource units. In this procedure an activity is not scheduled for performance until resource requirements can be satisfied. If this requires postponing performance beyond their float periods, project duration is extended. It is not uncommon to resort to this measure because when resource restrictions are severe project completion usually must be delayed. The objective of the procedure becomes that of minimizing the extension of project duration while satisfying all resource requirements within the limits of resource restrictions.

148. There are three principal questions to answer in this procedure. One involves the limits that should be established for resource availabilities. A

[illegible]

[illegible] them. [illegible]

be resolved as they arise.

[illegible] these [illegible] resource [illegible] involved starting with an exist[illegible] resource requirements are [illegible] is often somewhat flexible [illegible] make further improvements [illegible] can be made after [illegible] In the case [illegible] procedure, [illegible] beginning of the process before [illegible] establishing these limits [illegible] source that is actually available. Theoret[illegible] to define since if one is willing to spend [illegible] most always some measures possible that [illegible] However, if a criterion of normal maximum [illegible] and the development of a schedule [illegible] days in project completion become necessary [illegible] specific type, this resource limit [illegible] the units considered in making a [illegible] the limit beyond that which [illegible] be desirable to repeat the scheduling procedure [illegible] depend on the extent to which the project [illegible] resource limit that is being revised.

150. The Case II procedure does not [illegible] maximum limits. The scheduling of activities [illegible] established limits rather than on the [illegible] resource schedule curves. However, by progress[illegible] setting [illegible] and repeating the procedure, effective level[illegible] problem with this approach is that each scheduling pass requires [illegible] amount of effort. Iterative methods for cases such as this are not particularly suitable for manual processing. Therefore, it is often desirable [illegible] limits that are more severe than those based on normal maximums [illegible] force additional levelling and to keep resource peaks as [illegible] setting such limits there should exist a good possibility that [illegible]

[illegible]

[illegible]

[illegible] be presented for [illegible] the data may suggest reschedu- [illegible] rather than postponing the [illegible] suggest interrupting and reschedu- [illegible] and certain activities [illegible] their durations). Sometimes [illegible] temporarily borrow resource [illegible] requirements over the duration of certain acti- [illegible] on-the-spot decision may be reached [illegible] possibilities. If no al- [illegible] than simply scheduling the [illegible] satisfy resource restrictions, [illegible] this Case II procedure is not [illegible] a definite set of rules. [illegible] programmer may often be able to [illegible] resource allo- [illegible] figures.

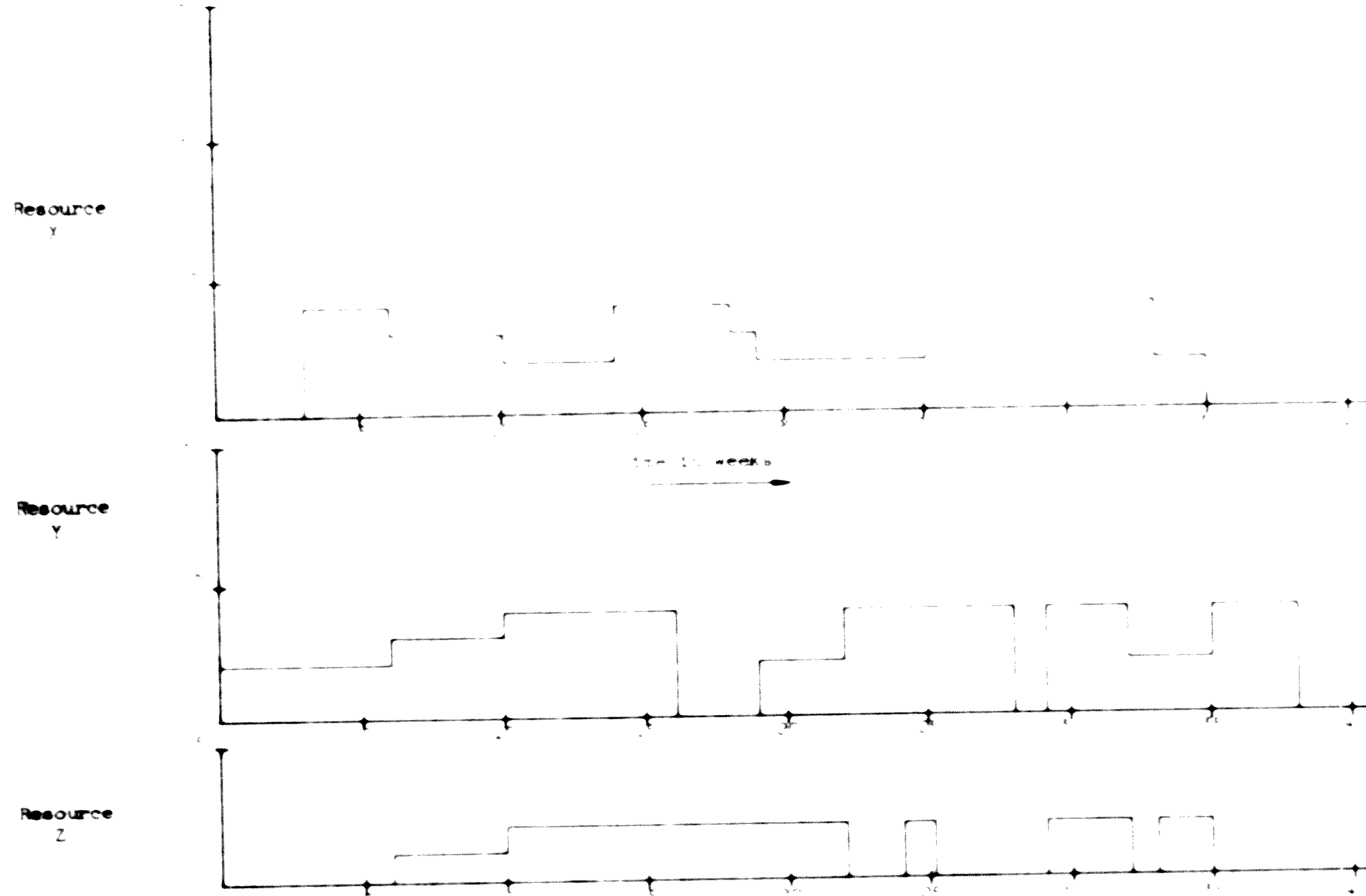
[illegible] when scheduling has progressed to a [illegible] at the earliest that Activity ? can [illegible] three weeks later than its latest finish date. [illegible] the schedule already developed indicates that by [illegible] had previously been scheduled to start [illegible] later than Activity [illegible], can be scheduled on the 19th week [illegible] project completion by one week instead of [illegible] activities [illegible] and P could not be [illegible] those activities at the earliest avail- [illegible]. The final project duration required [illegible] weeks. [illegible] all activities had simply been [illegible] units were available (Acti- [illegible] example that was done), the project duration [illegible] weeks. Figure 14 shows the results obtained by the Case II procedure [illegible] improvements can still be [illegible] the [illegible] in usage of Resource Z on the first week [illegible] scheduling Activity [illegible] a week later. A comparison [illegible] Figures [illegible] and 14 indicates the improvements that have been [illegible].

Case III. Projects of Long Duration

157. Projects of long duration require more effort for resource allocation scheduling principally because of the greater number of activities and the resulting larger number of entries to make and to update. There is questionable value in expending considerable effort for very detailed resource programming for that portion of a project to be performed in the distant future. Such programming will undoubtedly have to be repeated due to the changes that will inevitably occur in the performance of the earlier portion of the project. A combination of Case I and Case II procedures along with the use of subnetworks and the [illegible] cutoff method discussed in Chapter [illegible] offers a means of resource programming for projects of long duration.

158. Initially the resource requirements of the entire project should be analyzed to locate any major problems and determine measures for meeting them. For this purpose, only those resources should be considered that are expected to offer definite limitations either because they are scarce or expensive or because the project activities make extensive use of them in relation to their levels of availability. For these activities it is generally worthwhile to develop resource schedules based on early start times. An added short cut that can often be used effectively is to base the resource schedules on a larger time unit, e.g., weeks instead of days, months instead of weeks, or two-month periods instead of months. This preliminary study will help to establish reasonable resource limits to use in the more detailed scheduling procedure that follows, and it may also indicate the necessity for some immediate reprogramming of the project to eliminate major resource problems. If such reprogramming is necessary it is very desirable that the need be recognized and it be done at this time, since it will probably extend project duration. If a reasonably valid project completion date can be determined in the preliminary analysis a lot of effort will be avoided in the detailed analysis in an attempt to maintain a project duration that will eventually prove to be completely unrealistic.

159. Having completed the initial analysis, detailed schedule development using the Case II procedure can be performed on subnetworks that cover a limited time period in the immediate future. In these studies, additional resources can be considered because much smaller networks permit the additional data to be handled. A shorter time unit may be used as the basis for schedule develop-


Resource
X
Resource
Y
Resource
Z
weeks

[illegible] or more [illegible] month or even a day instead of a week. Other advantages [illegible] new activities may be further subdivided to expand the program in more detail. The effects on project completion of rescheduling any activity will be known by the amounts by which updated activity late finish times have to be exceeded. The effects on the preliminary resource schedules for the period ahead of the current subnetwork will not be known until subnetwork development has progressed to the period of interest. If extensive changes are made during the programming of subnetworks and there is reason for concern about major effects on the overall project schedule that were developed in the preliminary analysis, then procedures for this overall study may be repeated. But the major effort once subnetwork programming is commenced will be confined to very detailed programming for a limited period ahead. This type of programming is practical and profitable with this subnetwork approach, while this would not be true if it had to be accomplished using the entire project network.

16. The procedures outlined in this chapter should permit effective resource allocation for any major project of extended duration even though these procedures have been restricted to non-computer methods capable of being applied at the project management level.

[illegible] ALLOCATION

[illegible]

[illegible] industrial development programme [illegible] involve more than one project.
[illegible] projects [illegible] at the same time either throughout all or
during part of the [illegible] because resource limitations are
likely to [illegible] these projects, resource
allocation [illegible] Where each project draws its re-
sources from its own resource pool, the single-project resource allocation
procedures of Chapter [illegible] However, if the projects draw their
resources from a common pool, the fact that these projects are under control
of a single overall programme management suggests that conflicts for scarce
resources [illegible] manner that benefits the overall programme.
This requires a coordinated effort to consider the projects jointly rather
than separately. A procedure for multi-project resource allocation is
desirable.

1[illegible]. It [illegible] seem that the multi-project resource allocation problem can
[illegible] the same as the single project problem, since
[illegible] treated as subnetworks and combined into a single
master network [illegible] programme. The resulting problem would be more
complex [illegible] larger size. However, there are certain
other [illegible] case which add further to its com-
plexity. [illegible] that the different projects generally have dif-
ferent [illegible] serious to the effective implemen-
tation of the overall programme to delay one project rather than another.
[illegible] of priorities needs to be taken into account as long as there are
valid reasons for [illegible] priorities. Another factor that is sometimes in-
volved is that of mobility of resources between projects. There may be complete
interproject mobility of resource units or, on the other hand, each move
of resource units from one project to another may represent considerable
effort, time, and expense. This is usually a matter of the relative
geographical location of the projects and the degree of mobility of the re-
source itself. For example, one extreme arises when multi-project resource

allocation techniques are applied to the scheduling of the engineering and pre-construction phases of several projects within a programme. The resources may consist of various classifications of professional and skilled personnel, often employed under the same roof. In this case there is complete mobility of the resource units from their employment on one project to employment on another. A shift to a new project might only involve clearing one's desk of one set of plans and starting to work on another. However, the other extreme might arise in the construction phase of the same projects. They might be situated at widely separated locations, and each might require certain types of heavy equipment that are in very short supply. This would require moving units of equipment from site to site. However, each such move would involve a major cost as well as time which should be considered in scheduling the activities on the various projects which will require these moves.

163. Of course there are many ways in which the multi-project resource allocation problem can reach still higher levels of complexity. It is possible that some project resources will be drawn from the common resource pools of the overall programme while others come from pools available to a single project only. Or, there can be even more complicated combinations where projects within a programme have common pools for certain resources with certain other projects and different common pools with others. In a similar fashion, certain resources may have interproject mobility with respect to some projects but not with respect to others.

164. It is the purpose of this chapter to suggest possible modifications of the single-project resource allocation procedures of Chapter 5 that will meet these general types of additional complexities. It is not within the scope of this report to develop detailed procedures for the multi-project case nor to present a detailed treatment of sample problems.

B. A Procedure for Multi-Project Resource Allocation

165. The projects in a programme that draw resources from a common pool need to be considered jointly to determine the most favourable resource allocations for the entire programme. The basic scheduling data for each project (earliest

and latest start and finish times) can be computed separately. If all the projects are not to be commenced at the same time, the starting times of their initial activities should each be related to the overall programme time schedule. For example, if Project A is to start [illegible] days after the beginning of the first project in the programme, the earliest start time of its initial activity should be set equal to [illegible] before the forward pass calculations are commenced.

166. If the projects have different priorities in the opinion of programme management, these priorities need to be expressed in a quantitative manner. This is not always simple, nor perhaps diplomatic, to do. However, there will be conflicts for available resources which must be resolved. Otherwise a resource allocation analysis would not be needed. Resolving these conflicts will often require extending the durations of at least some of the projects in varying amounts. The higher the priority of a project, the less desirable it is to extend its duration. One logical basis for expressing priorities is to develop the costs incurred per time unit of added project duration. This is the same figure used in the time-cost trade-off procedure of Chapter 4 to determine when the most favourable schedule had been achieved. It is equal to the indirect costs for project performance during the time range following its earliest possible completion and also includes those costs representing lost income or extra expense as a result of not placing the completed project in operation. If such figures can be developed, they give an ideal quantitative basis for project priorities and can be used directly. A less precise alternative is to arbitrarily assign priorities on the basis of the judgment of programme management.

167. The basic procedures of Chapter 5 can now be applied. Assuming that the resource restrictions are sufficiently severe to require the Case II procedure, modifications of this procedure will be discussed in further detail. Resource availability limits must be initially set and can be determined by one of the methods suggested in Chapter 5. Then activities should be ordered in late start sequence. Rather than list the activity labels on the worksheet, as proposed in Chapter 5, there are advantages in listing the activity data on cards and using one card for each activity. One reason for this results from

the larger number of activities involved. It is generally not possible to assign one row on the grid of the worksheet to each single activity. Rather a band of rows should be assigned to each project. The magnetic bars which carry the activity resource data may also include the activity label. Then non-concurrent activities in the same project can be placed in the same grid row. As a result the band of vertical space required for each project need only be sufficient for containing the maximum number of activities that can be performed concurrently. Hence, this is a space saving measure. Another reason for listing activities on cards is that the ordering of activities will change during the scheduling procedure. If the completion time of one project is extended the late start times for activities in that project are all increased by an amount equal to the increase in project duration. However, the late start times of activities in other projects are unaffected. Since late start times are the basis for ordering the activities for scheduling, reordering is necessary due to the changes that occur. With the activities placed on cards this reordering is simple. The original late start time of each activity is listed in the upper corner of its card, and the cards are arranged in order based on these figures. Then scheduling of activities commences with the activity on the first card, then the next, etc. Whenever a change in project duration becomes necessary in order to resolve resource conflicts, the magnitude of the change is listed in an assigned column of the worksheet in the horizontal band of that project in a similar manner to the listing of delays in Figure 63. As each card is examined preparatory to scheduling its activity, the total delay of its project is noted from the worksheet. The late start time on the card is updated if necessary, and the card is reordered according to its updated late start time if necessary. If the card needs to be reordered it will not be scheduled at that time but, rather, the next card will be examined and the process continued.

168. When resource conflicts do occur during the scheduling process, their solution is complicated by the matter of project priorities. If there are no priorities and the only objective is to complete the overall programme in the least possible time, then all the projects can simply be combined into a single network. They can be tied together at the beginning by an event

node representing programme commencement and at the end to an event node representing programme completion. The problem is reduced to one of single project resource allocation, although the network may be quite large and with the exception that some attention may need to be given to interproject mobility in shifting resource units. However, in most cases, priorities for completion of the individual projects within the programme do exist and need to be considered. As each activity is scheduled, its gummed strips are placed on the grid sheet. It is helpful to enter the activity label on the upper left corner of this bar assembly and its latest finish time in the upper right corner. Latest finish time is preferable to latest start time for this purpose since the activity may be interrupted during the scheduling process. This figure will serve the purpose of showing when the project is being extended in duration and, during the scheduling of other activities, will permit a rapid determination of the float time of any previously scheduled activity. The original late finish time value should be used. It can be quickly updated by glancing at the column in which project delays are tabulated and adding the amount of any delay already incurred by its project. The float time of an activity can be determined easily by subtracting the date on which the activity is scheduled for completion (the time unit of the column in which the activity bar terminates) from its updated latest finish time. As activities are scheduled the initial attempt is made to start them immediately after the latest of their preceding activities have been completed. The identity of the preceding activities can be noted by reference to the appropriate network diagram, or it is helpful to list each activity's preceding activities on its card. The bars of the preceding activities are observed on the worksheet to determine the earliest possible start date for the activity being scheduled.

160. Due to resource limitations it may not be possible to schedule an activity at the earliest date following completion of its preceding activities. In this case either postponement of this activity, rescheduling of other activities, or some other measure is required. The entire group of concurrent activities requiring the limiting resource, including activities previously scheduled and the activity currently being scheduled, are considered together. The first objective is to try to confine any rescheduling to float ranges if possible. The second objective is to observe priorities if there are alter-

[illegible] possible [illegible] causes this would [illegible] to provide the [illegible] ferring floats during [illegible] shows rescheduling is not [illegible] rescheduling will [illegible] should be sought [illegible] sary. If equal amounts [illegible] ternative solutions, the [illegible] priority would be selected [illegible] delays should be weighted [illegible] solution. For example, suppose that [illegible] on a cost of $500 per day if it is delayed [illegible] of $0[illegible]. Suppose further that [illegible] A would have to be delayed [illegible] In this case it would appear more desirable [illegible] higher priority project since [illegible] unfavourable solution involving reschedul[illegible] been determined, alternatives [illegible] These might, for example, require reprogramming of activities [illegible] resource requirements or the [illegible] resource avail[illegible]ability levels. The best solution [illegible] and then the next activity in order [illegible] continues in this manner.

17[illegible]. The matter of interproject mobility of resources [illegible] should also be considered during the scheduling process but formal rules are difficult to outline. If there are costs associated with interproject shifting of resources, then these costs should be developed to assist in scheduling decisions. In general, if resource units will be required again on a particular project, an alternative to shifting them to another project is to retain them where they are until they can be used again. This may cause project delays in other projects. If so, it becomes a matter of whether the savings realised by not transferring the equipment away from and back to the project

[illegible]

[illegible]

[illegible]

[illegible] for Highway Programmes" Proceedings of a [illegible] Safety Foundation [illegible] Washington [illegible]

[illegible]

[illegible] and [illegible] a rather formal [illegible] state that re- [illegible] independent [illegible] time-cost [illegible] is applied to time-cost [illegible] costs against changes in pro- [illegible] ability levels or in resource [illegible] procedures would be dif- [illegible] trade-off efforts.

[illegible] without resource allocation [illegible] that [illegible], accord- [illegible] danger. A pro- [illegible] conditions and take actions [illegible] which are actually very in- [illegible] than any maximum re- [illegible] expenses related to the varia- [illegible] respect to time. Time-cost trade-offs [illegible] a simple sheet without consideration of resource re- [illegible] trade-off must of necessity affect the duration of [illegible] whether resource requirements of those activities [illegible] change will produce changes in resource [illegible] change in resource schedules must produce an ef- [illegible] effects should be considered in establishing the [illegible] time-cost trade-off is based. If they are considered, [illegible] faulty conclusions. Some changes [illegible] impossible due to lack of available re- [illegible] that time-cost trade-offs should not [illegible] independently [illegible] resource allocation analysis.

[illegible] appl[illegible]

[illegible] whether resource allocation and time-cost trade-off procedures can or [illegible] independently, the use of both is desirable to produce a [illegible] good project implementation plan and schedule. For example, a know- ledge [illegible] resource schedules [illegible] advisable in determining valid timeost trade-offs, but it opens up a whole new area of possibilities for additional favourable trade-offs. The knowledge of the existence of idle resources at specific times permits changes to be proposed which utilize these idle resources. Such changes may be possible to accomplish at zero or very low cost slopes. If a piece of equipment is idle during a certain time period, it may be possible to use it in the performance of activities scheduled during this time period to reduce their costs even though such use could not be justified if the equipment were not idle. Or if men are idle during a given time period, it is often possible to employ them on activities in progress even though the resulting crew sizes are larger than the ideal sizes for maximum efficiency. In both cases performance times can be reduced at little or no extra costs.

14. Resource allocation procedures may require postponing critical activities or scheduling non-critical activities beyond their float ranges, particularly if resource levels are difficult to satisfy. In either case, gaps in the critical path will be created. If any critical activity (i.e. any activity that has zero total float) is not scheduled at its earliest start time, its preceding activities will have float time and therefore cease to be critical. Then there will be no single critical path across the entire project network. After extensive resource allocation scheduling there may be very few critical activities remaining, in the sense in which critical activities have been defined so far. This has a rather drastic effect on the application of the time-cost trade-off procedures outlined in Chapter 4, since those procedures require the use of critical activities. Unless the definition of critical activities is broadened, there may be little opportunity for the application of time-cost trade-offs due to a lack of critical activities. However, if a critical activity is considered to be any activity whose performance has a direct effect on project completion, new opportunities for time-cost trade-offs are presented. Those activities whose resource requirements cause the scheduling that produces the gaps in the critical path can also be considered critical. Then methods for changing their performance and resource requirements become effective measures for changing

project duration. Hence, many new time-cost trade-off possibilities are introduced. Or, as pointed out in earlier discussion, the whole question of revising resource limits can be treated as a time-cost trade-off problem.

A Procedure for Combined Applications

271. A joint application of resource allocation and time-cost trade-offs has been shown to be desirable. This joint application will require modifications of the procedures already developed for independent applications. Therefore, it is appropriate to conclude this report with a proposed procedure for combining these techniques.

1. Develop a project implementation plan and reduce it to a network diagram. Attempt to minimize the amount of activity breakdown and detail while arriving at a reasonable realistic representation of the project implementation plant. Base network relationships primarily on physical sequencing requirements and avoid sequencing based on resource restrictions unless the programmer has no doubt concerning their necessity.

2. Obtain time estimates for each activity in the network. It is desirable that these estimates be based on those methods of performance that involve the least direct costs. This will allow an orderly development of the time-cost curve from a starting point at the "normal" solution. (It is not essential that this starting point be used. A "conventional estimate" starting point has also been proposed[12] when actual performance is known to be required in considerably less time than "normal" performance).

3. Perform the basic scheduling calculations to provide earliest and latest start and finish dates and a knowledge of critical activities.

12 John W. Fondahl, "A Non-Computer Approach to the Critical Path Method for the Construction Industry", Technical Report No. 9, Dept. of Civil Eng'r., Stanford Univ., 1961

4. Establish a cost figure that represents the value per time unit of any reductions in project duration. This will serve as a standard in determining if proposed time-cost trade-offs should be made. Then perform any time-cost trade-offs that can be accomplished at zero cost or at low cost slopes relative to the limiting figure just established. After critical activities have been identified it is often possible to expedite them by more detailed planning or to break them down further to allow additional overlapping in their performance, all with little or no cost increase. It is not necessary to develop either normal cost data or crash time and cost data for any of the activities. Decisions can be made solely on the basis of changes in time and cost of the critical activity being considered and the resulting change in project duration. It is true that these time-cost trade-offs will be made blindly with respect to the effects on resource schedules which have not been developed yet. However, once resource schedules are developed, time-cost trade-offs involving analysis of the entire network will require excessive data handling in revising those schedules. If time-cost trade-offs are made at this stage of the procedure and are limited to those involving low cost slopes, they will probably still be justified after the additional opportunities furnished by a knowledge of resource schedules are known. Moreover, those changes that will prove ineffective due to unfavourable resource demands can be reversed later, and generally the cost can be recovered by the "sell-back" of the time.

5. Develop resource schedules for selected resources that are known to be scarce or expensive or which seem to be needed in such quantity in relation to their availability levels that they are likely to create problems.

6. Examine the resulting resource schedules and the project duration for overall reasonableness. It is to be expected that resource schedules can be improved by measures to be taken and, if resource restrictions are not unduly severe, that project duration can be decreased by additional expenditures. However, if the existing results are so poor as to indicate that the basic implementation plan is probably not workable and that major reprogramming is advisable, it is desirable to accomplish this before further detailed analysis is undertaken.

7. The procedure from this point will depend on the severity of resource restrictions and the complexity and duration of the project. Assume a difficult situation involving low resource availability limits and a complex project extending over a considerable period of time.

8. Working with the entire network, attempt to develop solutions for eliminating resource peak requirements that are well above availability limits. Due to the amount of data involved such solutions, of necessity, will be largely limited to rescheduling within free float ranges or in interfering float ranges where chains of affected activities are short, or to other measures that have limited chain effects. Further time-cost trade-off analysis is generally impractical due to the amount of data that would be required to be updated.

9. Establish the resource availability limits for further detailed levelling.

10. Develop a subnetwork for the initial portion of the project using the dateline cutoff method. Use a time period for the subnetwork that produces a small enough network (after some further expansion if a more detailed breakdown is advantageous and after the addition of any other resources that deserve consideration also) to allow practical application of time-cost trade-off and resource allocation techniques. Do not use a longer time period ahead than that which warrants detailed programming in view of the amount of changes and resulting updating that is likely to be incurred.

11. Apply the Case II procedures discussed in Chapter 5 to develop a schedule that meets resource availability limits. Whenever conflicts develop that would require project extension, consider the application of time-cost trade-off measures such as replanning activities to reduce resource requirements, expediting activities to avoid concurrent performance that produces resource conflicts, or changing resource limits.

12. Having developed a scheduling for the subnetwork activities which meets resource restrictions, attempt to develop further time-cost trade-offs that appear to shorten project duration. Whether they actually do so will

[illegible]

[illegible] computer procedures that [illegible] that are considered [illegible] capacity computers [illegible]. The methods [illegible] claimed [illegible] extent upon the application [illegible]. It is believed that the [illegible] application of the procedures [illegible] more formal, mathematically or [illegible] are not intended for [illegible]

76. 02. 06