THE UNIVERSITY OF MICHIGAN

INDUSTRY PROGRAM OF THE COLLEGE OF ENGINEERING

A COMPUTER PROGRAM FOR LEVELING RESOURCE USAGE

R. V. Galbreath

August, 1964

IP-679

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

A COMPUTER PROGRAM FOR LEVELING RESOURCE USAGE

Robert V. Galbreath,[1] M. ASCE

SYNOPSIS

The Critical Path Method study for a construction project does not provide all the information requisite to a comprehensive and practical schedule for that project. An investigation into the results of such a study is necessary to reveal the demands which would be placed upon construction resources and the rates of change of these demands. The process by which this latter investigation is made and which, further, attempts to bring these usages and rates within workable limits is commonly referred to as "Resource Leveling." The object is to "level" the day-to-day, week-to-week usages of resources, principally manpower, within the current availability of those resources.

The leveling problem differs from CPM in that it is not susceptible to precise mathematical statement. There is as much difficulty attendant with deciding how a leveling problem shall be approached as with performing the operations once decided upon. Lack of general agreement as to how leveling should be accomplished and the apparent uniqueness of each construction project problem seem to preclude the formulation of a "universal" program capable of solving any and all leveling problems.

---

1 Assistant Professor of Civil Engineering, University of Michigan, Ann Arbor, Michigan.

The program described here has been formulated with three purposes in mind: (1) The manipulations built into the program have been designed with simplicity and logical directness as objectives, and the program is to operate within full view of the user. Also, features have been incorporated which provide the user with the ability to manipulate his problem data to test alternate plans and schedules. (2) The program is so consituted as to be readily adaptable to innovation. The testing of leveling schemes is as much an objective as the working of problems. (3) The intent is that by making available a basic program which can be modified or refined or, if it is warranted, completely supplanted this whole topic of resource leveling may be discussed with greater candor than has been the case in the past. This is an important problem which must be dealt with effectively if recent progress in the development of construction management tools is to be continued.

INTRODUCTION

The accomplishment of the Critical Path Method phase of planning and scheduling a construction project leaves undone an important task which must be performed if the benefits of the CPM study are to be fully realized. This operation is known by several names, probably the most common of which is "Resource Leveling" or sometimes simply "Manpower Leveling." The earliest presentations of the Critical Path Method expressly recognized the need for further investigation into the results of a CPM study for a given project with regard to the demands which would be placed upon construction resources and the rates of change of these demands. While CPM continues to become more widespread in its usage, it is suspected that few of its users, even the most experienced, have at their disposal an efficient method for examining and solving this very important companion problem. The results of the CPM study for a project may appear quite feasible on the surface, but investigation is likely to reveal that exceedingly high levels of resource requirements or far-ranging swings in such requirement levels make revisions or modifications of the CPM schedule as first rendered a necessity.

This basic problem, i.e., the need to efficiently utilize the resources which are required to accomplish the work, confronts every manager of construction whether CPM is employed or not. The objective, of course, is to "level" the day-to-day, week-to-week usages of resources, principally manpower, within the current availability of those resources. Further, transitions from one level of usage to another should not be so abrupt as to cause disruption in the steady flow of construction activity.

A few of the adverse influences working against efficient project operation which are present when wide variations in resource usage occur are as follows: (a) Hiring and layoff processes involve extra cost. The same is true of shipping equipment to and from the project site, whether it be contractor-owned or rented. (b) New men are less familiar with a project, and hence less efficient, and supervisors require time to become familiar with the capabilities of new people. (c) The better tradesmen naturally search out those projects or employers having a reputation for providing longer periods of sustained employment.

Systematic attempts at the solution of the resource leveling problem, while perhaps appearing with greater regularity in recent years, certainly predate the advent of computers and computer programs written to provide aid in the solution of problems of construction management. It is true, however, that the devising of computer-oriented methods of planning and scheduling construction work, most notably CPM, has constituted a sizable step forward in the quest for techniques which will permit the construction manager to search out the most systematic and efficient rate and order of utilization of the resources which must go into his project. Before CPM became available there were few systems for selecting the "best" order of activities for a given project which could, in turn, be used in preparation for the attempt at leveling resource usage. The concepts of "float" and "event time boundaries" within which an activity must be accomplished have provided a logical solution upon which schemes for leveling can be superimposed.

As was noted above, the earliest announcements of CPM fully recognized the companion problem of resource leveling. It generally has not been emphasized, however, that there are basic differences in the two problems which make the solution of the second far more involved than that of the first. Acceptance of the logic and assumptions of CPM has come with relative ease, and the resulting program is one of precise mathematical steps. There is only one CPM solution for a given arrow diagram.

The same cannot be said, however, of the problem of resource leveling. If assumptions could be agreed upon, a precise mathematical solution perhaps could be provided, but so large a number of assumptions would be necessary upon which to base such a solution that the likelihood of these being widely accepted seems extremely remote. Also, it should become apparent that the assumptions must vary with the specific project problem submitted. These factors lead to the conclusion that a "universal" program capable of solving all resource leveling problems is not likely to appear in the immediate future.

The purpose of this discussion is not to attempt to provide a scheme to solve the ultimate in resource leveling problems, but rather the program here described is intended as an intermediate step in that direction. In order to accomplish this purpose it is necessary to provide a means of arriving at usable solutions to present, practical, less-than-ultimate problems.

DIFFERENCES BETWEEN THE CPM AND LEVELING PROBLEMS

An example will serve to review briefly the methods which have been employed in the past to cope with the resource utilization problem and may serve to clarify the various complications which are encountered when one is confronted with this task.

Table 1 lists five activities which are but a part of an entire steam power plant construction project. Opposite each activity description are figures representing the number of pipefitters required to perform the activity and the estimated number of working days which will be required to execute the activity. (Time units for this paper will be "working days" unless otherwise noted.)

A number of things should be apparent concerning this selection of activities:

(a) Using only five activities to represent the installation of a main steam system in a power plant probably is an oversimplification of the work involved.

(b) The main steam system is only one of perhaps thirty or forty piping systems which would be undergoing installation in the power plant example chosen.

(c) Piping installation is but one phase of the entire construction and engineering project.

(d) The numbers of pipefitters required to perform the work involved in each activity constitute but one of many resources necessary to accomplish each of the activities listed.

Factors such as these may affect the degree of accuracy of a CPM network, but the validity of the program will remain unimpaired. The results, though imperfect, are usable. Such factors, not properly handled, will have a far more devastating effect upon the leveling solution, however, and all semblance to reality easily may be lost.

The basic principles and assumptions upon which CPM has been developed can be, indeed, frequently are, illustrated by a simple arrow diagram consisting of five, or so, activity arrows. The logic and arithmetic employed in the solution of such a sample diagram, however, remain valid regardless of the size of the diagram to which they are applied.

The same is not true of the resource leveling problem. A diagram such as the one which will be presented herein perhaps is susceptible to general agreement as to the leveling solution which could be pronounced "best". It should be readily apparent, however, that any scheme devised for arriving at this "best" solution for a simple problem such as this, would likely be ineffective for leveling resources for a larger or more complicated network. To illustrate:

(a) By oversimplifying the representation of the work involved in installing this system is there not a good possibility that misleading results will be obtained? Should not there be a larger number of activities, and, if there were, would not the resulting resource usage be different than that shown?

(b) Is not this but one group of a much larger number of activities which place demands upon the one resource considered?

(c) Is not the task of installing the mechanical systems of this plant, along with the attendant demands upon available resources, but a part of the whole project which must be considered in its entirety as regards resource allocation and relative importance of same?

(d) What of other resources which may be required for each of these five activities? May not the availability of the following be just as important as that which is shown?

(1) Pipefitter Welders.

(2) Welding Machines.

(3) Stress-Relieving Machines.

(4) Layout Engineers.

(5) Electricians.

(6) Tugger Hoists.

(7) Yard Handling Equipment.

(e) When more than one resource is involved which is to be considered paramount? Is there a priority order for the resource requirements of an activity? Is there a priority order for the resource requirements of the project as a whole?

(f) What of the relative levels of availability of the resources and the interplay of these as regards any given activity and the project as a whole? When a manager is to determine the availability levels within which the project is to operate (and these almost always are arbitrary within certain bounds) what is there upon which he can base this judgment?

(g) If a satisfactory solution were found for one resource leveling problem is it at all likely that the same scheme or program will yield a satisfactory solution to the next problem?

These, and other considerations which may occur to the reader give rise to the suspicion that any system or scheme or computer program which is claimed to be capable of providing "the best" solution to a resource leveling problem is either oversimplifying the situation or making assumptions which, if fully understood, would not necessarily be acceptable to construction managers.

For these reasons the program presented here is designed with three objectives in mind:

(1) <u>The manipulations of activities which have been built into the program are intended to be as logical and direct as possible.</u> The criteria used to arrive at a "trial solution," are, it is felt, those which an experienced construction manager might be expected to use in an attempt to solve his resource leveling problem. An effort has been made to keep these activity movements within the "view" and understanding of the manager in order that he may apply his judgment to make alterations in the planning and scheduling as required. Also, several features have been incorporated which provide the manager with the ability to manipulate his problem in order to seek out better solutions.

(2) The computer is to be used as a "workhorse" for testing not only variations on network logic and activity schedules but also for testing alternate schemes for the manipulation of activities. One approach to the design of a computer program for solving the resource leveling problem would be that it must first be decided what the entire scheme is to be before a program is written to accomplish its ends. While this method would seem to be logical, it is highly unlikely that such an approach will be entirely fruitful. One difficulty is that it is virtually impossible adequately to test a scheme which may be devised without resorting to a computer. Even the simplest of problems and trial schemes take a considerable amount of time to test by manual methods. Also, there may come a time when increasingly sophisticated programs can be written which will make evaluations and decisions which, in the program here presented, must be made by the user but, it is submitted, much more practical data must be collected before that point is reached.

(3) It is intended, in disclosing the program in detail, to encourage others to take an interest in this problem and to come forward with substitutions for, or variations on, this basic scheme. One such refinement, for example, would be to superimpose upon the scheme additional criteria for manipulation of activities. It is hoped this program is basic enough that it may be modified, preset, refined and biased to suit the tastes of the experimenter. Also, the hope is that it will be found to be of practical value to apply the program as it now is written and that the construction industry may have the benefit of a useful tool immediately and that data may be collected thereby with which to suggest improvements.

EXAMPLE PROBLEM

Before the appearance of CPM it was quite common to represent a schedule for construction project activities in the manner shown in Fig. 1. Most often the sequence selected followed the experience of the person making the schedule. Usually, not too much objection would be raised as the pattern was fairly familiar, and any differences generally would be a matter of personal preference. The contractor would find it not too difficult to work within the general schedule, and if he made any attempt at resource leveling at all it probably took the form shown in Fig. 1, namely adding the requirements of the most important one or two, or perhaps three, resources to determine the daily total requirements for the project. Some minor shifting might then take place until the levels of the rquirements were somewhat consistent and were within the realm of feasibility. It can be seen that criteria for attacking the leveling problem were few and relatively crude. The combinations of possible individual activity sequence for a project of any reasonable magnitude were virtually infinite. Since there was available no logical system for easily exploring the sequencing of activities the schedule finally agreed upon amounted to a selection based upon past experience, intuition, persuasiveness of individual personalities, and the sheer lack of time to explore more than a few of the more obvious possibilities.

The construction manager faced with our example problem, were he a user of CPM, would see to the resolution of a plan for accomplishing the work which would be acceptable to all concerned. This, of course, would take the form of the "official" arrow diagram for that project.

For the sample portion of the example power plant project this diagram might look something like that shown in Fig. 2.

At this point it is imperative that there be a clear understanding of an important premise which appears, to the author at least, to be axiomatic and upon which this paper is based. The solution of the planning and scheduling problem depicted by the arrow diagram, even though pronounced "official" by those with the authority to do so, is neither the only, nor the "best" solution possible. At most it is the "best" solution arrived at after careful consideration of all the factors which were available to that point. Not among the available factors which bear on the ultimate selection of an arrow diagram configuration and the execution of the work was the effect of the resulting work schedule upon resource usage. (It will be remembered that an arrow diagram is constructed assuming unlimited resources.) Any scheme for the solution of the resource leveling problem which takes from management the ability to review and revise the whole plan, or any part thereof, for accomplishing the work and substitutes some mathematical solution which is beyond the "view" or comprehension or control of management perverts the true purpose of these management tools. Those who contribute to the development of an arrow diagram and the CPM solution do so with the understanding that each sub-plan which they contribute and which becomes a part of the entire diagram merely is a plan, probably one of many which could be conceived, for that portion of the project in which they are involved. To place even a sub-plan, much less the integrated schedule, beyond the control of management to change or consider for change would be to alter the basic rules upon which a CPM solution is structured. Any resource leveling program should be so devised as to encourage rather than discourage frequent review of the entire plan, the CPM portion included, in light of the results provided thereby.

Returning to the sample problem; the values rendered for each activity by the CPM program appear in Table 2. It is assumed these are familiar to the reader, but a brief review of the notation used here may assist in understanding the tables.

→ = An activity - One part of the total project with a corresponding demand on time and resources;

○ = An event (node) - The point in time at which one or more activities have been completed and one or more activities may be started;

I = Identifying number assigned to the node at the tail of an activity arrow;

J = Identifying number assigned to the node at the head of an activity arrow;

TE(I) = Earliest event time of the I-node of an activity;

ST = Starting time of an activity (at the outset ST = TE(I) for all activities);

Y = Normal-time duration of the activity;

FT = Finish time of an activity (at the outset FT = ST + Y for all activities);

TL(J) = Latest event time of the J-node of an activity;

FF = Free float for an activity (FF = TE(J) - (ST + Y) , or FF = TE(J) - FT);

TF = Total float for an activity (TF = TL(J) - (ST + Y) , or TF = TL(J) - FT);

ITF = Interfering float for an activity (ITF = TF - FF, or ITF = TL(J) - TE(J)).

Figure 3a further illustrates the relation of the various values which are derived from the CPM solution for each activity.

One minor point which might cause confusion is that having to do with the values shown in Table 2, and subsequent tables, for "TE(I)" and "ST" for each activity. The first day during which work may be performed on an activity is the day following the value given by the CPM program. It is necessary that this or some similar rule be adopted in order to simplify the arithmetic used in the CPM and leveling programs, and this seems to be the standard convention. For example, the activity illustrated in Fig. 4 (activity "C") has a normal duration (Y). The simplest computation which will yield TE(J), the earliest event time of the J-node, (which, of course, also is TE(I), the earliest event time of the I-node for subsequent activities originating at that node) is

$$\text{Max}(TE(I) + Y = TE(J)) \quad . \; . \; . \; . \; . \; . \; . \; . \; . \; . \quad (1)$$

Once accustomed to the idea that all event times are as of the end of the day given the reader should experience little difficulty in visualizing the results of the arithmetic used.

In the left-hand portion of the chart in Fig. 3a are repeated, from Table 2, those values for each activity which are necessary to produce the leveling program bar chart which appears in the right-hand portion of the figure. TE(I) and TL(J) for each activity are represented by vertical lines at the right edge (the end of the respective event-time day as explained above) of the column representing the appropriate working day. These are the event-time boundaries for that activity. During the execution of the floating process in the leveling program TE(I) may be relocated to the right, toward the end of the project. TL(J) values are never changed in this program but remain fixed as does the total duration of the project which remains fixed at CPM normal duration. Trial 1 (Fig. 3a) in the leveling program is computed with all values, including TE(I), in the location given by the CPM program for an all-normal-duration, all-early-start project.

The decision to maintain unchanged all TL(J) values, and hence the duration of the project, throughout the program follows a second important premise which, like the one mentioned earlier, is basic to this discussion. Properly constituted, a leveling program should reveal trouble spots so that the user can re-evaluate the input data or assumptions and make changes as necessary to yield a feasible schedule. The length of the project should not be extended until all other possible remedies have failed to level resource usage as desired. Trouble areas should not be concealed by resorting too quickly to this solution. In this program trouble spots are flagged by the values "EXTEM," the excess of a resource requirement for a day over the resource availability for that day. Almost never is the data supplied by the user so precise and rigid that alterations cannot be made. Indeed, incentive to investigate new and different approaches is quite desirable. If a project is to be lengthened this decision should come from management, not from the internal workings of an imperfect leveling scheme.

In using this approach an additional important benefit is realized in that the concepts of total float and free float and critical path are not sacrificed to the leveling solution. These can be retained for use in the continual updating process which should be a part of project control by CPM.

In Figs. 3a and 3b (and in Fig. 4), the values for each activity are represented by a bar as follows: The left extremity of the bar is the starting time (ST) and may be coincident with, or to the right of TE(I). (ST will always be coincident with TE(I) in Trial 1.) The shaded portion of the bar is the activity and its length equals the duration (Y). As will be explained later the activity may be split into a number of segments (except in Trial 1). The right end of the last shaded (activity) segment of the bar is the finish time (FT). The entire remainder of the bar, open portion (if any) and cross-hatched portion combined, between FT and TL(J) is total float (TF). The left extremity of the cross-hatched portion of the bar is TE(J). If FT and TE(J) are not coincident the portion of the bar between them is left open and this represents free float (FF). If there is no open portion in the bar there is, of course, no free float for the acticity and TE(J) equals FT. The cross-hatched portion of the bar, lying between TE(J) and TL(J) is commonly known as interfering float (ITF).

PROGRAM METHOD

The Flow Chart for this program is found in Figs. 5a, 5b, 5c, and 5d. The accomplished programmer doubtless will be able to find many ways to improve upon its efficiency. Extensive format portions which are necessary in order to keep track of the operations performed and to properly organize results have been omitted in the interest of brevity. A complete list of all notation is found in the Appendix at the close of the paper. A brief description of the program follows:

I. Data Which Must be Supplied:

  A. Activity data from the CPM program, as in Table 2.

  B. Values supplied by user.

    1. Activity Character - Determination for each activity as to whether it may be split into segments in its performance or that it must be performed as a block, i.e., once started the activity must carry through to completion. (Code: -1 = "Cannot be split"; 0 = "Can be split".) If an activity is "splittable" and it is split during the leveling process the number of times this is done and the days upon which these gaps or splits occur will be printed with the results. The decision as to which activities can be split has an important effect on the problem solution and is one of the manipulative opportunities which is under the control of the user.

    2. Resources - Determination of the resources which are to be considered. As presently written nine resources can be handled and all nine can be used for any and all activities. This number could be expanded if the need arose.

3. Priority - Selection of the job-wise priority order for the resources considered. Other priority orders, of course, could be tested in subsequent runs.

4. Requirements - Determination of the requirements of each activity for each resource.

5. Availability - Determination of the level of availability of each resource. Each resource availability level can be preset to any curve desired. These levels can be varied to test for different results in subsequent runs.

6. Activity Order - Selection of the order in which the activity cards are to be fed to the program. This order will have an effect on the outcome except in the simplest of problems. There are several criteria for systematically selecting the ordering of activities, and, of course, the mathematical possibilities of ordering activities are more numerous than could be tested in a reasonable time even with a relatively small number of activities.

7. Miscellaneous -

   a. Number of activities.

   b. Normal project duration. As was discussed above this program does not alter the normal duration of the project. If the user wishes to consider such a change he can do so, but the program will not take such license with the data supplied.

   c. Number of trials desired. As presently written Trial 1 is computed in the all-early-start position. Trial 2 is not, of itself, usable. Trial 3 is computed after the floating operation. Experimentation to the time of writing indicates that subsequent trials usually yield no changes in the results.

II. Operation of the Program:

1. The CPM data and other data supplied is printed in proper format. Also computed and printed with this information are ITF and a value known as "HOLD." (HOLD = TL(J) - ST). The use of this value is explained below.

2. Trial 1, Daily Resource Requirements: Beginning at the first day each activity is examined in its turn to determine whether it is scheduled for performance on that day. If $ST < D \leq FT$ the activity is so scheduled. If this condition exists the requirement of the activity for each resource is added to the appropriate resource requirement running total, TRRQ. After the examination of the last activity the total resource requirement for each resource for that day, TRRQ(1)...TRRQ(RMAX), is printed; and the excess, if any, of each of these, EXTEM(1)...EXTEM(RMAX), over the total availability of each resource for that day, TRAV(1,D)...TRAV(RMAX,D) also is printed, all in proper format. Each subsequent day is treated similarly until the entire number of normal-duration days have been so handled. At that point the grand totals of the daily resource requirements for each resource are printed, RUNTOT(1)...RUNTOT(RMAX). These totals are used to inspect the action which has taken place during subsequent trials.

3. <u>Trial 2, Daily Resource Requirements:</u> Beginning with the first day (D=1) and the first resource (R=1) a total requirement for that resource, TRRQ(1), is computed as in Trial 1. If this amount is in excess of the availability of the resource, TRAV(1,D), for that day an attempt is made immediately to reduce the excess to zero in the following fashion:

   a. The activities scheduled for performance during that day are examined, in order, and the first such activity which has enough Free Float (FF) available will be moved to the right. If not enough Free Float is available the activity will not be disturbed until all other activities scheduled for that day have been similarly examined with regard to their available Free Float and moved if possible.

   b. If the excess has not been reduced to zero at the conclusion of the examination of all activities for available Free Float the activities scheduled for performance during that day are examined once more, again in order, and the first such activity which has enough Total Float (TF) available will be moved to the right. If not enough Total Float is available the activity will not be disturbed. All activities scheduled for that day will be so examined until the excess has been reduced to zero if possible.

   c. If, after having examined all activities in this fashion, an excess continues to exist the value of the excess, EXIEM(1), is stored as is the value of the total resource requirement for that day, TRRQ(1).

The next resource (R=2) is considered for the same day (D=1) in similar fashion, as is each succeeding resource, until the last resource (R=RMAX) has been so treated for that day. At this point the values TRRQ(1)...TRRQ(RMAX), EXTEM(1)...EXTEM(RMAX) and the identifying number (CPM order of activities), K(C), of any activity which has been split are printed and these storage locations are cleared for use during the process for the succeeding day. Consideration then moves to the next day (D=2) and the entire process is repeated. The program thus proceeds until the final day (D=DAYS) has been computed. At that point the grand totals of the daily resource requirement for each resource are printed, RUNTOT(1)...RUNTOT(RMAX). Almost always some of these totals for Trial 2 will differ from those in Trial 1 hence this table is unusable except as a record of the action of the program during Trial 2. The reason for this is explained in Subsection 7, below.

4. Trial 2, Results of the Leveling Process: A table is printed at the end of the above sequence which lists the values of all the items which describe each activity. Some of these values will differ from those in the similar table printed at the outset, and these new values will disclose the shifting of activities which has been brought about by the process described in the preceding paragraph. The values included in the table are I, J, [illegible], SPLIT (if the activity is 'splittable' the number of times split is printed), TE(I), ST, FI, TL(J), TF, FF, C (the order number of the activity in the program run), K (the original CPM order number

of the activity, ascending order of I's, subascending order of J's), ITF and HOLD. As is the case with the table "Trial 2, Resource Requirements" this "Trial 2, Results of Leveling" table is not usable except as a record of the action of the program during Trial 2.

5. Trial 3, Daily Resource Requirements: This is a repeat of the operation described in Subsection 3, except that the results now are valid. The difference is caused by the fact that some selected revised values of ST are carried over from Trial 2. Further elaboration on this action is found in Subsection 7, below.

6. Trial 3, Results of the Leveling Process: This is a repeat of the operation described in Subsection 4, except that the results now are valid. (See the following subsection for further explanation.) Experimentation to the time ofwriting indicates that additional trials make no changes in the results, however, a greater variety of problems must be run before it can be stated flatly that there is no set of circumstances peculiar to a particular problem which will produce results which will be different in trials subsequent to Trial 3.

7. The action of the leveling process during Trial 2 (or Trial 3 for that matter) on a non-splittable activity frequently will proceed as follows: An excess of resource requirement, TRRQ, over resource availability, TRAV, occurs. As a result, an activity is viewed as a candidate for shifting. The scheduled start time of the activity is located more than one day to the left of the

day under consideration. The activity is found to have available enough Free Float (or Total Float as the case may be) to allow the activity to be moved toward the end of the project until the scheduled start time of the activity, ST, equals the day under consideration (remembering that work on the activity begins on the day following start time). It is readily apparent that for those prior days during which this activity had been acheduled for performance there has been included all resource requirement values, AR(C,R)...AR(C,RMAX), for this activity in the values already printed out for total resource requirements, TRRQ(1)...TRRQ(RMAX). These total figures are, therefore, in error by said amounts. To overcome this and to provide the method by which the problem solution is refined by subsequent trials the new values for ST are permanently retained, in these instances, and when the next trial proceeds the totals will be computed properly.

When a splittable activity is to be moved the same action does not take place, and start time (ST) is not revised in these instances. The portion of the activity which is scheduled for performance during days prior to the day under consideration is not moved, and the remainder of the activity is simply moved one day toward the end of the project.

The action described for the non-splittable activity in the first paragraph sometimes results in the lowering of a total resource requirement, TRRQ, for a prior day to such a level that an activity which had been moved to reduce the excess could, under the new circumstances, remain scheduled for performance during that day. If such were the case the subsequent trial would recognize this, and an activity (splittable or non-splittable) which had been removed from that day might be allowed to remain.

It also occurs frequently that a non-slittable activity is "pushed" ahead one day at a time through an area of high usage of one or more resources. This action is contrasted with that described at the outset of this Subsection 7 in that the movement is not triggered at a day past the point of beginning, hence no incorrect daily total resource requirement is left in a prior day's figures. In this case start time (ST) is not permanently changed. If, due to the permanent relocation of other non-splittable activities, "vacancies" occur, some of these latter non-splittable activities may find places closer to the beginning of the project than had been found in the earlier trial.

From the foregoing it can be seen that the scheme is one of successive trials using the identical process during each trial and differing only in that retained with the data supplied at the beginning of the later trials are all starting times (ST) changed by the action described in the first paragraph of this Subsection 7. This is accomplished by the use of the value "HOLD" (HOLD = TL(J) - ST) for each activity. Also, any new TE(J) value affected by such a change is likewise retained by the use of the value "ITF" (ITF = TL(J) - TE(J)) for each activity. All other activities are restored to their original position, all-early-start, and all splits are removed at the beginning of each new trial.

The results of the leveling run can be put in bar chart form readily for general use and for further testing as to the validity and practicality of the new arrangement of activities.

IMPERFECTIONS OF THE PROGRAM

Brief mention should be made of some of the more obvious imperfections or fallacies of the above-described program.

1. Whatever the order selected for the activities in a leveling run a bias is introduced, but the bias itself is changing and unpredictable. For example:

   a. Using CPM order (ascending order of I's, subascending order of J's) as representative of the order in which activities will be performed. - The assumption may bear little relation to truth. Perhaps late start order is more appropriate.

   b. Using ascending order of Total Float so as to make a limited movement of an activity (within float boundaries) more productive. - The Free Floats will not necessarily fall in the same order. More important, after some floating has taken place the order will not necessarily remain ascending. The same would be true if descending order of total float were initially used to order the activities.

   c. Placing activities with the highest demands on resources at the top of the list. - If only slight reductions are needed the bias will tend to produce larger-than-necessary corrections. If the reverse of this order is selected too many activities may be relocated before the desired reduction is affected.

2. Relocating an activity without regard to the area into which it is placed is, to some at least, not a proper treatment of the problem. Lack of sufficient computer storage is the chief deterrent to taking this factor into account.

3. Relocating an activity strictly on the basis of an excess of requirement over availability for one resource considered alone is a scheme open to criticism. Suppose there is a similar excess for another resource during that day. Would not it be better to be more selective in choosing the activity to be removed? Perhaps an activity could be found which contains the proper combination of resource requirements, and which, if moved, would solve that day's problem, whereas the program as now written might move two or more activities to accomplish the same end.

CONCLUSIONS

The CPM study for a construction project, while extremely valuable, by itself is not really complete without a companion leveling study and certainly will be more meaningful if this latter study is made.

The program offered here is proposed as a means by which the construction industry can "learn by doing." Emphasis is placed on the concept that neither the leveling solution, nor the CPM study to which it is companion, should become static or unchangeable but both should remain open for review in the light of new insights as they become available. The theory is set forth that a leveling program of itself, should not lengthen a project and that the concepts of event time boundaries, float and critical path should not be abandoned when entering into the leveling phase of a problem solution.

In keeping with the theory that the user should be provided as much control as possible over his problem solution the program requires that he decide such things as; (1) the "splittable" character of an activity; (2) the selection of the resources which are to be considered; (3) the job-wise priority of resources; (4) the activity resource requirements; (5) the availability of resources; and (6) the order in which activities are to be considered for shifting.

APPENDIX---NOTATION

The following symbols have been adopted for use in this paper:

A = Total number of activities in a given problem;

AR = One-day requirement of an activity for a resource;

B = Number used in iteration of resource availability data;

C = Common subscript of variables related to a given activity;

D = Day number;

DAYS = Normal duration of project (from CPM solution);

DIDSPL = Indicator that an activity has been split on a given day and therefore cannot be split again;

EXCESS = Difference between a resource requirement and the corresponding resource availability for a given day;

EXTEM = Computed as is EXCESS but stored temporarily;

FF	=	Free Float;
FINISH	=	Last day of a level of resource availability, used as a subscript for TRAV;
FT	=	Finish Time of an activity;
HOLD	=	TL(J) - ST;
I	=	Number of node at the tail of an arrow;
ITF	=	Interfering Float;
J	=	Number of node at the head of an arrow;
K	=	Number of activity when in CPM order;
R	=	Number of resource in project priority order;
RCD	=	The total number of resource availability data cards;
RMAX	=	The total number of resources;
RSR	=	Alphabetic storage of resource description;
RUNTOT	=	Cumulative total of daily resource requirements for a resource;
S	=	Iteration subscript for SPOT;
SPLIT	=	Code for splitability and number of days split;
SPOT	=	Day upon which an activity was split;
START	=	Day preceding the first day of a level of resource availability, used as an subscript for TRAV;
ST	=	Start Time of an activity;
TE	=	Earliest event time of a node;
TEITEM	=	Temporary storage, earliest event time of an I-node;
TF	=	Total Float;
TL	=	Latest event time of a node;
TLJTEM	=	Temporary storage, latest event time of a J-node;

TRAV = Total availability of a resource for a given day;

TRIALS = Maximum number of trials for a problem run;

TRRQ = Total requirement of a resource for a given day;

TRVTEM = Temporary storage, total availability of a resource;

W = Number used in iteration of resource requirement;

Y = Normal duration of an activity.

TABLE 1

LIST OF PROJECT ACTIVITIES

ACT.	ACTIVITY DESCRIPTION	No. OF FITTR'S	No. OF DAYS
	INSTALL MAIN STEAM PIPING SYSTEM :		
A	ERECT PIPE : BOILER TO COLD - SPRING WELD	8	10
B	ERECT PIPE : THROTTLE VALVE TO C-S WELD	8	15
C	INSTALL HANGERS	5	7
D	COLD SPRING PIPE , MAKE FINAL WELD	6	3
E	PERFORM HYDROSTATIC TEST	4	2

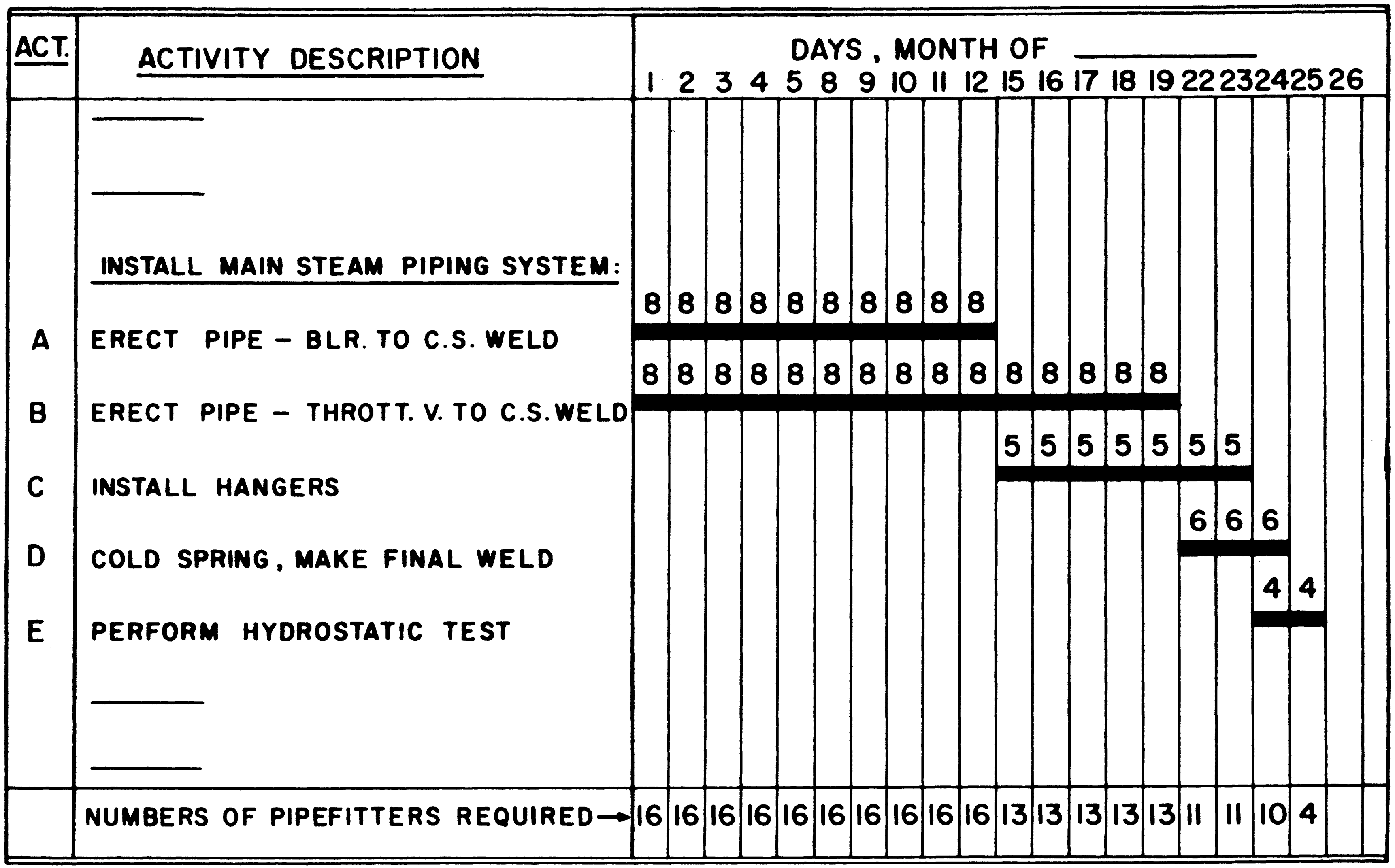

Figure 1. Conventional Bar Chart (Single Resource).

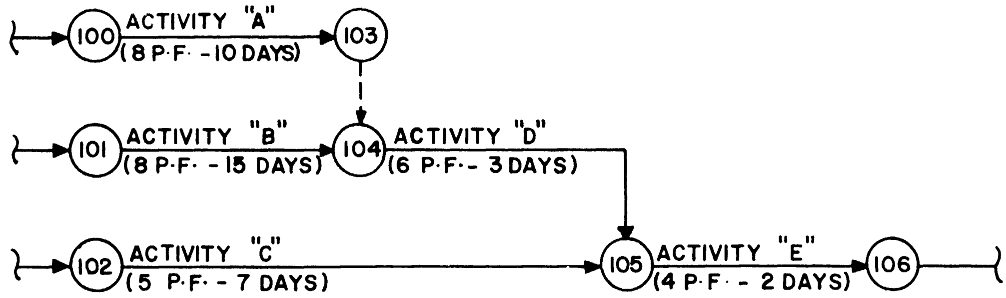

Figure 2. Arrow Diagram

TABLE 2

VALUES RETURNED BY CPM PROGRAM

I	J	TE(I)	ST	Y	FT	TL(J)	FF	TF	ACT·
100	103	202	202	10	212	220	0	8	A
101	104	200	200	15	215	220	0	5	B
→ 102	105	205	205	7	212	223	6	11	C ←
103	104	212	212	0	212	220	3	8	D
104	105	215	215	3	218	223	0	5	E
105	106	218	218	2	220	225	0	5	F

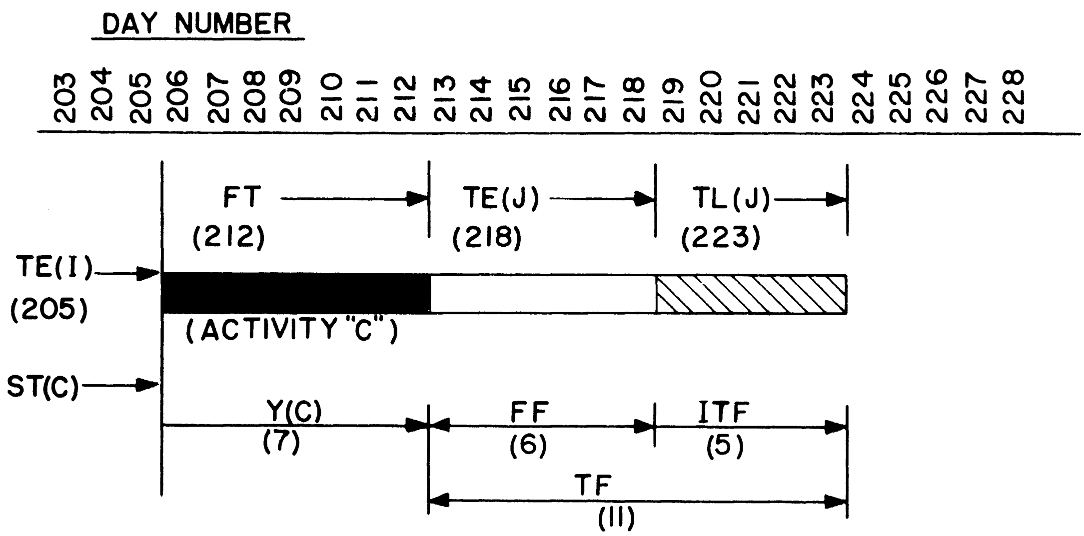

Figure 4. CPM Activity Bar.

= ACTIVITY DURATION (Y); = TOTAL FLOAT (TF); = FREE FLOAT (FF); = INTERFERING FLOAT (ITF)

I	J	TE(I)	ST	Y	FT	TL(J)	201	202	203	204	205	206	207	208	209	210	211	212	213	214	215	216	217	218	219	220	221	222	223	224	225
100	103	202	202	10	212	220			8	8	8	8	8	8	8	8	8	8													
101	104	200	200	15	215	220	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8										
102	105	205	205	7	212	223						5	5	5	5	5	5	5													
103	104	212	212	0	212	220												0													
104	105	215	215	3	218	223																6	6	6							
105	106	218	218	2	220	225																			4	4					
NUMBERS OF PIPEFITTERS REQUIRED→							8	8	16	16	16	21	21	21	21	21	21	21	8	8	8	6	6	6	4	4	0	0	0	0	0

(ALL ACTIVITIES AT EARLY START POSITION)

Figure 3a. Bar Chart, Leveling Program Trial No. 1 - CPM Output Activity Location (Single Resource).

I	J	TE(I)	ST	Y	FT	TL(J)	201	202	203	204	205	206	207	208	209	210	211	212	213	214	215	216	217	218	219	220	221	222	223	224	225
100	103	202	202	10	212	220			8	8	8	8	8	8	8	8	8	8													
101	104	200	200	15	215	220	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8										
102	105	205	212	7	219	223													5	5	5	5	5	5	5						
103	104	212	212	0	212	220												0													
104	105	215	215	3	218	223																6	6	6							
105	106	218	218	2	220	225																			4	4					
NUMBERS OF PIPEFITTERS REQUIRED→							8	8	16	16	16	16	16	16	16	16	16	16	13	13	13	11	11	11	9	4	0	0	0	0	0

TE(I)
ST

Figure 3b. Bar Chart, Leveling Program Trial No. 3 - Floated Activity Location (Single Resource).

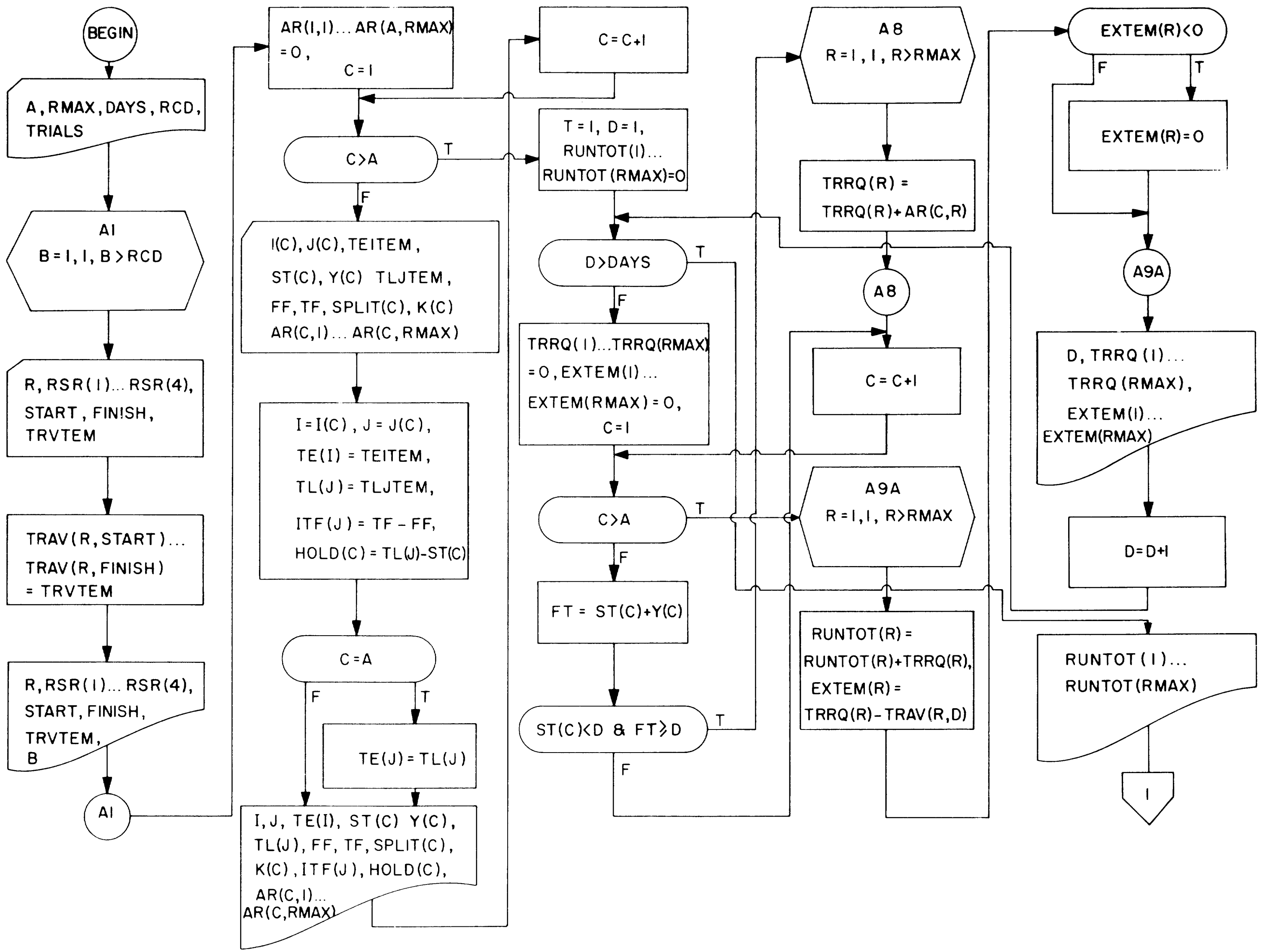

Figure 5a. Flow Chart, Resource Leveling Program. (Cont'd in Fig. 5b).

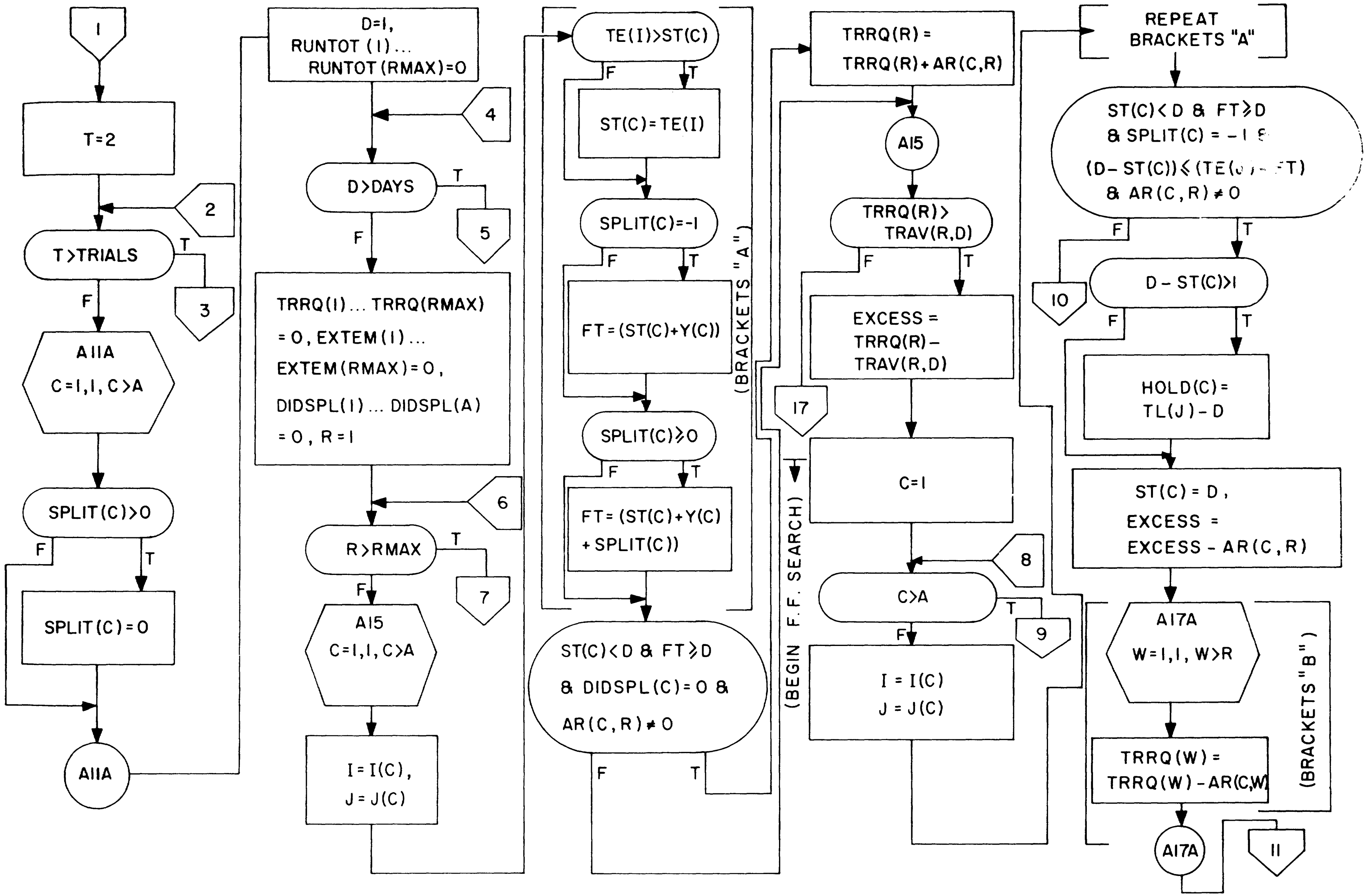

Figure 5b. (Cont'd.)

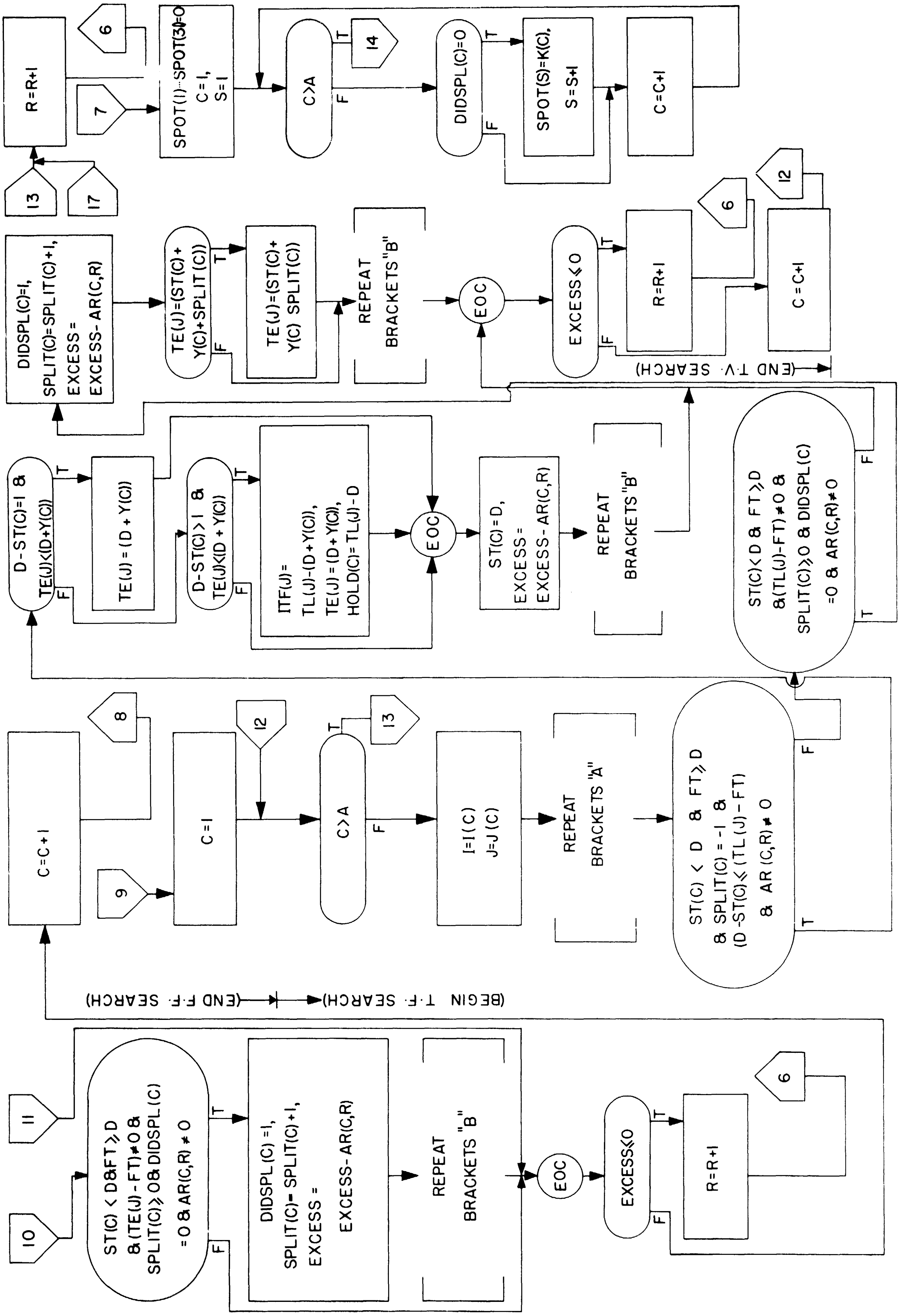

Figure 5c. (Continued).

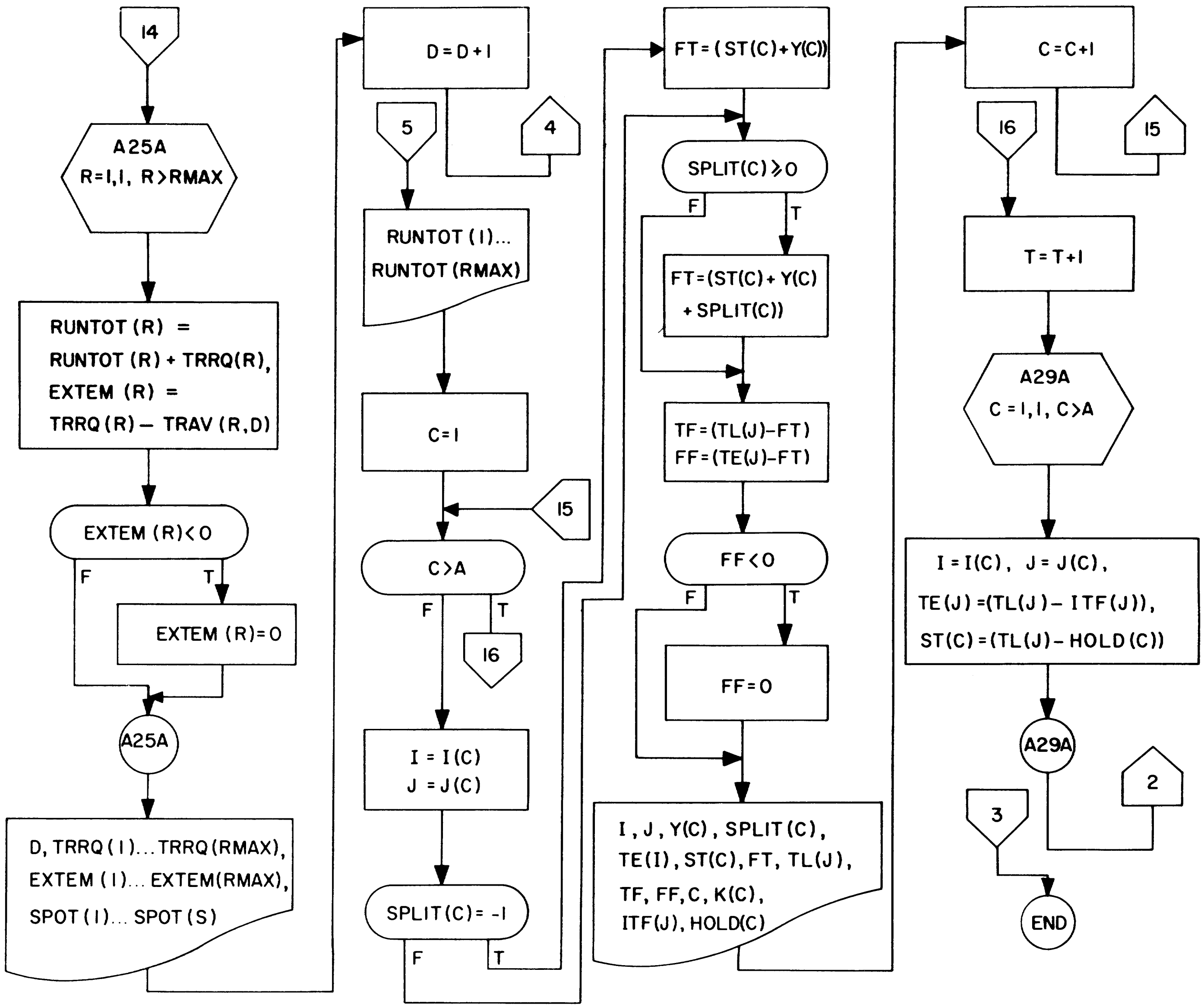

Figure 5d. (Continued.)