OAK RIDGE NATIONAL LABORATORY
operated by
UNION CARBIDE CORPORATION
for the
U.S. ATOMIC ENERGY COMMISSION

ORNL-TM-394

COPY NO. - 26

DATE - August 30, 1962

CRITICAL PATH SCHEDULING BY THE LINEAR PROGRAMMING METHOD

By

Royes Salmon

ABSTRACT

A method is presented for extending the scope of critical path analyses so that the minimization of over-all project cost becomes the primary objective, rather than the minimization of total project duration. Any desired limit can be placed on the over-all duration of the project, and any desired dollar value can be placed on over-all project time, so that project time can be included as one of the cost considerations making up the total cost of the project. The method is based on linear programming, with time as the fundamental variable. Each critical path problem is converted into an equivalent linear programming problem. This permits the inclusion of time-cost relationships. The solution of the linear programming problem gives the optimum project schedule, and also indicates which jobs are critical.

Costs of individual jobs may be handled in the form of time-cost curves, which are approximated piecewise by straight lines.

Existing IBM-7090 linear programming codes are adequate for handling networks of up to 500 nodes.

The procedure is illustrated by the solution of nine sample problems, the results of which adequately substantiate the above conclusions.

NOTICE

This document contains information of a preliminary nature and was prepared primarily for internal use at the Oak Ridge National Laboratory. It is subject to revision or correction and therefore does not represent a final report. The information is not to be abstracted, reprinted or otherwise given public dissemination without the approval of the ORNL patent branch, Legal and Information Control Department.

LEGAL NOTICE

This report was prepared as an account of Government sponsored work. Neither the United States, nor the Commission, nor any person acting on behalf of the Commission:

A. Makes any warranty or representation, expressed or implied, with respect to the accuracy, completeness, or usefulness of the information contained in this report, or that the use of any information, apparatus, method, or process disclosed in this report may not infringe privately owned rights; or

B. Assumes any liabilities with respect to the use of, or for damages resulting from the use of any information, apparatus, method, or process disclosed in this report.

As used in the above, "person acting on behalf of the Commission" includes any employee or contractor of the Commission, or employee of such contractor, to the extent that such employee or contractor of the Commission, or employee of such contractor prepares, disseminates, or provides access to, any information pursuant to his employment or contract with the Commission, or his employment with such contractor.

CONTENTS

		Page
1.0	INTRODUCTION	3
2.0	LINEAR PROGRAMMING	3
	2.1 Restraints	3
	2.2 Conversion of a Critical Path Problem to a Linear Programming Problem	4
	2.3 Maximum Duration	5
	2.4 Job Costs	5
3.0	SAMPLE PROBLEM M-100	7
	3.1 Job Costs	9
	3.2 Solving the Linear Programming Problem	10
4.0	PROBLEM M-101	10
5.0	PROBLEM M-102	12
6.0	PROBLEM M-103	13
7.0	PROBLEM M-200	15
8.0	PROBLEM M-201	15
9.0	PROBLEM M-300	16
10.0	PROBLEM M-301	17
11.0	PROBLEM M-400	20
12.0	REFERENCES	23
13.0	APPENDIX	24
	13.1 Project Network Diagrams, Job Durations, and Cost Slopes	24
	13.2 Input Cards	36
	13.3 Solution Output Tapes	57

1.0 INTRODUCTION

Critical path scheduling is a method for finding the fastest way of accomplishing a project which consists of many individual jobs. The jobs are interconnected according to certain sequence requirements, so that certain jobs must be done before others can be started; on the other hand, many jobs can proceed concurrently. Thus the jobs form an interlocking network; the points at which the jobs join are called "nodes." The object of the analysis is to find the shortest way to accomplish all the jobs, thus minimizing the time required to complete the project.

Existing critical path methods are adequate for determining over-all project duration and determining which of the component jobs are critical. Less attention, however, has been given to methods by which the minimization of over-all project cost can be made the primary objective of the calculation. The purpose of this study was to develop such a method.

2.0 LINEAR PROGRAMMING

Many physical systems can be expressed in the form of n linear equations involving more than n variables. The linear programming problem is to find the one solution, out of the infinite number of possible solutions, which minimizes (or maximizes) some predetermined linear function of the variables. The function being minimized might define, for example, the total cost of an operation; an example of a function which might be maximized is the total profit from the operation.

2.1 Restraints

Restraints or restrictions are relationships representing the physical limitations upon the variables.

Normally the restraints are in the form of linear inequalities. These can be converted to equalities by adding slack variables as necessary. For example

$$X_1 + X_2 \leq 10$$

becomes

$$X_1 + X_2 + S_2 = 10$$

where S_2 is some nonnegative number. It is a characteristic of all linear programming problems that no variable may take on a negative value. Thus, the equation limits the sum of X_1 and X_2 to not more than 10, and also limits each of them individually to not more than 10.

2.2 Conversion of a Critical Path Problem to a Linear Programming Problem

Critical path scheduling applies to networks of jobs which obey certain sequencing and duration requirements. Its objective is to find the minimum time necessary to accomplish all the jobs. This problem can be solved by a simple arithmetic procedure which has been presented in the literature and is now widely known[1].

In a scheduling problem, the individual jobs join at points called "nodes." The times at which these points are reached can be considered as variables. The project starts at node 1, which is assigned the time zero. Then we let

t_2 = time at which point 2 is reached

t_3 = time at which point 3 is reached

.
.
.

t_n = time at which the project is complete (that is, point n is reached).

Thus, for a project involving n nodes, we have n - 1 variables. The restraints on these time variables may now be written in the form of linear inequalities. These arise from the sequencing and duration restrictions.

Letting $t_1 = 0$, the sequence restrictions may be expressed as

$$t_2 \geq t_1$$

$$t_3 \geq t_2$$

$$t_4 \geq t_3$$

$$\vdots$$

$$t_n \geq t_{n-1}$$

The fact that each job has a certain minimum duration may also be expressed in the form of linear inequalities. For example, if job 1,2 has a minimum duration of four days, we may write

$$t_2 - t_1 \geq 4$$

And if job 2,3 has a minimum duration of five days, we may write

$$t_3 - t_2 \geq 5$$

A similar inequality may be written for each job.

Now we observe that these duration restrictions can be used to replace the sequencing restrictions; that is, the latter are no longer necessary, as they are implicit in the duration restrictions. For example, comparing

$$t_2 \geq t_1 \text{ and } t_2 - t_1 \geq 4$$

we see that, if the latter is true, the former also must be true. Thus the duration restrictions form the entire set of restrictions on the project. It follows that the number of restrictions is equal to the number of jobs in the project.

In linear programming, it is a great advantage to decrease the number of restrictions as much as possible; the amount of computing time and the memory capacity required increase rapidly as the number of restrictions is increased. Thus the elimination of the sequencing restrictions is well worthwhile.

2.3 Maximum Duration

If we wish to place a limit of 50 days maximum duration on the total project, we write

$$t_n \leq 50$$

2.4 Job Costs

A job which takes a certain amount of time to complete at normal speed can usually be done in less time at a greater cost. For example, overtime wages might be paid. In order to approach this problem simply, suppose that the job cost is related linearly to the job duration, in such a way that decreasing duration means increasing cost (Fig. 1). An actual job cost curve, of course, might be considerably more complicated (Fig. 2). Non-linear job cost curves such as this may be handled by piecewise linear approximation; that is, several straight lines are drawn which roughly approximate the curved line. The approximation can be made as close as desired by increasing the number of time segments. This is discussed in detail later in this report.

Returning to the simplified case shown in Fig. 1, if the slope of this line is -0.40 dollars per day for job 1,2, then the cost of this job may be expressed

$$C_{1,2} = A_{1,2} - 0.40(t_2 - t_1)$$

where the constant $A_{1,2}$ represents the intercept on the vertical axis.

The over-all cost of the project is the sum of the individual job costs, each job having its individual time-cost slope. Thus the over-all project cost is a linear

function of the variables t_1, t_2, ... t_n. The problem now is to find values of t_1, t_2, ... t_n which minimize the total project cost. The constants $A_{1,2}$, $A_{2,3}$, etc., are irrelevant and do not enter into the problem, since they cannot affect the minimization process involving the variables t_1, t_2, ... t_n.

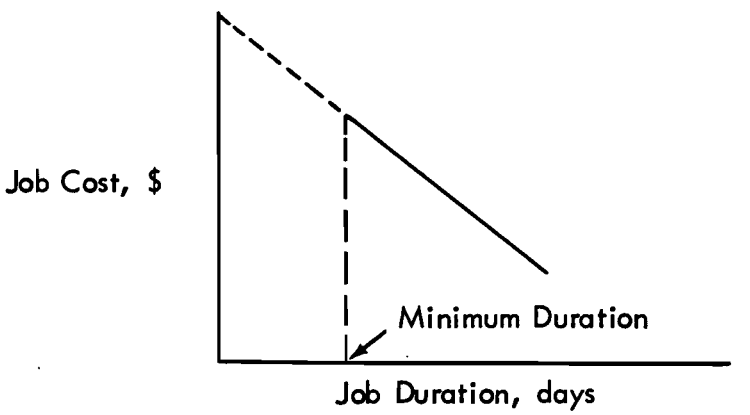


Fig. 1. Job cost curve.

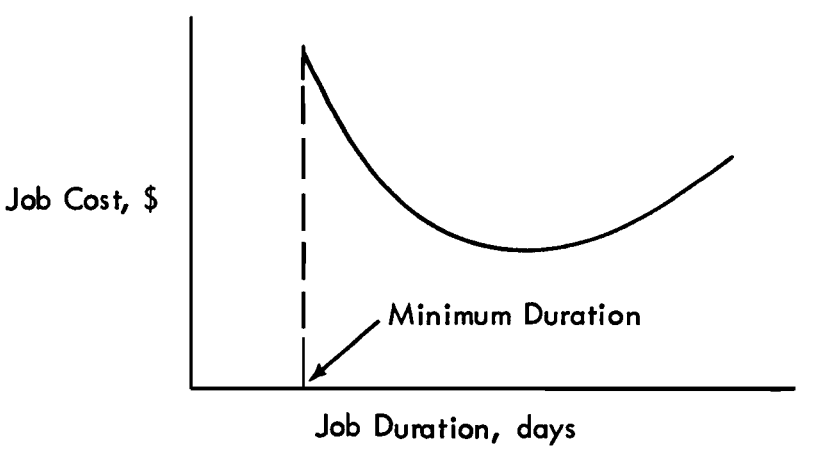


Fig. 2. Actual job cost curve.

3.0 SAMPLE PROBLEM M-100

Sample problem M-100 was designed to demonstrate the basic procedure. There are nine nodes and sixteen jobs in the network (Fig. 3.).

The duration and sequencing restrictions for this project were written as follows:

$t_1 = 0$ (by definition)

$t_2 - t_1 \geq 4$	$t_5 - t_4 \geq 7$
$t_3 - t_1 \geq 9$	$t_8 - t_4 \geq 11$
$t_3 - t_2 \geq 5$	$t_6 - t_5 \geq 6$
$t_4 - t_2 \geq 4$	$t_7 - t_5 \geq 12$
$t_5 - t_2 \geq 2$	$t_8 - t_5 \geq 7$
$t_6 - t_2 \geq 8$	$t_7 - t_6 \geq 3$
$t_4 - t_3 \geq 1$	$t_8 - t_7 \geq 6$
$t_8 - t_3 \geq 4$	$t_9 - t_8 \geq 5$

In addition, an upper limit of 50 days was placed on the over-all duration of the project:

$$t_9 \leq 50$$

The restrictions were converted to equalities by adding nonnegative slack variables as previously indicated. The new set of restrictions is

$$-t_2 + u_1 = -4$$

$$-t_3 + u_2 = -9$$

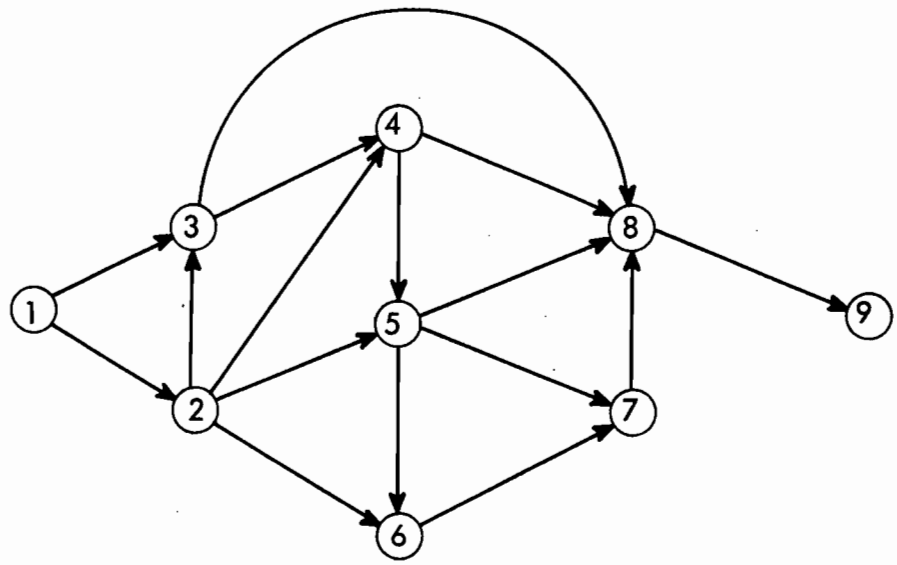
$$+t_2 - t_3 + u_3 = -5$$

$$+t_2 - t_4 + u_4 = -4$$

$$+t_2 - t_5 + u_5 = -2$$

$$+t_2 - t_6 + u_6 = -8$$

$$+t_3 - t_4 + u_7 = -1$$



Job	Minimum Duration, days	Cost Slope, $	Job	Minimum Duration, days	Cost Slope, $
1,2	4	-0.40	4,5	7	-0.60
1,3	9	-0.20	4,8	11	-0.10
2,3	5	-0.30	5,6	6	-0.30
2,4	4	0	5,7	12	-0.30
2,5	2	-0.40	5,8	7	0
2,6	8	-0.20	6,7	3	-0.40
3,4	1	-0.10	7,8	6	-0.20
3,8	4	0	8,9	5	-0.10

Fig. 3. Sample Problem M-100.

$$+t_3 - t_8 + u_8 = -4$$

$$+t_4 - t_5 + u_9 = -7$$

$$+t_4 - t_8 + u_{10} = -11$$

$$+t_5 - t_6 + u_{11} = -6$$

$$+t_5 - t_7 + u_{12} = -12$$

$$+t_5 - t_8 + u_{13} = -7$$

$$+t_6 - t_7 + u_{14} = -3$$

$$+t_7 - t_8 + u_{15} = -6$$

$$+t_8 - t_9 + u_{16} = -5$$

3.1 Job Costs

The job costs were expressed as follows:

$$C_{1,2} = A_{1,2} - 0.40(t_2 - t_1)$$

$$C_{1,3} = A_{1,3} - 0.20(t_3 - t_1)$$

$$C_{2,3} = A_{2,3} - 0.30(t_3 - t_2)$$

$$\vdots$$

$$C_{8,9} = A_{8,9} - 0.10(t_9 - t_8)$$

Writing out and adding up all these gives the total cost

$$C = K + 0.50t_2 - 0.40t_3 + 0.60t_4 - 0.40t_5 - 0.10t_6 - 0.50t_7 - 0.20t_8 - 0.10t_9$$

where K represents the sum of the A's.

One additional cost factor was added: a penalty of \$10 per day on the overall duration of the project. This was done in order to force the solution toward minimum project duration, to demonstrate that this would give the same result as an ordinary critical path analysis. Adding $+10\,t_9$ to the cost equation above gives

$$C = K + 0.50t_2 - 0.40t_3 + 0.60t_4 - 0.40t_5 - 0.10t_6 - 0.50t_7 - 0.20t_8 + 9.90t_9$$

This total cost function C is the function that is to be minimized during the linear programming calculation; as mentioned before, the constant K has no bearing on the values of the t's, and was omitted.

The operation of putting the problem into linear programming form has now been completed. The variables have been assigned, the restrictions have been written, and the cost function has been defined in terms of the variables.

3.2 Solving the Linear Programming Problem

Linear programming problems such as this can be handled by the SCROL code on the IBM-7090 computer[2]. The input data are the identification symbols for the variables, the restriction equations, and the cost function. The program proceeds automatically to minimize the cost function. The solution gives the values of the t's that accomplish this, together with the minimal value of the cost function. The values of the t's constitute the optimum job schedule. The question of which jobs are critical can be readily answered by comparing the t's and the minimum job durations; all jobs that must be done in their minimum duration are critical. All noncritical jobs that have negative cost slopes will be stretched out to their maximum duration, since this reduces over-all cost.

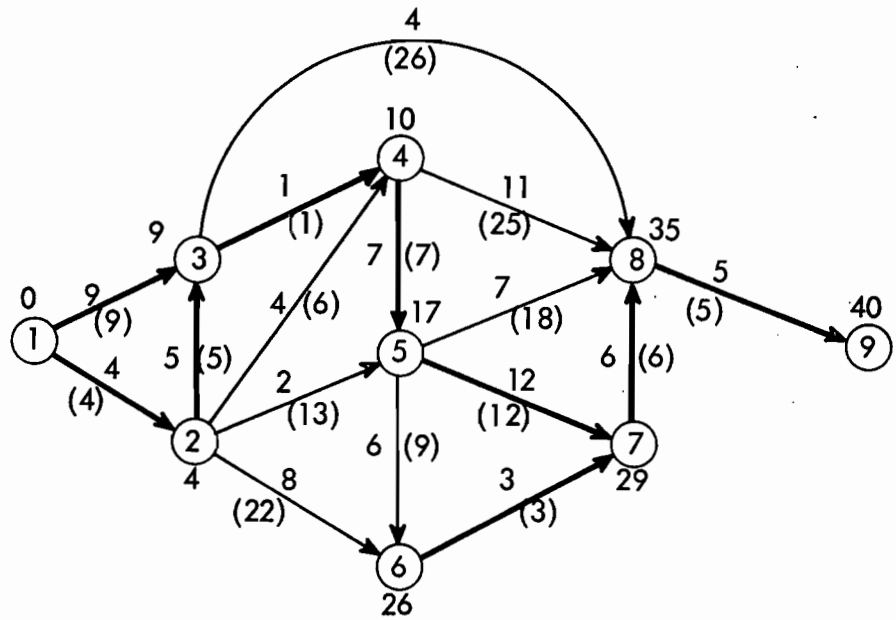
The input data for problem M-100 were set up on IBM cards as required by the SCROL code. Ninety cards were needed. The input cards are listed completely in Appendix 13.2. The solution output for this problem is given in Appendix 13.3, with certain unnecessary portions of the output omitted.

The solution (Fig. 4) shows the critical path and the time at which each node is reached. The over-all project duration was 40 days, and the total cost was $369.50. A solution by hand, using the ordinary critical path method, showed a minimum project time of 40 days; this solution agreed in every respect with the computer solution, showing that the minimum cost schedule in this case corresponds to the minimum over-all time. This is a consequence of the large penalty ($10 per day) placed on over-all project time.

4.0 PROBLEM M-101

All the other sample problems are variations based on the nine-node network used in M-100. M-101 was run on the computer as an addendum to, and together with, problem M-100, using the feature of the SCROL code that permits the inclusion of additional right-hand sides. The term "right-hand side," in linear programming, refers to the constant term in the linear equation, and is so called because of its location when the equation is written in the normal form:

$$a_1x_1 + a_2x_2 + \dots + a_nx_n = b_1$$



Numbers on nodes indicate time in days. Numbers on lines indicate minimum durations of jobs Numbers in parentheses indicate actual durations. Critical jobs are shown by heavy lines. These jobs must be done in their minimum duration, in order to minimize over-all cost.

<u>Times at nodes:</u>

$t_1 = 0$	$t_6 = 26$
$t_2 = 4$	$t_7 = 29$
$t_3 = 9$	$t_8 = 35$
$t_4 = 10$	$t_9 = 40$
$t_5 = 17$	

Fig. 4. Solution of Problem M-100.

The linear programming matrix consists of a number of equations, each having its b, and the vector which consists of all these b's is the complete right-hand side of the matrix.

The SCROL code permits different right-hand side vectors to be inserted at will after the first optimal solution has been reached. In a scheduling problem, these right-hand sides represent the minimum job durations. To test this feature of the code, therefore, all that was necessary was to change the minimum duration of one or more of the jobs. Job 2,6 was chosen for this purpose; its minimum duration was changed from 8 to 28 days, all other items being unchanged.

The result of this modification was to develop a new critical path, with an over-all time of 46 days. This again agreed with the minimum time found by the critical path algorithm. The solution network is shown in Fig. 5.

The SCROL code permits any number of additional right-hand sides to be inserted after the main problem. This was utilized in problem M-400, where eight different right-hand sides were added. The advantage of this feature is that the additional solutions are reached much more rapidly than if the problems are run individually.

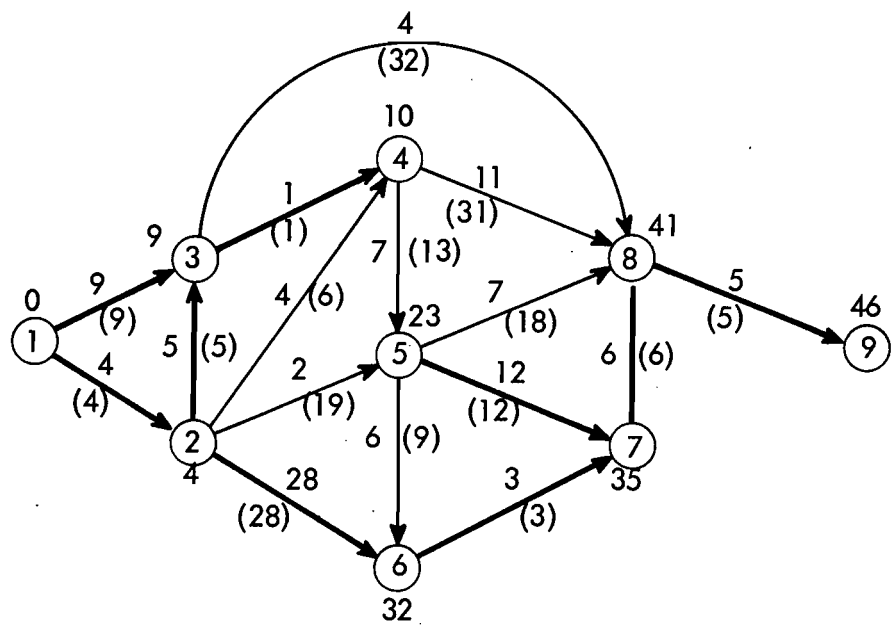
5.0 PROBLEM M-102

In this problem, the cost penalty on over-all project time was reduced from $10 per day to $0.50 per day. With this change, the program was able to effect a net saving by stretching out the project to its maximum duration of 50 days, because the amount saved by stretching out the individual jobs outweighed the penalty on project time. The solution diagram is shown in Fig. 6. All individual jobs with negative cost slopes were stretched out to their maximum duration. The durations of jobs whose cost slopes were zero were not affected one way or the other; these jobs can have any duration up to the maximum permitted by the 50-day over-all limitation.

This problem therefore demonstrates that the minimum cost schedule for a project may be quite different from the minimum time schedule. Here, for the first time, the procedure gives a result different from that of an ordinary critical path analysis. Further, there is no longer any "critical path," in the sense of a continuous chain of critical jobs from the first node to the last; a noncritical gap occurs in the chain. This is a consequence of working toward minimum cost rather than minimum time. Within this gap a reduction in time would actually increase cost, as may be easily verified.

6.0 PROBLEM M-103

As before, an additional right-hand side was inserted following M-102, changing the minimum duration of job 2,6 to 28 days. In this case, however, the solution was identical with that of M-102, since the latter showed job 2,6 stretched out to 32 days; thus imposing a minimum of 28 days had no effect.

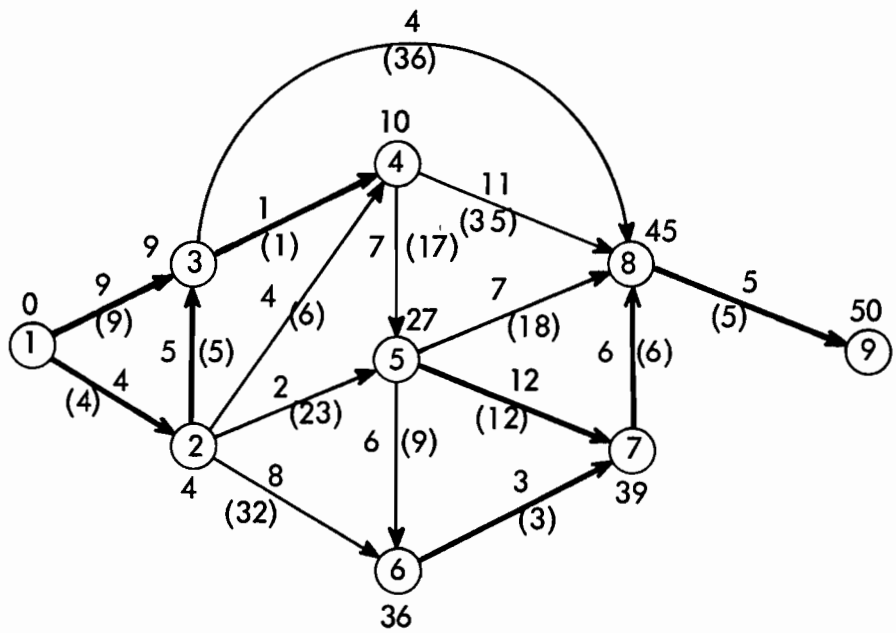


Numbers on lines indicate minimum durations of jobs. Numbers in parentheses indicate actual durations. Critical jobs are shown by heavy lines. These jobs must be done in their minimum duration, in order to minimize over-all cost.

<u>Times at nodes:</u>

$t_1 = 0$	$t_6 = 32$
$t_2 = 4$	$t_7 = 35$
$t_3 = 9$	$t_8 = 41$
$t_4 = 10$	$t_9 = 46$
$t_5 = 23$	

Fig. 5. Solution of Problem M-101.



Numbers on lines indicate minimum durations of jobs. Numbers in parentheses indicate actual durations. Critical jobs are shown by heavy lines. These jobs must be done in their minimum duration, in order to minimize over-all cost.

<u>Times at nodes:</u>

$t_1 = 0$	$t_6 = 36$
$t_2 = 4$	$t_7 = 39$
$t_3 = 9$	$t_8 = 45$
$t_4 = 10$	$t_9 = 50$
$t_5 = 27$	

Fig. 6. Solution of Problem M-102 (also applies to M-103 and M-201).

7.0 PROBLEM M-200

The dual formulation of the linear programming matrix was used with this problem. The dual formulation converts the original set of restrictions into a new set in which the coefficients are obtained by reading down the columns of the original restrictions. Since the linear programming matrix is a tabulation of these coefficients, the new matrix is the old matrix rotated 90°; that is, the rows become columns, and the columns, rows. The original right-hand side becomes the new cost row, and the original cost row becomes the new right-hand side. Also, the sense of all the original inequalities is reversed. Surprisingly, the solution of the dual form gives the same minimum cost as the solution of the primal form. The solution also indicates the same optimal job schedule. The proof of this is found in standard references on linear programming[3].

The advantage of the dual form is that it frequently permits a considerable reduction in the number of restrictions in the matrix. Thus a problem like M-100, which had 17 inequalities and 8 variables, can be converted to one having 8 inequalities and 17 variables. The reduction in the number of inequalities permits a considerable saving in computer time. The increase in the number of variables has relatively little effect.

Of equal importance from a practical point of view is that the size of the maximum problem that can be handled by the SCROL code is limited to 511 equations. The limit on the number of variables is 6000. A project consisting of 500 nodes and 2000 jobs would have about 2000 equations in the primal form, thus exceeding the capacity of the computer. In the dual form, however, it would have only 500 equations, which is within the capacity limitation. This is the reason for the statement made earlier in this report that projects having up to about 500 nodes can be handled by the code.

The solution of M-200 was identical with that of M-100, thus demonstrating the validity of the dual procedure.

8.0 PROBLEM M-201

This was run as a second right-hand side of M-200. In the dual form, the right-hand side corresponds to the original cost row, so the change made in going from M-200 to M-201 was a change in a cost coefficient rather than a change in job duration. The cost coefficient changed was the one giving the penalty on overall project time. By changing this from $10 to $0.50, the problem was made identical with M-102, thus providing a means of checking the solution. The solution of M-201 was, in fact, identical with that of M-102, thus validating the result.

Thus converting a problem to the dual form permits the investigation of various cost coefficients by means of added right-hand sides. On the other hand, if the primary interest is in varying job durations, the primal form will be more convenient.

9.0 PROBLEM M-300

The next step was the introduction of nonlinear job cost curves. In M-300 a three-piece cost curve was constructed for job 7,8. In all other respects, the problem was identical with M-201. The three-piece cost curve for job 7,8 is shown in Fig. 7.

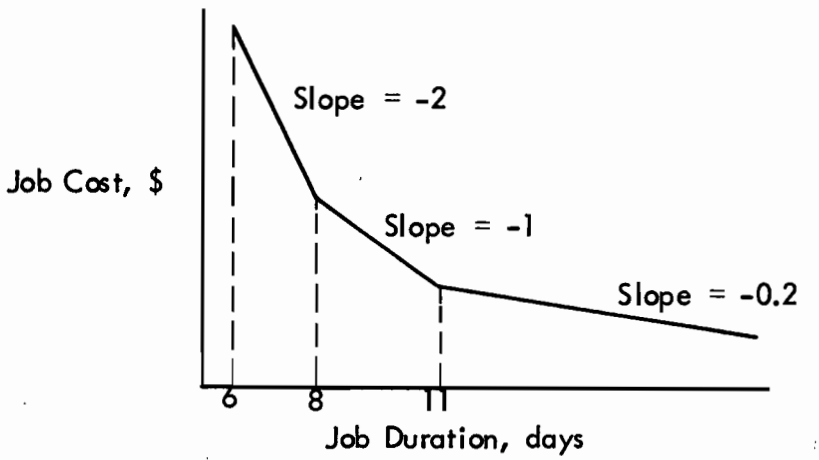


Fig. 7. Cost curve for Job 7,8.

In practice, the job cost curve might be a smoothly curving line as in Fig. 2; the piecewise-linear representation shown here is supposed to be an approximation to such an actual curve. By increasing the number of straight-line segments, the approximation can be made as close as desired.

Each straight-line segment has the effect of introducing one additional node into the network diagram. The additional nodes for job 7,8 were numbered 10 and 11, thus making it possible to represent job 7,8 as shown in Fig. 8.

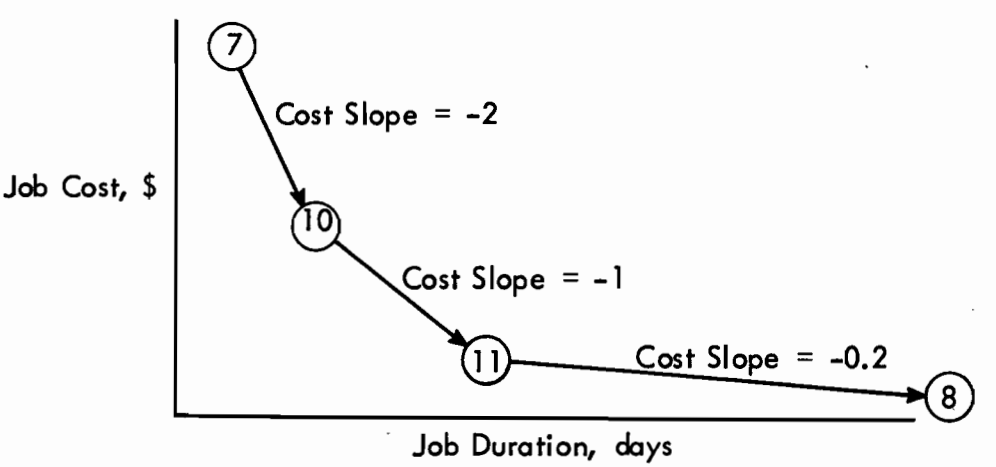


Fig. 8. Piecewise representation of Job 7,8.

Job 7,8 has thus been broken down into three jobs: job 7,10, job 10,11, and job 11,8. These are handled like ordinary jobs of the network, using the same basic procedure as in M-100. The minimum durations, however, must be chosen with care. The minimum for job 7,10 is 6 days; but jobs 10,11 and 11,8 have minimums of zero, because the whole job 7,8 could be done in 6 days.

In addition to minimum duration restrictions, there must be maximum duration restrictions placed on each segment except the last. Job 7,10 has a maximum of 8 days because if it went to 9 days it would enter segment 10,11. In the same way, job 10,11 has a maximum length of 3 days (11 minus 8). Job 11,8 has no maximum.

The inequalities expressing these restrictions are

$$t_{10} - t_7 \geq 6$$

$$t_{11} - t_{10} \geq 0$$

$$t_8 - t_{11} \geq 0$$

$$t_{10} - t_7 \leq 8$$

$$t_{11} - t_{10} \leq 3$$

These restrictions were added to those of problem M-201 to obtain the complete set of restrictions for M-300. One restriction was eliminated, the one expressing the minimum duration of job 7,8, which has been replaced by those above.

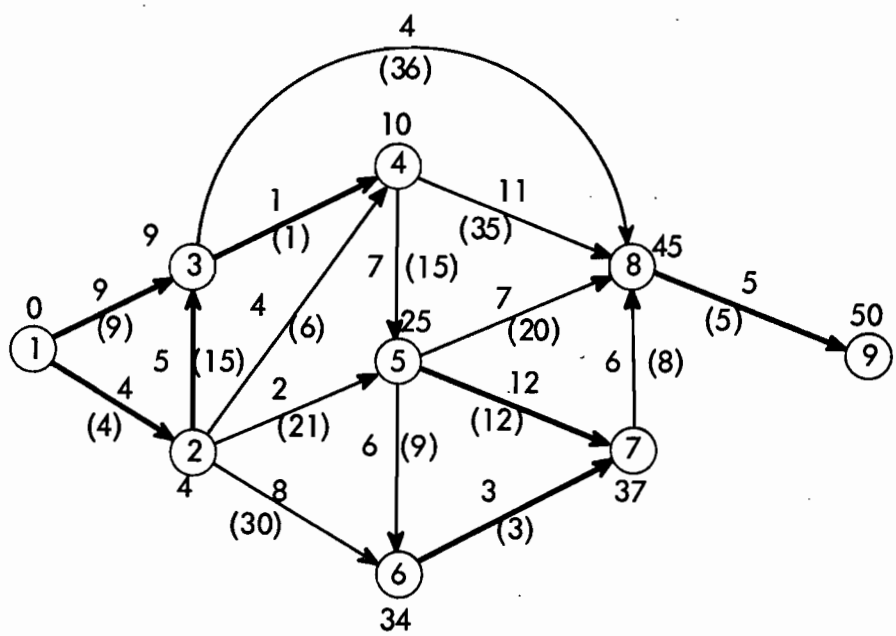
The cost coefficients of the variables were calculated in the usual manner, using the cost slopes shown for the new jobs. This problem was run in the dual form. The solution is shown in Fig. 9.

The differences in the solutions of M-201 (same as M-102) and M-300 are due to the different cost slopes introduced in job 7,8. A duration of 8 days was found for this job; this avoids the rapid increase of cost in the 6- to 8-day duration region. In the solution of M-201, the duration of job 7,8 was 6 days; but the slope of the cost curve in this case was only -0.20.

Thus the program is able to discriminate between slopes and choose the optimum duration for each job. If the cost curve for a job is nonlinear, it may be approximated by straight lines to any desired degree of refinement.

10.0 PROBLEM M-301

M-301 was an additional right-hand side on M-300; it had a cost penalty on project time of 1.50 instead of 0.50. This increase in cost penalty was sufficient to cause a reduction in over-all time from 50 to 42 days. The solution is shown in Fig. 10.

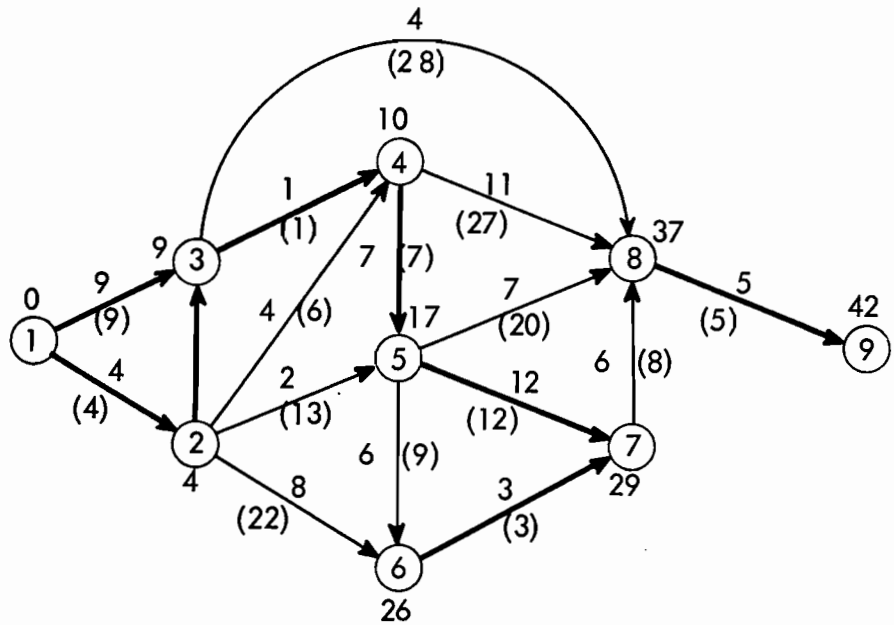


Numbers on lines indicate minimum durations of jobs. Numbers in parentheses indicate actual durations. Critical jobs are shown by heavy lines. These jobs must be done in their minimum duration, in order to minimize over-all cost.

<u>Times at nodes:</u>

$t_1 = 0$	$t_6 = 34$
$t_2 = 4$	$t_7 = 37$
$t_3 = 9$	$t_8 = 45$
$t_4 = 10$	$t_9 = 50$
$t_5 = 25$	

Fig. 9. Solution of Problem M-300.



Numbers on jobs indicate minimum durations of jobs. Numbers in parentheses indicate actual durations. Critical jobs are shown by heavy lines. These jobs must be done in their minimum duration, in order to minimize over-all cost.

<u>Times at nodes:</u>

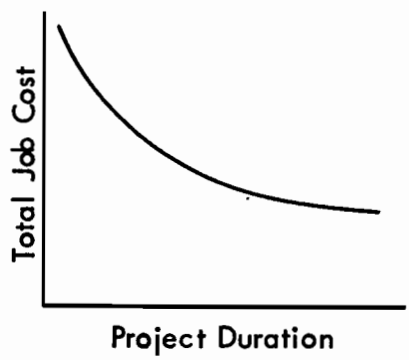
$t_1 = 0$	$t_6 = 26$
$t_2 = 4$	$t_7 = 29$
$t_3 = 9$	$t_8 = 37$
$t_4 = 10$	$t_9 = 42$
$t_5 = 17$	

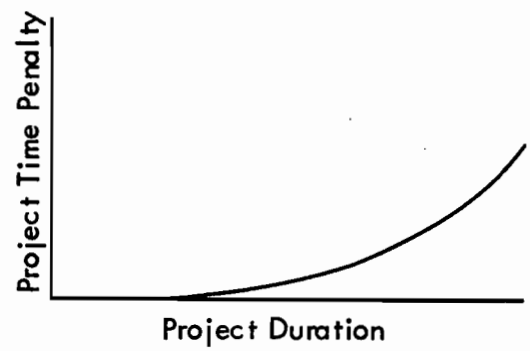
Fig. 10. Solution of Problem M-301.

Comparing this solution with that of M-300, it is seen that the durations of jobs 2,6; 3,8; 4,5; and 4,8 have each been reduced by 8 days. The cost slopes of these jobs are respectively -0.20, 0, -0.60, and -0.10. The additional cost incurred for these jobs therefore amounts to 0.90 per day. The project time penalty of 1.50 per day evidently makes the reduction in time worth while.

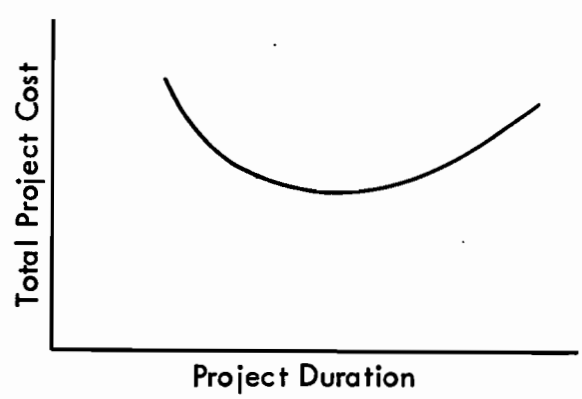
11.0 PROBLEM M-400

Problem M-400 illustrates the procedure used in developing a project cost curve, i.e., a plot of project cost versus project duration. For convenience, project cost may be considered as the sum of the individual job costs plus the project time penalty. The total job costs and the project time penalty may be plotted separately vs. project duration:





The project cost curve, then, is the sum of the total job cost and project time penalty curves:



Because of the shape of the two component curves, the project cost curve will generally have a minimum cost point, indicating the optimum project duration.

In problem M-400, the solution was carried only to the point of developing a total job cost curve, since this was the phase of essential interest. The addition of the cost penalty curve can be handled separately without linear programming; this presents no problems beyond assessing what the cost penalty should be in the actual project situation.

The total job cost curve was developed by the technique of multiple right-hand sides. Each right-hand side included some specified maximum project duration. No project time penalty was included, so the solution always called for the maximum project duration permitted. The cost function in each case represented the total job cost, since there was no time penalty. Thus a series of points was obtained, each project duration being accompanied by a total job cost which was the minimum possible for that duration. A plot of these points gave the total job cost curve (Fig. 11). It was necessary to add a constant term to make the job costs positive, since the cost function consisted of only the job-cost slopes, and therefore showed a negative total cost. The solution data used in plotting the curve were:

Right-Hand Side	Project Duration, days	Total Job Cost, dollars	Plus Constant Cost of $200.0
1	35	-65.0	$135.0
2	36	-68.2	131.8
3	38	-74.6	125.4
4	40	-79.0	121.0
5	45	-88.8	111.2
6	50	-98.3	101.7
7	55	-107.8	92.2
8	60	-117.3	82.7

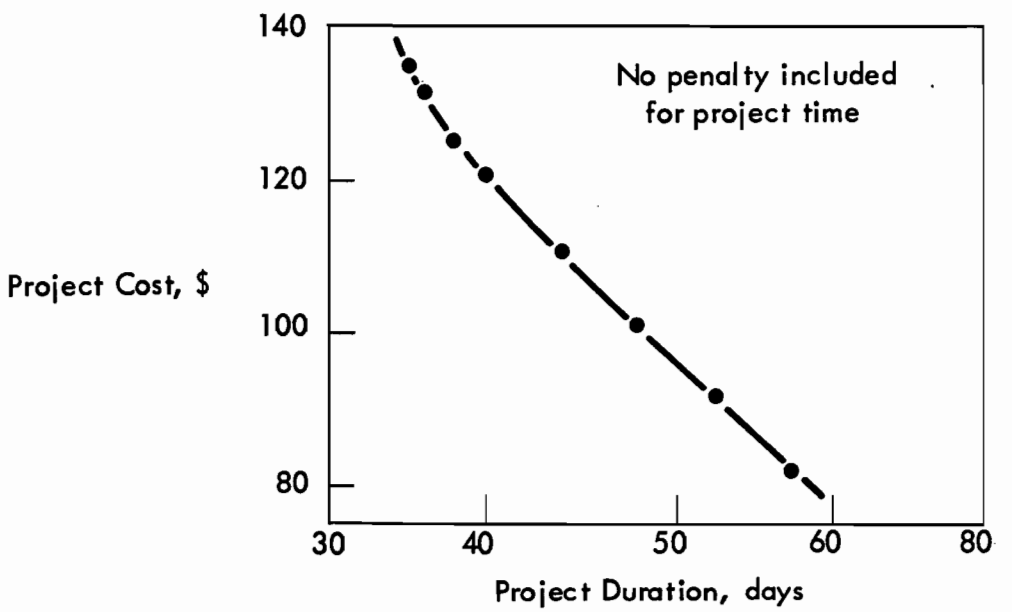


Fig. 11. Problem M-400 project cost curve.

12.0 REFERENCES

1. W. A. Gray and E. M. Kidd, "Critical Path Scheduling with Resource Leveling on the IBM-7090," AEC Research and Devel. Rpt. K-1499, March 20, 1962.

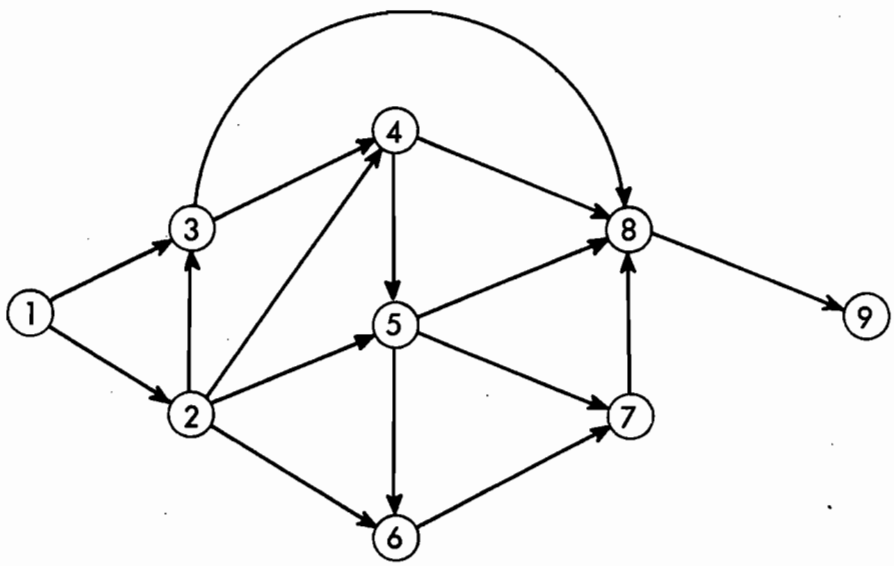
2. W. Orchard-Hays, "SCROL - A Comprehensive Operating System for Linear Programming on the IBM-704; Elementary Usage Manual," CEIR, Inc., Washington 5, D. C.

3. S. Dano, "Linear Programming in Industry," Springer Verlag, Vienna, 1960.

13.0 APPENDIX

13.1 Project Network Diagrams, Job Durations and Cost Slopes, and Equations

Problems M-100 and M-200



Job	Minimum Duration, days	Cost Slope, $	Job	Minimum Duration, days	Cost Slope, $
1,2	4	-0.40	4,5	7	-0.60
1,3	9	-0.20	4,8	11	-0.10
2,3	5	-0.30	5,6	6	-0.30
2,4	4	0	5,7	12	-0,30
2,5	2	-0.40	5,8	7	0
2,6	8	-0.20	6,7	3	-0.40
3,4	1	-0.10	7,8	6	-0.20
3,8	4	0	8,9	5	-0.10

Problems M-100 and M-200 Equations

$$-t_2 + u_1 = -4$$

$$-t_3 + u_2 = -9$$

$$+t_2 - t_3 + u_3 = -5$$

$$+t_2 - t_4 + u_4 = -4$$

$$+t_2 - t_5 + u_5 = -2$$

$$+t_2 - t_6 + u_6 = -8$$

$$+t_3 - t_4 + u_7 = -1$$

$$+t_3 - t_8 + u_8 = -4$$

$$+t_4 - t_5 + u_9 = -7$$

$$+t_4 - t_8 + u_{10} = -11$$

$$+t_5 - t_6 + u_{11} = -6$$

$$+t_5 - t_7 + u_{12} = -12$$

$$+t_5 - t_8 + u_{13} = -7$$

$$+t_6 - t_7 + u_{14} = -3$$

$$+t_7 - t_8 + u_{15} = -6$$

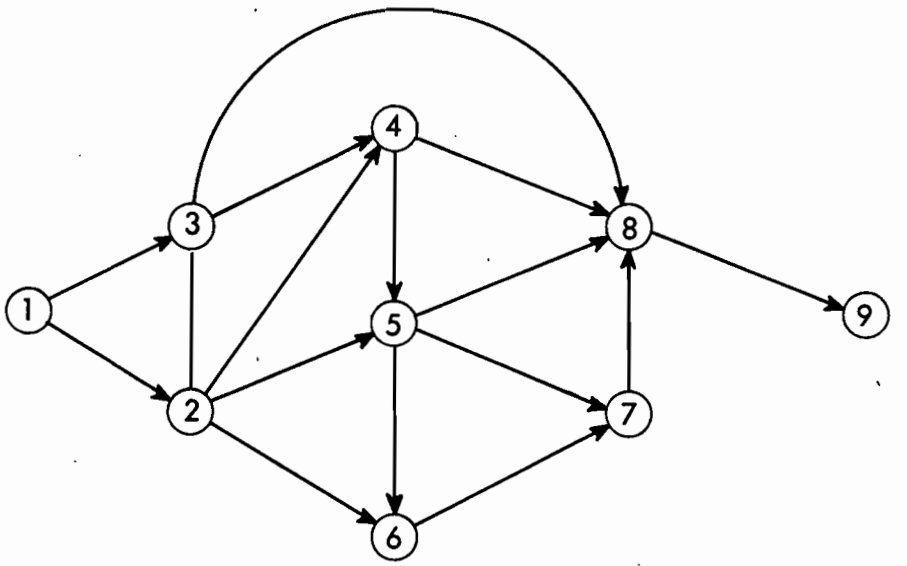
$$+t_8 - t_9 + u_{16} = -5$$

$$+t_9 + u_{17} = +50$$

Problems M-100 and M-200 Cost Equation

$$C = 0.50\ t_2 - 0.40\ t_3 + 0.60\ t_4 - 0.40\ t_5 - 0.10\ t_6 - 0.50\ t_7 - 0.20\ t_8 + 9.90\ t_9$$

Problem M-101



Job	Minimum Duration, days	Cost Slope, $	Job	Minimum Duration, days	Cost Slope, $
1,2	4	-0.40	4,5	7	-0.60
1,3	9	-0.20	4,8	11	-0.10
2,3	5	-0.30	5,6	6	-0.30
2,4	4	0	5,7	12	-0.30
2,5	2	-0.40	5,8	7	0
2,6	28	-0.20	6,7	3	-0.40
3,4	1	-0.10	7,8	6	-0.20
3,8	4	0	8,9	5	-0.10

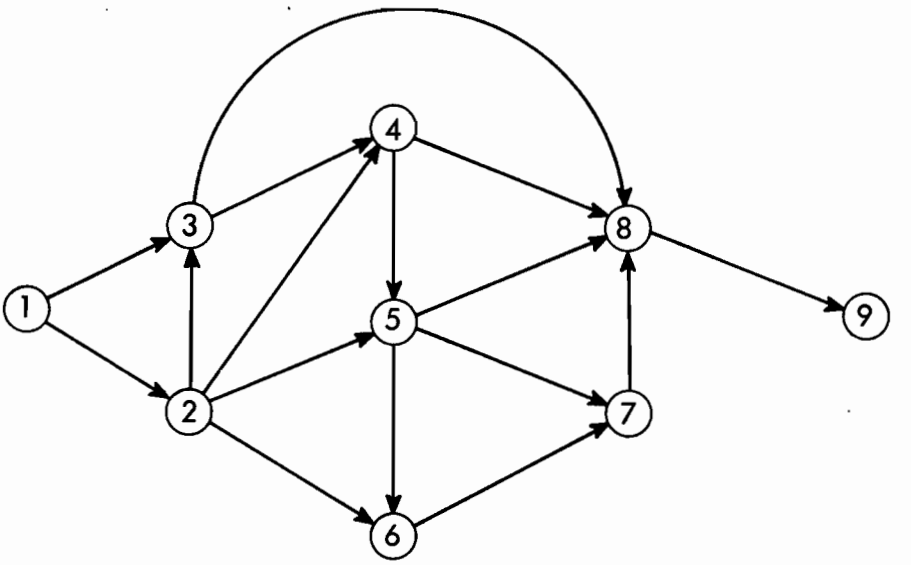
Problem M-101 Equations

$$-t_2 + u_1 = -4$$
$$-t_3 + u_2 = -9$$
$$+t_2 - t_3 + u_3 = -5$$
$$+t_2 - t_4 + u_4 = -4$$
$$+t_2 - t_5 + u_5 = -2$$
$$+t_2 - t_6 + u_6 = -28$$
$$+t_3 - t_4 + u_7 = -1$$
$$+t_3 - t_8 + u_8 = -4$$
$$+t_4 - t_5 + u_9 = -7$$
$$+t_4 - t_8 + u_{10} = -11$$
$$+t_5 - t_6 + u_{11} = -6$$
$$+t_5 - t_7 + u_{12} = -12$$
$$+t_5 - t_8 + u_{13} = -7$$
$$+t_6 - t_7 + u_{14} = -3$$
$$+t_7 - t_8 + u_{15} = -6$$
$$+t_8 - t_9 + u_{16} = -5$$
$$+t_9 + u_{17} = +50$$

Problem M-101 Cost Equation

$$C = 0.50\ t_2 - 0.40\ t_3 + 0.60\ t_4 - 0.40\ t_5 - 0.10\ t_6 - 0.50\ t_7 - 0.20\ t_8 + 9.90\ t_9$$

Problems M-102 and M-201



Job	Minimum Duration, days	Cost Slope, $	Job	Minimum Duration, days	Cost Slope, $
1,2	4	-0.40	4,5	7	-0.60
1,3	9	-0.20	4,8	11	-0.10
2,3	5	-0.30	5,6	6	-0.30
2,4	4	0	5,7	12	-0.30
2,5	2	-0.40	5,8	7	0
2,6	8	-0.20	6,7	3	-0.40
3,4	1	-0.10	7,8	6	-0.20
3,8	4	0	8,9	5	-0.10

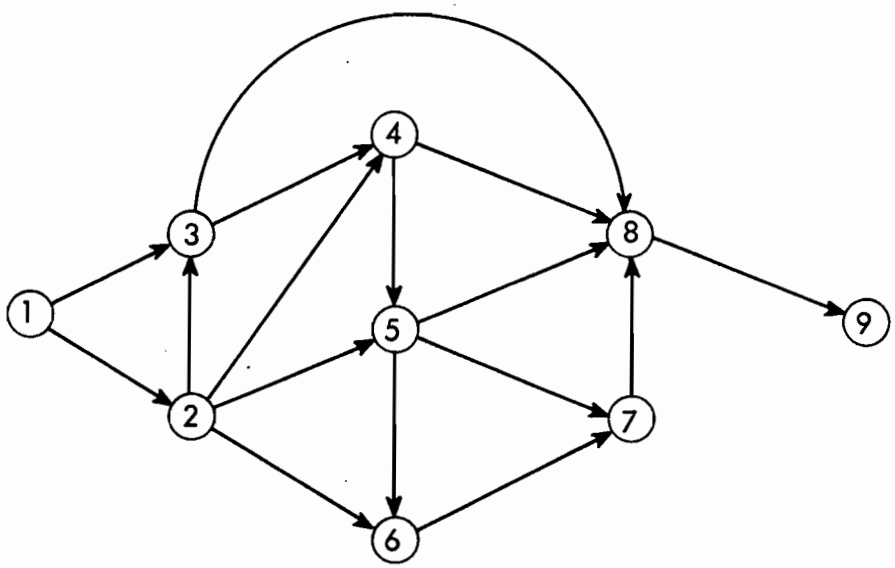
Problems M-102 and M-201 Equations

$$-t_2 + u_1 = -4$$
$$-t_3 + u_2 = -9$$
$$+t_2 - t_3 + u_3 = -5$$
$$+t_2 - t_4 + u_4 = -4$$
$$+t_2 - t_5 + u_5 = -2$$
$$+t_2 - t_6 + u_6 = -8$$
$$+t_3 - t_4 + u_7 = -1$$
$$+t_3 - t_8 + u_8 = -4$$
$$+t_4 - t_5 + u_9 = -7$$
$$+t_4 - t_8 + u_{10} = -11$$
$$+t_5 - t_6 + u_{11} = -6$$
$$+t_5 - t_7 + u_{12} = -12$$
$$+t_5 - t_8 + u_{13} = -7$$
$$+t_6 - t_7 + u_{14} = -3$$
$$+t_7 - t_8 + u_{15} = -6$$
$$+t_8 - t_9 + u_{16} = -5$$
$$+t_9 + u_{17} = +50$$

Problems M-102 and M-201 Cost Equation

$$C = 0.50\ t_2 - 0.40\ t_3 + 0.60\ t_4 - 0.40\ t_5 - 0.10\ t_6 - 0.50\ t_7 - 0.20\ t_8 + 0.40\ t_9$$

Problem M-103



Job	Minimum Duration, days	Cost Slope, $	Job	Minimum Duration, days	Cost Slope, $
1,2	4	-0.40	4,5	7	-0.60
1,3	9	-0.20	4,8	11	-0.10
2,3	5	-0.30	5,6	6	-0.30
2,4	4	0	5,7	12	-0.30
2,5	2	-0.40	5,8	7	0
2,6	28	-0.20	6,7	3	-0.40
3,4	1	-0.10	7,8	6	-0.20
3,8	4	0	8,9	5	-0.10

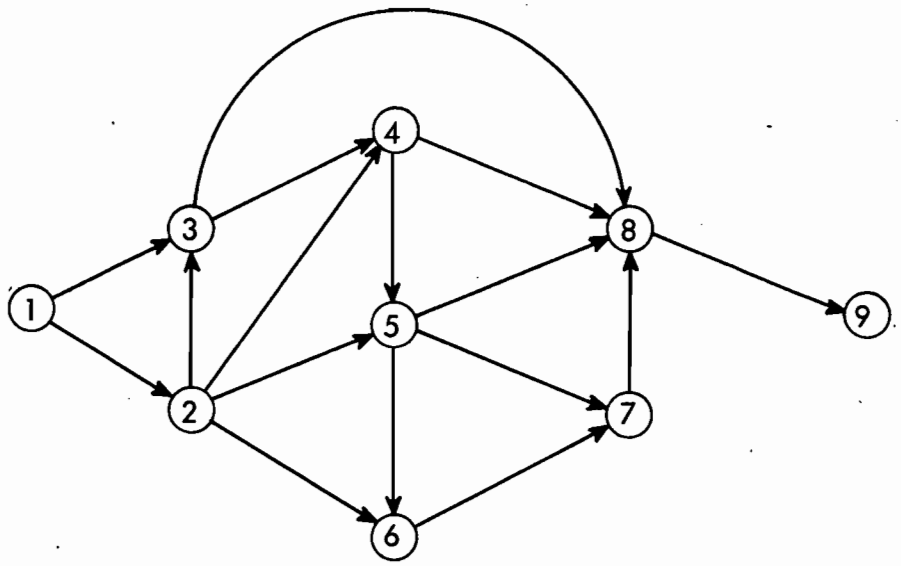
Problem M-103 Equations

$$-t_2 + u_1 = -4$$
$$-t_3 + u_2 = -9$$
$$+t_2 - t_3 + u_3 = -5$$
$$+t_2 - t_4 + u_4 = -4$$
$$+t_2 - t_5 + u_5 = -2$$
$$+t_2 - t_6 + u_6 = -28$$
$$+t_3 - t_4 + u_7 = -1$$
$$+t_3 - t_8 + u_8 = -4$$
$$+t_4 - t_5 + u_9 = -7$$
$$+t_4 - t_8 + u_{10} = -11$$
$$+t_5 - t_6 + u_{11} = -6$$
$$+t_5 - t_7 + u_{12} = -12$$
$$+t_5 - t_8 + u_{13} = -7$$
$$+t_6 - t_7 + u_{14} = -3$$
$$+t_7 - t_8 + u_{15} = -6$$
$$+t_8 - t_9 + u_{16} = -5$$
$$+t_9 + u_{17} = +50$$

Problem M-103 Cost Equation

$$C = 0.50\ t_2 - 0.40\ t_3 + 0.60\ t_4 - 0.40\ t_5 - 0.10\ t_6 - 0.50\ t_7 - 0.20\ t_8 + 0.40\ t_9$$

Problems M-300 and M-301



Job	Minimum Duration, days	Cost Slope, $	Job	Minimum Duration, days	Cost Slope, $
1,2	4	-0.40	4,5	7	-0.60
1,3	9	-0.20	4,8	11	-0.10
2,3	5	-0.30	5,6	6	-0.30
2,4	4	0	5,7	12	-0.30
2,5	2	-0.40	5,8	7	0
2,6	8	-0.20	6,7	3	-0.40
3,4	1	-0.10	7,8	6	see note
3,8	4	0	8,9	5	-0.10

Note: Job 7,8 has a three-piece cost curve; see Fig. 8, page 16.

Problems M-300 and M-301 Restrictions

$t_2 \geq 4$

$t_3 \geq 9$

$t_3 - t_2 \geq 5$

$t_4 - t_2 \geq 4$

$t_5 - t_2 \geq 2$

$t_6 - t_2 \geq 8$

$t_4 - t_3 \geq 1$

$t_8 - t_3 \geq 4$

$t_5 - t_4 \geq 7$

$t_8 - t_4 \geq 11$

$t_6 - t_5 \geq 6$

$t_7 - t_5 \geq 12$

$t_8 - t_5 \geq 7$

$t_7 - t_6 \geq 3$

$t_{10} - t_7 \geq 6$

$t_9 - t_8 \geq 5$

$-t_9 \geq -50$

$-t_{10} + t_7 \geq -8$

$t_{11} - t_{10} \geq 0$

$-t_{11} + t_{10} \geq -3$

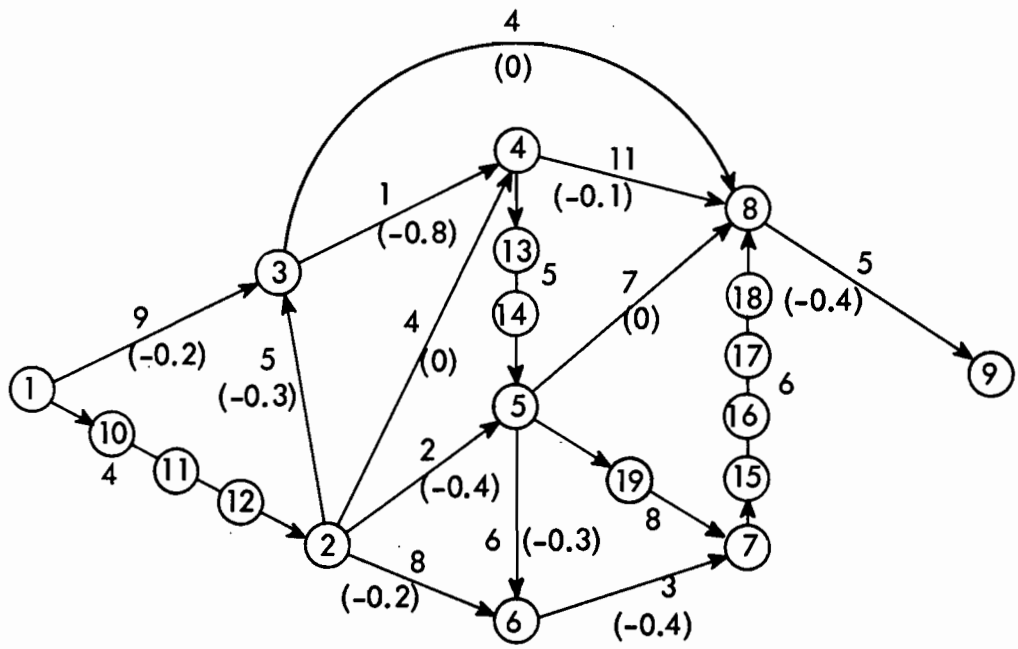
$t_8 - t_{11} \geq 0$

Problem M-300 Cost Equation

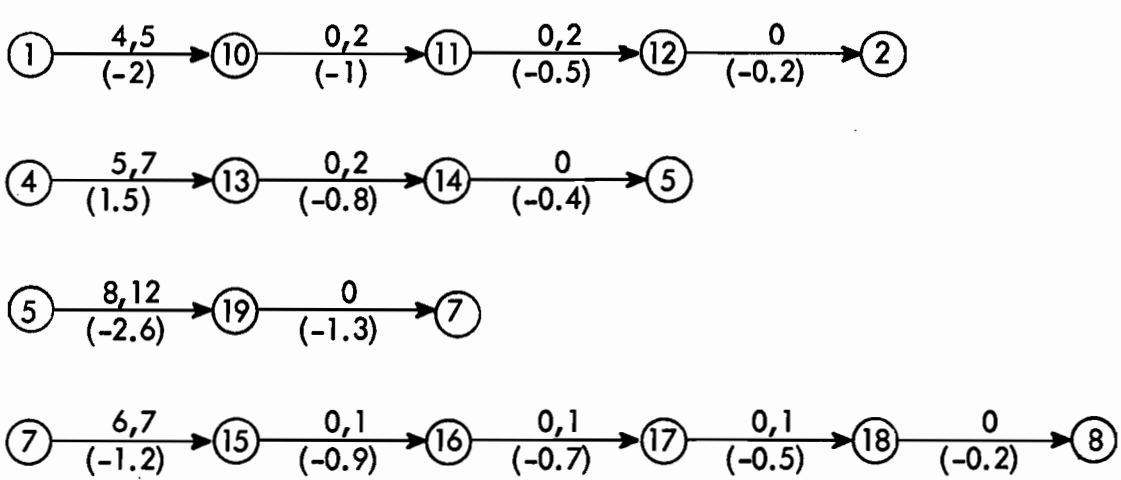
$$C = 0.50\, t_2 - 0.40\, t_3 + 0.60\, t_4 - 0.40\, t_5 - 0.10\, t_6 + 1.30\, t_7 - 0.20\, t_8 + 0.40\, t_9 - 1.00\, t_{10} - 0.80\, t_{11}$$

M-301 cost equation is identical except that coefficient of t_9 is +1.40.

Problem M-400



Number on line indicates minimum duration of job. Second number indicates maximum duration, if any. Numbers in parentheses indicate cost slopes.



Problem M-400 Restrictions

$-t_3 \leq -9$	$+t_{10} \leq 5$	$+t_{15} - t_7 \leq 7$
$-t_3 + t_2 \leq -5$	$-t_{11} + t_{10} \leq 0$	$-t_{16} + t_{15} \leq 0$
$-t_4 + t_3 \leq -1$	$+t_{11} - t_{10} \leq 2$	$+t_{16} - t_{15} \leq 1$
$-t_4 + t_2 \leq -4$	$-t_{12} + t_{11} \leq 0$	$-t_{17} + t_{16} \leq 0$
$-t_5 + t_2 \leq -2$	$+t_{12} - t_{11} \leq 2$	$+t_{17} - t_{16} \leq 1$
$-t_6 + t_2 \leq -8$	$-t_2 + t_{12} \leq 0$	$-t_{18} + t_{17} \leq 0$
$-t_8 + t_3 \leq -4$	$-t_{13} + t_4 \leq -5$	$+t_{18} - t_{17} \leq 1$
$-t_8 + t_4 \leq -11$	$+t_{13} - t_4 \leq 7$	$-t_8 + t_{18} \leq 0$
$-t_8 + t_5 \leq -7$	$-t_{14} + t_{13} \leq 0$	$-t_{19} + t_5 \leq -8$
$-t_6 + t_5 \leq -6$	$+t_{14} - t_{13} \leq 2$	$+t_{19} - t_5 \leq 12$
$-t_7 + t_6 \leq -3$	$-t_5 + t_{14} \leq 0$	$-t_7 + t_{19} \leq 0$
$-t_9 + t_8 \leq -5$	$-t_{15} + t_7 \leq -6$	$+t_9 \leq 35$ (see note)
$-t_{10} \leq -4$		

Note: The first right-hand side limited the over-all duration to 35 as indicated; successive right-hand sides changed this to 36, 38, 40, 45, 50, 55, and 60 days.

Problem M-400 Cost Equation

$$
\begin{aligned}
C = {} & +0.7\, t_2 + 0.3\, t_3 + 0.8\, t_4 + 2.1\, t_5 - 0.1\, t_6 \\
& - 0.5\, t_7 + 0.1\, t_8 - 0.4\, t_9 - 1.0\, t_{10} - 0.5\, t_{11} \\
& - 0.3\, t_{12} - 0.7\, t_{13} - 0.4\, t_{14} - 0.3\, t_{15} - 0.2\, t_{16} \\
& - 0.2\, t_{17} - 0.3\, t_{18} - 1.3\, t_{19}
\end{aligned}
$$

13.2 Input Cards

Problem M-100

```
*ID   CRITICAL PATH SCHEDULING PROBLEM M-100  R.SALMON JUNE 6 1962
*CHARGE(3215)
*TAPE(1,5574,SAVE)(2,POOL)(3,POOL)(4,POOL)(5,POOL,SAVE)(6,OUTPUT)
*EXECUTE
*TAPE(10,POOL,SAVE)
*ASSIGN(1,A5)(2,A0)(3,A4)(4,B3)(5,B5)(10,B6)
*TYPE(COMP,LT,4,16K)
RDDATA
9       9
*  LP M-100 JUNE 6,1962
17   0    1
1   UP001 2   UP002 3   UP003 4   UP004 5   UP005 6   UP006 7   UP007
8   UP008 9   UP009 10  UP010 11  UP011 12  UP012 13  UP013 14  UP014
15  UP015 16  UP016 17  UP017
MATRIX
 AA0020   .5
 AA0021  -1
 AA0023   1
 AA0024   1
 AA0025   1
 AA0026   1
 AA0030  -.4
 AA0032  -1
 AA0033  -1
 AA0037   1
 AA0038   1
 AA0040   .6
 AA0044  -1
 AA0047  -1
 AA0049   1
 AA00410  1
 AA0050  -.4
 AA0055  -1
 AA0059  -1
 AA00511  1
 AA00512  1
 AA00513  1
 AA0060  -.1
 AA0066  -1
 AA00611 -1
 AA00614  1
 AA0070  -.5
 AA00712 -1
 AA00714 -1
 AA00715  1
 AA0080  -.2
 AA0088  -1
 AA00810 -1
```

(Problem M-100, continued)

```
 AA00813 -1
 AA00815 -1
 AA00816  1
 AA0090   9.9
 AAC0916 -1
 AAC0917  1
 UP0011   1
 UPC022   1
 UP0033   1
 UP0044   1
 UP0055   1
 UP0066   1
 UP0077   1
 UP0088   1
 UPC099   1
 UP01010  1
 UP01111  1
 UP01212  1
 UP01313  1
 UP01414  1
 UP01515  1
 UP01616  1
 UP01717  1
FIRST B
       1  -4
       2  -9
       3  -5
       4  -4
       5  -2
       6  -8
       7  -1
       8  -4
       9  -7
       10 -11
       11 -6
       12 -12
       13 -7
       14 -3
       15 -6
       16 -5
       17  50
NEXT B,2
       1  -4
       2  -9
       3  -5
       4  -4
       5  -2
       6  -28
```

(Problem M-100, continued)

```
        7  -1
        8  -4
        9  -7
        10 -11
        11 -6
        12 -12
        13 -7
        14 -3
        15 -6
        16 -5
        17  50
EOF
SETUP
NORMAL
NORMAL,,2
GETOFF
```

Problem M-102

```
*ID   CRITICAL PATH SCHEDULING PROBLEM M-102  R.SALMON JUNE 12 1962
*CHARGE(3215)
*TAPE(1,5574,SAVE)(2,POOL)(3,POOL)(4,POOL)(5,POOL,SAVE)(6,OUTPUT)
*EXECUTE
*TAPE(10,POOL,SAVE)
*ASSIGN(1,A5)(2,A0)(3,A4)(4,B3)(5,B5)(10,B6)
*TYPE(CCMP,LT,4,16K)
RDDATA
9       9
*  LP M-102 JUNE 12 1962
17   0    1
1   UP001 2   UP002 3   UP003 4   UP004 5   UP005 6   UP006 7   UP007
8   UP008 9   UP009 10  UP010 11  UP011 12  UP012 13  UP013 14  UP014
15  UP015 16  UP016 17  UP017
MATRIX
```

(Problem M-102, continued)

AA0020	.5
AA0021	-1
AA0023	1
AA0024	1
AA0025	1
AA0026	1
AA0030	-.4
AA0032	-1
AA0033	-1
AA0037	1
AA0038	1
AA0040	.6
AA0044	-1
AA0047	-1
AA0049	1
AA00410	1
AA0050	-.4
AA0055	-1
AA0059	-1
AA00511	1
AA00512	1
AA00513	1
AA0060	-.1
AA0066	-1
AA00611	-1

(Problem M-102, continued)

AA00614	1
AA0070	-.5
AA00712	-1
AA00714	-1
AA00715	1
AA0080	-.2
AA0088	-1
AA00810	-1
AA00813	-1
AA00815	-1
AA00816	1
AA0090	.4
AA00916	-1
AA00917	1
UP0011	1
UP0022	1
UP0033	1
UP0044	1
UP0055	1
UP0066	1
UP0077	1
UP0088	1
UP0099	1
UP01010	1
UP01111	1

(Problem M-102, continued)

```
 UP01212  1
 UP01313  1
 UP01414  1
 UP01515  1
 UP01616  1
 UP01717  1
FIRST B
      1  -4
      2  -9
      3  -5
      4  -4
      5  -2
      6  -8
      7  -1
      8  -4
      9  -7
      10 -11
      11 -6
      12 -12
      13 -7
      14 -3
      15 -6
      16 -5
      17  50
NEXT B,2
```

(Problem M-102, continued)

```
        1  -4
        2  -9
        3  -5
        4  -4
        5  -2
        6  -28
        7  -1
        8  -4
        9  -7
        10 -11
        11 -6
        12 -12
        13 -7
        14 -3
        15 -6
        16 -5
        17  50
EOF
SETUP
NORMAL
NORMAL,,2
GETOFF
```

Problem M-200

```
*ID   CRITICAL PATH PROBLEM M-200  R.SALMON JUNE 12 1962
*CHARGE(3215)
*PAPER(1,1)
*TAPE(1,5574,SAVE)(2,POOL)(3,POOL)(4,POOL)(5,POOL,SAVE)(6,OUTPUT)
*EXECUTE
*TAPE(10,POOL,SAVE)
*ASSIGN(1,A5)(2,A0)(3,A4)(4,B3)(5,B5)(10,B6)
*TYPE(COMP,LT,4,16K)
RDDATA
9       9
*  LP M-200 JUNE 12 1962
8     0     1
1    UP001 2    UP002 3    UP003 4    UP004 5    UP005 6    UP006 7    UP007
8    UP008
MATRIX
 BB0010  -4
 BB0011   1
 BB0020  -9
 BB0022   1
 BB0030  -5
 BB0031  -1
 BB0032   1
 BB0040  -4
 BB0041  -1
 BB0043   1
 BB0050  -2
 BB0051  -1
 BB0054   1
 BB0060  -8
 BB0061  -1
 BB0065   1
 BB0070  -1
 BB0072  -1
 BB0073   1
 BB0080  -4
 BB0082  -1
 BB0087   1
 BB0090  -7
 BB0093  -1
 BB0094   1
 BB0100  -11
 BB0103  -1
 BB0107   1
 BB0110  -6
 BB0114  -1
 BB0115   1
```

(Problem M-200, continued)

```
 BB0120  -12
 BB0124  -1
 BB0126   1
 BB0130  -7
 BB0134  -1
 BB0137   1
 BB0140  -3
 BB0145  -1
 BB0146   1
 BB0150  -6
 BB0156  -1
 BB0157   1
 BB0160  -5
 BB0167  -1
 BB0168   1
 BB0170   50
 BB0178  -1
 UP0011   1
 UP0022   1
 UP0033   1
 UP0044   1
 UP0055   1
 UP0066   1
 UP0077   1
 UP0088   1
FIRST B
      1   .5
      2  -.4
      3   .6
      4  -.4
      5  -.1
      6  -.5
      7  -.2
      8   9.9
NEXT B,2
      1   .5
      2  -.4
      3   .6
      4  -.4
      5  -.1
      6  -.5
      7  -.2
      8   .4
EOF
SETUP
NORMAL
NORMAL,,2
GETOFF
```

Problem M-300

```
*ID   CRITICAL PATH SCHEDULING PROBLEM M-300  R.SALMON JUNE 12 1962
*CHARGE(3215)
*TAPE(1,5574,SAVE)(2,POOL)(3,POOL)(4,POOL)(5,POOL,SAVE)(6,OUTPUT)
*EXECUTE
*TAPE(10,POOL,SAVE)
*ASSIGN(1,A5)(2,A0)(3,A4)(4,B3)(5,B5)(10,B6)
*TYPE(COMP,LT,4,16K)
RDDATA
9       9
*  LP M-300 JUNE 12 1962
10    0     1
1    UP001
2    UP002
3    UP003
4    UP004
5    UP005
6    UP006
7    UP007
8    UP008
9    UP009
10   UP010
MATRIX
 BB0010  -4
 BB0011   1
 BB0020  -9
 BB0022   1
 BB0030  -5
 BB0031  -1
 BB0032   1
 BB0040  -4
 BB0041  -1
 BB0043   1
 BB0050  -2
 BB0051  -1
 BB0054   1
 BB0060  -8
 BB0061  -1
 BB0065   1
 BB0070  -1
 BB0072  -1
 BB0073   1
 BB0080  -4
 BB0082  -1
 BB0087   1
 BB0090  -7
 BB0093  -1
```

(Problem M-300, continued)

```
 BB0094    1
 BB0100   -11
 BB0103   -1
 BB0107    1
 BB0110   -6
 BB0114   -1
 BB0115    1
 BB0120   -12
 BB0124   -1
 BB0126    1
 BB0130   -7
 BB0134   -1
 BB0137    1
 BB0140   -3
 BB0145   -1
 BB0146    1
 BB0150   -6
 BB0156   -1
 BB0159    1
 BB0160   -5
 BB0167   -1
 BB0168    1
 BB0170    50
 BB0178   -1
 BB0180    8
 BB0186    1
 BB0189   -1
 BB0199   -1
 BB01910   1
 BB0200    3
 BB0209    1
 BB02010  -1
 BB0217    1
 BB02110  -1
 UP0011    1
 UP0022    1
 UP0033    1
 UP0044    1
 UP0055    1
 UP0066    1
 UP0077    1
 UP0088    1
 UP0099    1
 UP01010   1
FIRST B
       1    .5
       2   -.4
       3    .6
```

(Problem M-300, continued)

```
        4  -.4
        5  -.1
        6   1.3
        7  -.2
        8   .4
        9  -1
        10 -.8
NEXT 8,2
        1   .5
        2  -.4
        3   .6
        4  -.4
        5  -.1
        6   1.3
        7  -.2
        8   1.4
        9  -1
        10 -.8
EOF
SETUP
NORMAL
NORMAL,,2
GETOFF
```

Problem M-400

```
*ID   PROBLEM M-400 R.SALMON JUNE 25,1962
*CHARGE(3215)
*TAPE(1,5574,SAVE)(2,POOL)(3,POOL)(4,POOL)(5,POOL,SAVE)(6,OUTPUT)
*TAPE(10,POOL,SAVE)
*ASSIGN(1,A5)(2,A0)(3,A4)(4,B3)(5,B5)(10,B6)
*EXECUTE
*TYPE(COMP,LT,4,16K)
RDDATA
9       9
37    0     1
1    UP001
2    UP002
3    UP003
4    UP004
5    UP005
6    UP006
7    UP007
8    UP008
9    UP009
10   UP010
11   UP011
12   UP012
13   UP013
14   UP014
15   UP015
16   UP016
17   UP017
18   UP018
19   UP019
20   UP020
21   UP021
22   UP022
23   UP023
24   UP024
25   UP025
26   UP026
27   UP027
28   UP028
29   UP029
30   UP030
31   UP031
32   UP032
33   UP033
34   UP034
35   UP035
36   UP036
```

(Problem M-400, continued)

```
37  UP037
MATRIX
 AA0020    .7
 AA0022    1
 AA0024    1
 AA0025    1
 AA0026    1
 AA00219  -1
 AA0030    .3
 AA0031   -1
 AA0032   -1
 AA0033    1
 AA0037    1
 AA0040    .8
 AA0043   -1
 AA0044   -1
 AA0048    1
 AA00420   1
 AA00421  -1
 AA0050    2.1
 AA0055   -1
 AA0059    1
 AA00510   1
 AA00524  -1
 AA00534   1
 AA00535  -1
 AA0060   -.1
 AA0066   -1
 AA00610  -1
 AA00611   1
 AA0070   -.5
 AA00711  -1
 AA00725   1
 AA00726  -1
 AA00736  -1
 AA0080    .1
 AA0087   -1
 AA0088   -1
 AA0089   -1
 AA00812   1
 AA00833  -1
 AA0090   -.4
 AA00912  -1
 AA00937   1
 AA0100   -1
 AA01013  -1
 AA01014   1
 AA01015   1
```

(Problem M-400, continued)

```
AA01016 -1
AA0110  -.5
AA01115 -1
AA01116  1
AA01117  1
AA01118 -1
AA0120  -.3
AA01217 -1
AA01218  1
AA01219  1
AA0130  -.7
AA01320 -1
AA01321  1
AA01322  1
AA01323 -1
AA0140  -.4
AA01422 -1
AA01423  1
AA01424  1
AA0150  -.3
AA01525 -1
AA01526  1
AA01527  1
AA01528 -1
AA0160  -.2
AA01627 -1
AA01628  1
AA01629  1
AA01630 -1
AA0170  -.2
AA01729 -1
AA01730  1
AA01731  1
AA01732 -1
AA0180  -.3
AA01831 -1
AA01832  1
AA01833  1
AA0190  -1.3
AA01934 -1
AA01935  1
AA01936  1
UP0011   1
UP0022   1
UP0033   1
UP0044   1
UP0055   1
UP0066   1
```

(Problem M-400, continued)

```
 UP0077    1
 UP0088    1
 UP0099    1
 UP01010   1
 UP01111   1
 UP01212   1
 UP01313   1
 UP01414   1
 UP01515   1
 UP01616   1
 UP01717   1
 UP01818   1
 UP01919   1
 UP02020   1
 UP02121   1
 UP02222   1
 UP02323   1
 UP02424   1
 UP02525   1
 UP02626   1
 UP02727   1
 UP02828   1
 UP02929   1
 UP03030   1
 UP03131   1
 UP03232   1
 UP03333   1
 UP03434   1
 UP03535   1
 UP03636   1
 UP03737   1
FIRST B
       1  -9
       2  -5
       3  -1
       4  -4
       5  -2
       6  -8
       7  -4
       8  -11
       9  -7
       10 -6
       11 -3
       12 -5
       13 -4
       14  5
       16  2
       18  2
```

(Problem M-400, continued)

```
        20 -5
        21  7
        23  2
        25 -6
        26  7
        28  1
        30  1
        32  1
        34 -8
        35  12
        37  35
NEXT B,2
        1  -9
        2  -5
        3  -1
        4  -4
        5  -2
        6  -8
        7  -4
        8  -11
        9  -7
        10 -6
        11 -3
        12 -5
        13 -4
        14  5
        16  2
        18  2
        20 -5
        21  7
        23  2
        25 -6
        26  7
        28  1
        30  1
        32  1
        34 -8
        35  12
        37  36
NEXT B,3
        1  -9
        2  -5
        3  -1
        4  -4
        5  -2
        6  -8
        7  -4
        8  -11
```

(Problem M-400, continued)

```
        9  -7
        10 -6
        11 -3
        12 -5
        13 -4
        14  5
        16  2
        18  2
        20 -5
        21  7
        23  2
        25 -6
        26  7
        28  1
        30  1
        32  1
        34 -8
        35  12
        37  38
NEXT B,4
        1  -9
        2  -5
        3  -1
        4  -4
        5  -2
        6  -8
        7  -4
        8  -11
        9  -7
        10 -6
        11 -3
        12 -5
        13 -4
        14  5
        16  2
        18  2
        20 -5
        21  7
        23  2
        25 -6
        26  7
        28  1
        30  1
        32  1
        34 -8
        35  12
        37  40
```

(Problem M-400, continued)

```
NEXT B,5
       1  -9
       2  -5
       3  -1
       4  -4
       5  -2
       6  -8
       7  -4
       8  -11
       9  -7
       10 -6
       11 -3
       12 -5
       13 -4
       14  5
       16  2
       18  2
       20 -5
       21  7
       23  2
       25 -6
       26  7
       28  1
       30  1
       32  1
       34 -8
       35  12
       37  45
NEXT B,6
       1  -9
       2  -5
       3  -1
       4  -4
       5  -2
       6  -8
       7  -4
       8  -11
       9  -7
       10 -6
       11 -3
       12 -5
       13 -4
       14  5
       16  2
       18  2
       20 -5
       21  7
       23  2
```

(Problem M-400, continued)

```
        25 -6
        26  7
        28  1
        30  1
        32  1
        34 -8
        35  12
        37  50
NEXT B,7
        1  -9
        2  -5
        3  -1
        4  -4
        5  -2
        6  -8
        7  -4
        8  -11
        9  -7
        10 -6
        11 -3
        12 -5
        13 -4
        14  5
        16  2
        18  2
        20 -5
        21  7
        23  2
        25 -6
        26  7
        28  1
        30  1
        32  1
        34 -8
        35  12
        37  55
NEXT B,8
        1  -9
        2  -5
        3  -1
        4  -4
        5  -2
        6  -8
        7  -4
        8  -11
        9  -7
        10 -6
        11 -3
```

(Problem M-400, continued)

```
        12 -5
        13 -4
        14  5
        16  2
        18  2
        20 -5
        21  7
        23  2
        25 -6
        26  7
        28  1
        30  1
        32  1
        34 -8
        35  12
        37  60
EOF
SETUP
NORMAL
NORMAL,,2
NORMAL,,3
NORMAL,,4
NORMAL,,5
NORMAL,,6
NORMAL,,7
NORMAL,,8
GETOFF
```

13.3 Solution Output Tapes

Problem M-100 Solution

TOTAL ITERS	NO. ETAS	TP.2 RECS	PHS TYP	ROW NO.	CURRENT VALUE	SOLN FEAS	RHS NO.
015	015	000	II	000	-369.49999	YES	001

PRIMAL-DUAL SOLUTIONS

J	BETA	I	B	PI
	369.50000038-			1.00000000
UP003	.	1	4.00000000-	.50000000
AA004	10.00000000	2	9.00000000-	8.90000002
AA002	4.00000000	3	5.00000000-	.
UP008	22.00000000	4	4.00000000-	.
UP005	11.00000000	5	2.00000000-	.
UP010	14.00000000	6	8.00000000-	.
UP004	2.00000000	7	1.00000000-	9.30000001
AA008	35.00000000	8	4.00000000-	.
UP006	14.00000000	9	7.00000000-	8.70000001
AA006	26.00000000	10	11.00000000-	.
UP013	11.00000000	11	6.00000000-	.
UP011	3.00000000	12	12.00000000-	9.10000000
AA007	29.00000000	13	7.00000000-	.
AA005	17.00000000	14	3.00000000-	.09999999
AA003	9.00000000	15	6.00000000-	9.70000000
AA009	40.00000000	16	5.00000000-	9.89999999
UP017	10.00000000	17	50.00000000	.

Problem M-101 Solution

TOTAL ITERS	NO. ETAS	TP.2 RECS	PHS TYP	ROW NO.	CURRENT VALUE	SOLN FEAS	RHS NO.
017	017	000	II	000	-421.69999	YES	002

PRIMAL-DUAL SOLUTIONS

J	BETA	I	B	PI
	421.70000049-		.	1.00000000
UP003	.	1	4.00000000-	9.20000001
AA004	10.00000000	2	9.00000000-	.20000000
AA002	4.00000000	3	5.00000000-	.
UP008	28.00000000	4	4.00000000-	.
UP005	17.00000000	5	2.00000000-	.
UP010	20.00000000	6	28.00000000-	8.70000001
UP004	2.00000000	7	1.00000000-	.59999999
AA008	41.00000000	8	4.00000000-	.
UP009	6.00000000	9	7.00000000-	.
AA006	32.00000000	10	11.00000000-	.
UP013	11.00000000	11	6.00000000-	.
UP011	3.00000000	12	12.00000000-	.39999999
AA007	35.00000000	13	7.00000000-	.
AA005	23.00000000	14	3.00000000-	8.80000001
AA003	9.00000000	15	6.00000000-	9.70000000
AA009	46.00000000	16	5.00000000-	9.89999999
UP017	4.00000000	17	50.00000000	.

Problem M-102 Solution

TOTAL ITERS	NO. ETAS	TP.2 RECS.	PHS TYP	ROW NO.	CURRENT VALUE	SOLN FEAS	RHS NO.
016	016	000	II	000	18.499999	YES	001

PRIMAL-DUAL SOLUTIONS

J	BETA	I	B	PI
	18.49999943		.	1.00000000
UP003	.	1	4.00000000-	.50000000
AA004	10.00000000	2	9.00000000-	.20000000
AA002	4.00000000	3	5.00000000-	.
UP008	32.00000000	4	4.00000000-	.
UP005	21.00000000	5	2.00000000-	.
UP010	24.00000000	6	8.00000000-	.
UP004	2.00000000	7	1.00000000-	.59999999
AA008	45.00000000	8	4.00000000-	.
UP006	24.00000000	9	7.00000000-	.
AA006	36.00000000	10	11.00000000-	.
UP013	11.00000000	11	6.00000000-	.
UP011	3.00000000	12	12.00000000-	.39999999
AA007	39.00000000	13	7.00000000-	.
AA005	27.00000000	14	3.00000000-	.09999999
AA003	9.00000000	15	6.00000000-	.99999998
AA009	50.00000000	16	5.00000000-	1.19999997
UP009	10.00000000	17	50.00000000	.79999998

Problem M-103 Solution

TOTAL ITERS	NO. ETAS	TP.2 RECS	PHS TYP	ROW NO.	CURRENT VALUE	SOLN FEAS	RHS NO.
016	016	000	II	000	18.499999	YES	002

PRIMAL-DUAL SOLUTIONS

J	BETA	I	B	PI
	18.49999943		.	1.00000000
UP003	.	1	4.00000000-	.50000000
AA004	10.00000000	2	9.00000000-	.20000000
AA002	4.00000000	3	5.00000000-	.
UP008	32.00000000	4	4.00000000-	.
UP005	21.00000000	5	2.00000000-	.
UP010	24.00000000	6	8.00000000-	.
UP004	2.00000000	7	1.00000000-	.59999999
AA008	45.00000000	8	4.00000000-	.
UP006	24.00000000	9	7.00000000-	.
AA006	36.00000000	10	11.00000000-	.
UP013	11.00000000	11	6.00000000-	.
UP011	3.00000000	12	12.00000000-	.39999999
AA007	39.00000000	13	7.00000000-	.
AA005	27.00000000	14	3.00000000-	.09999999
AA003	9.00000000	15	6.00000000-	.99999998
AA009	50.00000000	16	5.00000000-	1.19999997
UP009	10.00000000	17	50.00000000	.79999998

Problem M-200 Solution (same as M-100 but done in dual form)

TOTAL ITERS	NO. ETAS	TP.2 RECS	PHS TYP	ROW NO.	CURRENT VALUE	SOLN FEAS	RHS NO.
011	011	000	II	000	369.49999	YES	001

PRIMAL-DUAL SOLUTIONS

J	BETA	I	B	PI
	369.49999999		.	1.00000000
BB001	.50000000	1	.50000000	4.00000000
BB007	9.29999999	2	.39999999-	9.00000000
BB002	8.89999999	3	.60000000	10.00000000
BB012	9.09999999	4	.39999999-	17.00000000
BB014	.10000000	5	.10000000-	26.00000000
BB015	9.69999999	6	.50000000-	29.00000000
BB016	9.89999999	7	.20000000-	35.00000000
BB009	8.69999999	8	9.89999999	40.00000000

Check: The solution is identical with that of M-100. This is all the check required.

Problem M-201 Solution (cost change, dual form)

TOTAL ITERS	NO. ETAS	TP.2 RECS	PHS TYP	ROW NO.	CURRENT VALUE	SOLN FEAS	RHS NO.
016	016	000	II	000	-18.499999	YES	002

PRIMAL-DUAL SOLUTIONS

J	BETA	I	B	PI
	18.50000000-		.	1.00000000
BB001	.50000000	1	.50000000	4.00000000
BB017	.80000000	2	.39999999-	9.00000000
BB012	.39999999	3	.60000000	10.00000000
BB007	.60000000	4	.39999999-	27.00000000
BB014	.10000000	5	.10000000-	36.00000000
BB015	1.00000000	6	.50000000-	39.00000000
BB016	1.20000000	7	.20000000-	45.00000000
BB002	.20000000	8	.39999999	50.00000000

Time stretched out to 50 days. Second RHS of M-200. Cost penalty lowered from $10 to $0.50, causing a stretchout to a maximum of 50 days. This corresponds to M-102 but in the dual form.

Check: The solution is identical with M-102.

Problem M-300 Solution

TOTAL ITERS	NO. ETAS	TP.2 RECS	PHS TYP	ROW NO.	CURRENT VALUE	SOLN FEAS	RHS NO.
014	014	000	II	000	-30.899999	YES	001

PRIMAL-DUAL SOLUTIONS

J	BETA	I	B	PI
	30.90000000-		.	1.00000000
BB001	.50000000	1	.50000000	4.00000000
BB007	.60000000	2	.39999999-	9.00000000
BB002	.20000000	3	.60000000	10.00000000
BB017	.80000000	4	.39999999-	25.00000000
BB014	.10000000	5	.10000000-	34.00000000
BB018	.79999999	6	1.29999999	37.00000000
BB016	1.20000000	7	.20000000-	45.00000000
BB019	.20000000	8	.39999999	50.00000000
BB012	.39999999	9	1.00000000-	45.00000000
BB021	1.00000000	10	.79999999-	45.00000000

Job cost curve added on Job 7,8.

Problem M-301 Solution (project time penalty 1.50; otherwise same as M-300)

TOTAL ITERS	NO. ETAS	TP.2 RECS	PHS TYP	ROW NO.	CURRENT VALUE	SOLN FEAS	RHS NO.
016	016	000	II	000	17.499999	YES	002

PRIMAL-DUAL SOLUTIONS

J	BETA	I	B	PI
	17.49999999		.	1.00000000
BB001	.50000000	1	.50000000	4.00000000
BB007	.79999999	2	.39999999-	9.00000000
BB002	.39999999	3	.60000000	10.00000000
BB009	.19999999	4	.39999999-	17.00000000
BB014	.10000000	5	.10000000-	26.00000000
BB018	.60000000	6	1.29999999	29.00000000
BB016	1.39999999	7	.20000000-	37.00000000
BB019	.39999999	8	1.39999999	42.00000000
BB012	.59999999	9	1.00000000-	37.00000000
BB021	1.19999999	10	.79999999-	37.00000000

Job 5,7 has become critical, forcing node 5 back to 17 (29 - 12 = 17). Total project time reduced to 42 (2 days greater than absolute minimum; this reflects the cost curve on job 7,8 which has a slope of -2.0 for two days.

Problem M-400 Solution

TOTAL ITERS	NO. ETAS	TP.2 RECS	PHS TYP	ROW NO.	CURRENT VALUE	SOLN FEAS	RHS NO.
031	031	000	II	000	64.999997	YES	001

PRIMAL-DUAL SOLUTIONS

J	BETA	I	B	PI
	64.99999864		.	1.00000000
UP001	.	1	9.00000000-	.
UP007	17.00000000	2	5.00000000-	2.10000000
UP010	.	3	1.00000000-	1.80000001
UP009	8.00000000	4	4.00000000-	.
UP004	2.00000000	5	2.00000000-	.
AA005	15.00000000	6	8.00000000-	.
AA008	30.00000000	7	4.00000000-	.
AA002	4.00000000	8	11.00000000-	.
AA003	9.00000000	9	7.00000000-	.
UP034	1.00000000	10	6.00000000-	.
AA007	24.00000000	11	3.00000000-	.09999999
AA009	35.00000000	12	5.00000000-	2.79999996
AA004	10.00000000	13	4.00000000-	1.00000000
UP014	1.00000000	14	5.00000000	.
AA010	4.00000000	15	.	2.00000000
UP016	2.00000000	16	2.00000000	.
AA011	4.00000000	17	.	2.50000000
UP018	2.00000000	18	2.00000000	.
AA012	4.00000000	19	.	2.79999999
UP035	3.00000000	20	5.00000000-	1.00000001
UP021	2.00000000	21	7.00000000	.
AA013	15.00000000	22	.	1.70000000
UP023	2.00000000	23	2.00000000	.
AA014	15.00000000	24	.	2.09999999
UP008	9.00000000	25	6.00000000-	1.89999999
UP026	1.00000000	26	7.00000000	.
AA015	30.00000000	27	.	2.19999998
UP028	1.00000000	28	1.00000000	.
AA016	30.00000000	29	.	2.39999997
UP030	1.00000000	30	1.00000000	.
AA017	30.00000000	31	.	2.59999996
UP032	1.00000000	32	1.00000000	.
AA018	30.00000000	33	.	2.89999996
AA006	21.00000000	34	8.00000000-	.
UP006	9.00000000	35	12.00000000	.
AA019	24.00000000	36	.	1.29999999
UP005	9.00000000	37	35.00000000	3.19999995

NORMAL,,2

(Problem M-400 Solution, continued)

TOTAL ITERS	NO. ETAS	TP.2 RECS	PHS ROW TYP NO.	CURRENT VALUE	SOLN FEAS	RHS NO.
031	031	000	II 000	68.199997	YES	002

PRIMAL-DUAL SOLUTIONS

J	BETA	I	B	PI
	68.19999860		.	1.00000000
UP001	.	1	9.00000000-	.
UP007	18.00000000	2	5.00000000-	2.10000000
UP010	1.00000000	3	1.00000000-	1.80000001
UP009	9.00000000	4	4.00000000-	.
UP004	2.00000000	5	2.00000000-	.
AA005	15.00000000	6	8.00000000-	.
AA008	31.00000000	7	4.00000000-	.
AA002	4.00000000	8	11.00000000-	.
AA003	9.00000000	9	7.00000000-	.
UP034	2.00000000	10	6.00000000-	.
AA007	25.00000000	11	3.00000000-	.09999999
AA009	36.00000000	12	5.00000000-	2.79999996
AA004	10.00000000	13	4.00000000-	1.00000000
UP014	1.00000000	14	5.00000000	.
AA010	4.00000000	15	.	2.00000000
UP016	2.00000000	16	2.00000000	.
AA011	4.00000000	17	.	2.50000000
UP018	2.00000000	18	2.00000000	.
AA012	4.00000000	19	.	2.79999999
UP035	2.00000000	20	5.00000000-	1.00000001
UP021	2.00000000	21	7.00000000	.
AA013	15.00000000	22	.	1.70000000
UP023	2.00000000	23	2.00000000	.
AA014	15.00000000	24	.	2.09999999
UP008	10.00000000	25	6.00000000-	1.89999999
UP026	1.00000000	26	7.00000000	.
AA015	31.00000000	27	.	2.19999998
UP028	1.00000000	28	1.00000000	.
AA016	31.00000000	29	.	2.39999997
UP030	1.00000000	30	1.00000000	.
AA017	31.00000000	31	.	2.59999996
UP032	1.00000000	32	1.00000000	.
AA018	31.00000000	33	.	2.89999996
AA006	22.00000000	34	8.00000000-	.
UP006	10.00000000	35	12.00000000	.
AA019	25.00000000	36	.	1.29999999
UP005	9.00000000	37	36.00000000	3.19999995

NORMAL,,3

(Problem M-400 Solution, continued)

TOTAL ITERS	NO. ETAS	TP.2 RECS	PHS TYP	ROW NO.	CURRENT VALUE	SOLN FEAS	RHS NO.
031	031	000	II	000	74.599998	YES	003

PRIMAL-DUAL SOLUTIONS

J	BETA	I	B	PI
	74.59999851		.	1.00000000
UP001	.	1	9.00000000-	.
UP007	20.00000000	2	5.00000000-	2.10000000
UP010	3.00000000	3	1.00000000-	1.80000001
UP009	11.00000000	4	4.00000000-	.
UP004	2.00000000	5	2.00000000-	.
AA005	15.00000000	6	8.00000000-	.
AA008	33.00000000	7	4.00000000-	.
AA002	4.00000000	8	11.00000000-	.
AA003	9.00000000	9	7.00000000-	.
UP034	4.00000000	10	6.00000000-	.
AA007	27.00000000	11	3.00000000-	.09999999
AA009	38.00000000	12	5.00000000-	2.79999996
AA004	10.00000000	13	4.00000000-	1.00000000
UP014	1.00000000	14	5.00000000	.
AA010	4.00000000	15	.	2.00000000
UP016	2.00000000	16	2.00000000	.
AA011	4.00000000	17	.	2.50000000
UP018	2.00000000	18	2.00000000	.
AA012	4.00000000	19	.	2.79999999
UP035	.	20	5.00000000-	1.00000001
UP021	2.00000000	21	7.00000000	.
AA013	15.00000000	22	.	1.70000000
UP023	2.00000000	23	2.00000000	.
AA014	15.00000000	24	.	2.09999999
UP008	12.00000000	25	6.00000000-	1.89999999
UP026	1.00000000	26	7.00000000	.
AA015	33.00000000	27	.	2.19999998
UP028	1.00000000	28	1.00000000	.
AA016	33.00000000	29	.	2.39999997
UP030	1.00000000	30	1.00000000	.
AA017	33.00000000	31	.	2.59999996
UP032	1.00000000	32	1.00000000	.
AA018	33.00000000	33	.	2.89999996
AA006	24.00000000	34	8.00000000-	.
UP006	12.00000000	35	12.00000000	.
AA019	27.00000000	36	.	1.29999999
UP005	9.00000000	37	38.00000000	3.19999995

NORMAL,,4

(Problem M-400 Solution, continued)

TOTAL ITERS	NO. ETAS	TP.2 RECS	PHS ROW TYP	NO.	CURRENT VALUE	SOLN FEAS	RHS NO.
034	034	000	II	000	78.99999/	YES	004

PRIMAL-DUAL SOLUTIONS

J	BETA	I	B	PI
	78.99999842		.	1.00000000
UP001	1.00000000	1	9.00000000-	.
UP007	21.00000000	2	5.00000000-	1.09999999
UP010	3.00000000	3	1.00000000-	.79999999
UP009	11.00000000	4	4.00000000-	.
UP004	2.00000000	5	2.00000000-	.
AA005	17.00000000	6	8.00000000-	.
AA008	35.00000000	7	4.00000000-	.
AA002	5.00000000	8	11.00000000-	.
AA003	10.00000000	9	7.00000000-	.
UP034	4.00000000	10	6.00000000-	.
AA007	29.00000000	11	3.00000000-	.09999999
AA009	40.00000000	12	5.00000000-	1.79999995
AA004	11.00000000	13	4.00000000-	.
UP013	1.00000000	14	5.00000000	.00000001
AA010	5.00000000	15	.	.99999998
UP016	2.00000000	16	2.00000000	.
AA011	5.00000000	17	.	1.49999998
UP018	2.00000000	18	2.00000000	.
AA012	5.00000000	19	.	1.79999998
UP020	1.00000000	20	5.00000000-	.
UP021	1.00000000	21	7.00000000	.
AA013	17.00000000	22	.	.69999998
UP023	2.00000000	23	2.00000000	.
AA014	17.00000000	24	.	1.09999997
UP008	13.00000000	25	6.00000000-	.89999997
UP026	1.00000000	26	7.00000000	.
AA015	35.00000000	27	.	1.19999997
UP028	1.00000000	28	1.00000000	.
AA016	35.00000000	29	.	1.39999996
UP030	1.00000000	30	1.00000000	.
AA017	35.00000000	31	.	1.59999994
UP032	1.00000000	32	1.00000000	.
AA018	35.00000000	33	.	1.89999994
AA006	26.00000000	34	8.00000000-	.
UP006	13.00000000	35	12.00000000	1.00000001
AA019	29.00000000	36	.	.29999998
UP005	10.00000000	37	40.00000000	2.19999994

NORMAL,,5

(Problem M-400 Solution, continued)

TOTAL ITERS	NO. ETAS	TP.2 RECS	PHS ROW TYP	NO.	CURRENT VALUE	SOLN FEAS	RHS NO.
035	035	000	II	000	88.799996	YES	005

PRIMAL-DUAL SOLUTIONS

J	BETA	I	B	PI
	88.79999820		.	1.00000000
UP001	1.00000000	1	9.00000000-	.
UP007	26.00000000	2	5.00000000-	.80000001
UP010	7.00000000	3	1.00000000-	.50000001
UP009	15.00000000	4	4.00000000-	.
UP004	2.00000000	5	2.00000000-	.
AA005	18.00000000	6	8.00000000-	.
AA008	40.00000000	7	4.00000000-	.
AA002	5.00000000	8	11.00000000-	.
AA003	10.00000000	9	7.00000000-	.
UP034	4.00000000	10	6.00000000-	.
AA007	34.00000000	11	3.00000000-	.09999999
AA009	45.00000000	12	5.00000000-	1.49999997
AA004	11.00000000	13	4.00000000-	.
UP013	1.00000000	14	5.00000000	.29999999
AA010	5.00000000	15	.	.70000000
UP016	2.00000000	16	2.00000000	.
AA011	5.00000000	17	.	1.20000000
UP018	2.00000000	18	2.00000000	.
AA012	5.00000000	19	.	1.50000000
UP020	2.00000000	20	5.00000000-	.
UP036	4.00000000	21	7.00000000	.29999998
AA013	18.00000000	22	.	.40000000
UP023	2.00000000	23	2.00000000	.
AA014	18.00000000	24	.	.79999999
UP008	18.00000000	25	6.00000000-	.59999999
UP026	1.00000000	26	7.00000000	.
AA015	40.00000000	27	.	.89999999
UP028	1.00000000	28	1.00000000	.
AA016	40.00000000	29	.	1.09999997
UP030	1.00000000	30	1.00000000	.
AA017	40.00000000	31	.	1.29999996
UP032	1.00000000	32	1.00000000	.
AA018	40.00000000	33	.	1.59999996
AA006	31.00000000	34	8.00000000-	.
UP006	18.00000000	35	12.00000000	1.29999999
AA019	30.00000000	36	.	.
UP005	11.00000000	37	45.00000000	1.89999996

NORMAL,,6

(Problem M-400 Solution, continued)

TOTAL ITERS	NO. ETAS	TP.2 RECS	PHS TYP	ROW NO.	CURRENT VALUE	SOLN FEAS	RHS NO.
035	035	000	II	000	98.299997	YES	006

PRIMAL-DUAL SOLUTIONS

J	BETA	I	B	PI
	98.29999801		.	1.00000000
UP001	1.00000000	1	9.00000000-	.
UP007	31.00000000	2	5.00000000-	.80000001
UP010	12.00000000	3	1.00000000-	.50000001
UP009	20.00000000	4	4.00000000-	.
UP004	2.00000000	5	2.00000000-	.
AA005	18.00000000	6	8.00000000-	.
AA008	45.00000000	7	4.00000000-	.
AA002	5.00000000	8	11.00000000-	.
AA003	10.00000000	9	7.00000000-	.
UP034	4.00000000	10	6.00000000-	.
AA007	39.00000000	11	3.00000000-	.09999999
AA009	50.00000000	12	5.00000000-	1.49999997
AA004	11.00000000	13	4.00000000-	.
UP013	1.00000000	14	5.00000000	.29999999
AA010	5.00000000	15	.	.70000000
UP016	2.00000000	16	2.00000000	.
AA011	5.00000000	17	.	1.20000000
UP018	2.00000000	18	2.00000000	.
AA012	5.00000000	19	.	1.50000000
UP020	2.00000000	20	5.00000000-	.
UP036	9.00000000	21	7.00000000	.29999998
AA013	18.00000000	22	.	.40000000
UP023	2.00000000	23	2.00000000	.
AA014	18.00000000	24	.	.79999999
UP008	23.00000000	25	6.00000000-	.59999999
UP026	1.00000000	26	7.00000000	.
AA015	45.00000000	27	.	.89999999
UP028	1.00000000	28	1.00000000	.
AA016	45.00000000	29	.	1.09999997
UP030	1.00000000	30	1.00000000	.
AA017	45.00000000	31	.	1.29999996
UP032	1.00000000	32	1.00000000	.
AA018	45.00000000	33	.	1.59999996
AA006	36.00000000	34	8.00000000-	.
UP006	23.00000000	35	12.00000000	1.29999999
AA019	30.00000000	36	.	.
UP005	11.00000000	37	50.00000000	1.89999996

NORMAL,,7

(Problem M-400 Solution, continued)

TOTAL ITERS	NO. ETAS	TP.2 RECS	PHS TYP	ROW NO.	CURRENT VALUE	SOLN FEAS	RHS NO.
035	035	000	II	000	107.79999	YES	007

PRIMAL-DUAL SOLUTIONS

J	BETA	I	B	PI
	107.79999782		.	1.00000000
UP001	1.00000000	1	9.00000000-	.
UP007	36.00000000	2	5.00000000-	.80000001
UP010	17.00000000	3	1.00000000-	.50000001
UP009	25.00000000	4	4.00000000-	.
UP004	2.00000000	5	2.00000000-	.
AA005	18.00000000	6	8.00000000-	.
AA008	50.00000000	7	4.00000000-	.
AA002	5.00000000	8	11.00000000-	.
AA003	10.00000000	9	7.00000000-	.
UP034	4.00000000	10	6.00000000-	.
AA007	44.00000000	11	3.00000000-	.09999999
AA009	55.00000000	12	5.00000000-	1.49999997
AA004	11.00000000	13	4.00000000-	.
UP013	1.00000000	14	5.00000000	.29999999
AA010	5.00000000	15	.	.70000000
UP016	2.00000000	16	2.00000000	.
AA011	5.00000000	17	.	1.20000000
UP018	2.00000000	18	2.00000000	.
AA012	5.00000000	19	.	1.50000000
UP020	2.00000000	20	5.00000000-	.
UP036	14.00000000	21	7.00000000	.29999998
AA013	18.00000000	22	.	.40000000
UP023	2.00000000	23	2.00000000	.
AA014	18.00000000	24	.	.79999999
UP008	28.00000000	25	6.00000000-	.59999999
UP026	1.00000000	26	7.00000000	.
AA015	50.00000000	27	.	.89999999
UP028	1.00000000	28	1.00000000	.
AA016	50.00000000	29	.	1.09999997
UP030	1.00000000	30	1.00000000	.
AA017	50.00000000	31	.	1.29999996
UP032	1.00000000	32	1.00000000	.
AA018	50.00000000	33	.	1.59999996
AA006	41.00000000	34	8.00000000-	.
UP006	28.00000000	35	12.00000000	1.29999999
AA019	30.00000000	36	.	.
UP005	11.00000000	37	55.00000000	1.89999996

NORMAL,,8

(Problem M-400 Solution, continued)

TOTAL ITERS	NO. ETAS	TP.2 RECS	PHS TYP	ROW NO.	CURRENT VALUE	SOLN FEAS	RHS NO.
035	035	000	II	000	117.29999	YES	008

PRIMAL-DUAL SOLUTIONS

J	BETA	I	B	PI
	117.29999762		.	1.00000000
UP001	1.00000000	1	9.00000000-	.
UP007	41.00000000	2	5.00000000-	.80000001
UP010	22.00000000	3	1.00000000-	.50000001
UP009	30.00000000	4	4.00000000-	.
UP004	2.00000000	5	2.00000000-	.
AA005	18.00000000	6	8.00000000-	.
AA008	55.00000000	7	4.00000000-	.
AA002	5.00000000	8	11.00000000-	.
AA003	10.00000000	9	7.00000000-	.
UP034	4.00000000	10	6.00000000-	.
AA007	49.00000000	11	3.00000000-	.09999999
AA009	60.00000000	12	5.00000000-	1.49999997
AA004	11.00000000	13	4.00000000-	.
UP013	1.00000000	14	5.00000000	.29999999
AA010	5.00000000	15	.	.70000000
UP016	2.00000000	16	2.00000000	.
AA011	5.00000000	17	.	1.20000000
UP018	2.00000000	18	2.00000000	.
AA012	5.00000000	19	.	1.50000000
UP020	2.00000000	20	5.00000000-	.
UP036	19.00000000	21	7.00000000	.29999998
AA013	18.00000000	22	.	.40000000
UP023	2.00000000	23	2.00000000	.
AA014	18.00000000	24	.	.79999999
UP008	33.00000000	25	6.00000000-	.59999999
UP026	1.00000000	26	7.00000000	.
AA015	55.00000000	27	.	.89999999
UP028	1.00000000	28	1.00000000	.
AA016	55.00000000	29	.	1.09999997
UP030	1.00000000	30	1.00000000	.
AA017	55.00000000	31	.	1.29999996
UP032	1.00000000	32	1.00000000	.
AA018	55.00000000	33	.	1.59999996
AA006	46.00000000	34	8.00000000-	.
UP006	33.00000000	35	12.00000000	1.29999999
AA019	30.00000000	36	.	.
UP005	11.00000000	37	60.00000000	1.89999996

GETOFF

GETOFF CALLED FOR AT ITER 035, 000 RECORDS ON TAPE 2

ALL REQUIRED CONTENTS OF MACHINE ON TAPE 5. PULL JOB.

DISTRIBUTION

1. E. D. Arnold
2. R. E. Blanco
3. J. C. Bresee
4. K. B. Brown
5. W. D. Burch
6. F. L. Culler, Jr.
7. D. E. Ferguson
8. H. E. Goeller
9. H. B. Graham
10. A. T. Gresky
11. R. P. Milford
12. J. P. Nichols
13. E. L. Nicholson
14. F. L. Peishel
15. B. N. Robards

16-20. Royes Salmon, U. of Miss.

21. M. J. Skinner
22. W. G. Stockdale
23. J. W. Ullmann
24. W. E. Unger
25. M. E. Whatley
26. Laboratory Records, RC

27-31. Laboratory Records

32-33. Central Research Library

34. Document Reference Section

35-49. DTIE

50. R & D Div., ORO-AEC