RESOURCE LEVELLING

YASUHIKO TAMURA

RESOURCE LEVELLING

A THESIS SUBMITTED

by

Yasuhiko Tamura

IN PARTIAL FULFILMENT OF

THE DEGREE OF

M.Eng.

MEMORIAL UNIVERSITY OF NEWFOUNDLAND

SEPTEMBER, 1974

TABLE OF CONTENTS

Appendices

List of Tables

List of Figures

ACKNOWLEDGEMENTS

I wish to express my sincere gratitude to Professor H.N. Ahuja for suggesting this research problem to me and for his guidance and constructive criticism. During the past two years he has been a source of informed opinion on matters both scientific and personal. I would also like to express my gratitude and appreciation to Mr.M. Iwadan and associates of the Kawashima Textile Mills Ltd. for providing an opportunity for me to study in Canada. I wish to acknowledge Professor E. Moore, Professor M.E. El-Hawary, Professor P.J. Amaria, and Mr. Terry Buckley, for their helpful discussions. Finally, I would like to thank Memorial University of Newfoundland for awarding a scalarship to me, this enabling me to pursue my studies.

ABSTRACT

RESOURCE LEVELLING

For projects where highly skilled workers are needed whose training could take a long time, gradualness in the resource profile could provide the necessary training period. This thesis presents an heuristic procedure for optimal resource levelling achieving a gradual (parabolic) resource distribution. It aims at minimizing the fluctuation of day-to-day resource change using the variance of resource change as the primary index. Mathematically based, this heuristic gives a measure whereby an optimal schedule can be recognized.

Using the computation procedure for this heuristic a computer programme called BACS (Build-up Activity Chain Scheduler) has been developed which yields an optimal schedule for large projects from data obtained from CPM analysis. This programme is both easy to use and has a short computational time.

CHAPTER I

1.1 Introduction

The utilization of resources is an ever present condition of project accomplishment. Resources mean money, labour, equipment, materials, and any other commodity which is expressible in physical units.

Present CPM (Critical Path Method) scheduling and resource allocation are based on one of three possible arrangements:

1. The availability of resources for each date is fixed and from this the minimum project duration is determined.
2. The project duration is fixed and the resources required to meet this duration is determined.
3. The resource availability for each day is a variable, the project duration is fixed, and optimization is achieved by varying the daily resource level in order to keep the total resource as close to the minimum required as possible.

In many projects it is desirable to avoid sharp fluctuations in resource requirements. Recruiting, hiring, and training personnel is costly and time-consuming. It is not feasible to hire and lay-off the work-force according to the fluctuating needs. Therefore, the build-up of the work force should be gradual, that is, the resource profile should gradually rise step by step to a peak and then fall gradually towards project completion. This problem , known as Resource Levelling is studied in this thesis under the constraints of the third type of arrangement, that is the project duration is fixed and the daily resource level is varied to obtain a step by step build-up and a gradual lay-off keeping the total resource requirement to a minimum.

1.2 Two Approaches to Resource Levelling

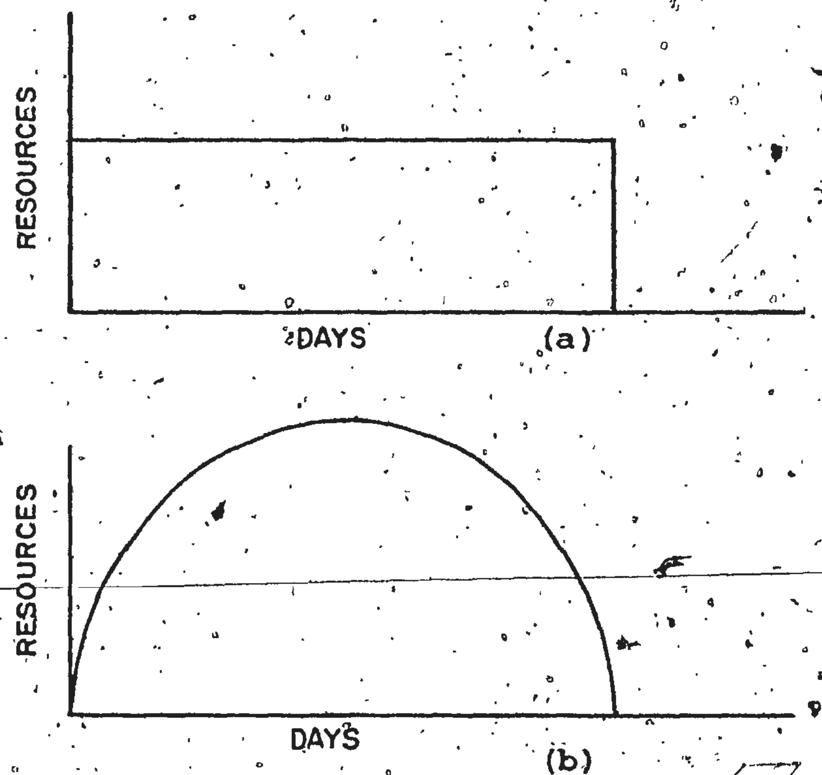
Much thought and study has been devoted to the resource levelling problem. Two approaches to resource levelling are in existence.

The first approach keeps the work force to a minimum and attempts to achieve a flat level profile (see Figure I-1(a)) previously considered by De-Witte (Data Processing Science/Engineering, 2.2,29 (1964)), Levy et al (Naval Research Logistics Quarterly, 9,1,37-44, (1962)), IBM (IBM Programme Number 5734-xp4), and Burgess et al (J.Indust. Eng., XIII, (1962)). This is important for a project where camp facilities have to be provided for the workers or where work progresses on a continuous basis and the size of the work crew is governed by peak requirements. By keeping the level of resources required on any day to the minimum, the cost of providing the camps or maintaining a work crew can be kept to the minimum. Also in the first approach, many resources are required on the starting date and have to be cut off abruptly at project completion.

It is not always possible to maintain a constant work force, particularly in one-time jobs. It is easier, however, to achieve a smooth build up of resources followed by a gradual tapering off with a minimum fluctuations during the project.

The second approach gradually builds up the work force to a peak and then gradually lays them off towards project completion (see Figure I-1(b)) work in this area has been done by Martino (Project Management and Control Series III, (1965)). This approach is useful for projects where highly skilled workers are needed whose recruitig and, or training would take a long time. The gradualness in such a case would provide the necessary recruiting and training period.

Figure I-1(a) can be conceived as the result of vertical compression of the resource profile, which results in a rectangule. Similary, Figure I-1(b) can be conceived as the result of a horizontal compression resulting in a parabola.



Rectangular and Parabolic Resource Allocation

Figure I-1

1.3 Methods of Resource Levelling

Present methods of resource levelling can be divided into three areas: a) handsmoothing, b) mathematics, c) heuristics.

In the hand-solution, use is made of the human's ability to take a comprehensive look at a resource profile and to detach particular jobs causing unwanted peaks that can be shifted to fill up some valleys elsewhere in the schedule. This works fine for small projects but for larger and more complex projects, handsmoothing by juggling slack jobs can be very difficult and time consuming. Just drawing and redrawing the resource profiles can be a tedious task as the number of jobs increases. If there were several resources to level, not only would there be more resource profiles to keep track of but also the technological interrelationships of jobs would make simultaneous smoothing of all resources a tricky proposition.

Mathematical approaches have not been found practical for large scale projects. The most recognized mathematical approach to the resource levelling problem can be seen in "Variation in Activity Level on a Cyclical Arrow Diagram".[2] The approach, which mathematicians refer to as a large combinatorial problem, leads to a rectangular resource profile. In this case, there are a very large number of combinations of activity start times - too large to enumerate even with a big computer. Unfortunately, analytical techniques are computationally impractical for most real-life problems of this kind.

Several kinds of heuristic methods using vertical and horizontal compression have been presented for the resource levelling problem, some of these will be discussed later, and several heuristic type computer programmes have also been written.[2,4,6,10,16,17] Their objective is to smooth resource requirement by shifting slack jobs beyond periods of peak requirement. Some of these essentially replicate what a human scheduler would do if he had enough time. Others make use of unusual devices or procedures designed especially for the computer. All of them, of course, depend for their success on the tremendous capabilities of electronic data processing machines. However, no attempt has been made to provide a mathematical base for these heuristic approaches.

1.4 Present State of the Art

R.L.Martino presented one heuristic approach to resource levelling problem in his "Allocation and Scheduling Resources.[9]" The fundamental algorithm is that the resource profile is built up towards the peak and is demolished towards the end. After all critical activities have been scheduled the idea is to postpone the start of non-critical activities so that each activity attains its latest start time and becomes critical. This is done until a peak of resources is reached. This peak is evident when no additional resources are required to maintain the same level of activity as required on the previous day. The idea of levelling is to maintain this peak as long as possible. In order to accomplish this, other jobs that are non-critical must be scheduled at their earliest start time. This will result in a step by step dropping of resources. Activities are pushed forward from the front end and backwards from the back end on the resource profile, forcing a peak.[1]

Martino's algorithm attempts to achieve a gradual resource profile. However, it has no measure of gradualness, makes no attempt to achieve optimality, has no mathematical bases, and a computer programme is not developed. This approach once all activities are scheduled does not look back at the final resource profile to see if any further improvements can be made by re-scheduling activities where unwanted peaks in the resource profile occur. Also, since the maximum resource requirement depends upon the push forward towards a peak until it is reached, and the push backward beyond the peak, there is no control over the height of the peak. This may result in a very high requirement in the region of the peak.

Another heuristic approach is the MS (for Multi-ship Multi-shops, Workload Smoothing Programme by F.K.Levy, G.L.Thompson and J.D.Wiest)[8] originally designed to smooth manpower requirement in naval shipyards.

In many project operations, manpower requirement may vary greatly from day to day depending on the arrival of jobs and how big they are. Since crew size generally must be sufficient to meet maximum manpower requirements, a schedule which reduces peak loads by increasing usage during slack periods would allow smaller shop sizes and hence reduce labor expense. The MS programme attempts to do this by scheduling all jobs at their earliest start times and then shifting some of them that occur during peak periods to later slack periods.

First the programme sets an early start schedule along with total slack values for all jobs calculated by regular network procedure. Then "Trigger levels" or resource limits are set, one unit (man) below the peak requirement in each of the shops, and it attempts to reschedule the jobs so that peak requirements do not exceed the trigger levels. If the trigger levels of each shop are met then all of them are reduced one unit again, and the process of loading and shifting is repeated. The trigger level is lowered, one unit at a time, until no further reduction is possible. Then the next most expensive shop is examined and the trigger levels reduced to the lowest feasible point. This operation is repeated until no trigger levels can be further reduced and a final schedule is reached.

This heuristic leads to a final rectangular resource profile using a computer programme having a short computation time. As this heuristic does not seem to have been developed from a mathematical base, nothing can be said about the optimality of the final solution.

A.R.Burgess and J.B.Killebrew presented a paper "Variation in Activity Level on a Cyclical Arrow Diagram"[2]. The basic theme is a consideration of smoothing out the level of resources as much as possible. To reduce the fluctuations of resource requirement, they proposed a method of minimizing the sums of squares of resource requirement through the project duration. This technique is known as "method of least squares". Final optimum solution is determined by trying all feasible combinations of slack variables of activities.

Burgess's heuristic, using the mathematical base of the minimization of the sum of squares of resource requirement, will achieve an optimum rectangular resource profile. The computer programme developed, although very easy to use, takes an extremely long computational time for small networks and would be impractical to use for large scale networks.

IBM has developed a programme called RAP (Resource Allocation Processor as part of the Project Management System) which attempts as one of its objectives to schedule a project to finish by the schedule duration without exceeding a set resource level. If this primary resource level can not be fulfilled the programme relaxes the resource limit to a secondary user defined resource level which it will now attempt to schedule. If the schedule is not completed after a set computation time the programme terminates.

IBM's programme yields a final near rectangular resource profile without a mathematical base, hence no measure of optimality is achieved. However, the programme is easy to use and has a short computation time,

RAMPS is another technique for resource allocation and multiproject scheduling which was developed by J.Moshman, J.Johnson and Madalyn Larsen. RAMPS has a rather elaborate set of heuristics for choosing jobs to be scheduled on a given day. Not only the job slack but also such factors as work continuity, cost of idle resources, number of successors to an eligible job, and several others, are considered but are also given weights to reflect their relative importance. All feasible schedules for each shceduling period are examined, and the one is chosen which rates highest by the above weighted criteria factors.

RAMPS, developed without a mathematical base, has no measure for gradualness, whereby an optimal solution, when achieved can be identified.

1.5 Problem Statement

Most of the heuristics discussed earlier develop a rectangular resource distribution. Such a heuristics will not be beneficial in a case where gradualness of recruiting is the objective. Martino's heuristic, the only one having a parabolic distribution, has many major drawbacks as pointed out earlier. Therefore, it is necessary to develop a new heuristic with a parabolic distribution.

No attempt has been made to develop this heuristic procedure for resource levelling with a mathematical base. First, a mathematical model for resource levelling is needed where both the total resource requirement and project duration are fixed although the daily resource level is variable. Then a heuristic method of allocating resources can be developed so that the distribution of the total resources will be as gradual as possible. This requires a definition of gradualness where-by a measure of the gradualness of resource change is defined. Thus in the search for an optimal solution, the optimality can be measured by the gradualness of resource change and the necessary cut-off point can be provided the gradualness indicating that the optimal solution has been achieved.

1.6 Design Criteria

A resource levelling procedure is needed to generate resource schedules:

(1) which is based on a mathematical model leading to a parabolic resource profile where an objective function is minimized or maximized subject to constraints, in order to have a gradual build up of resources followed by a gradual tapering off, with minimal fluctuations in the resource level.

(2) which makes optimal use of the resources within the fixed project duration.

(3) which provides a measure for the gradualness of resource change on a project.

(4) which is a computer based procedure for a large scale network and the solution is achieved within an economical computation time.

(5) which is easy for the user to use, by inputting the project CPM network and resource requirement.

CHAPTER II

MATHEMATICAL APPROACH TO RESOURCE LEVELLING

This chapter presents the mathematical approach to the resource levelling problem in three sections. First, the theoretical approach to solve a general resource levelling problem is studied and the optimum function is introduced without taking activity constraints into account. Second, optimization of a CPM network based project is studied and a mathematical model is formulated by taking activity constraints into account. Third, an heuristic algorithm based on the mathematical model is presented for resource levelling of a large scale CPM network.

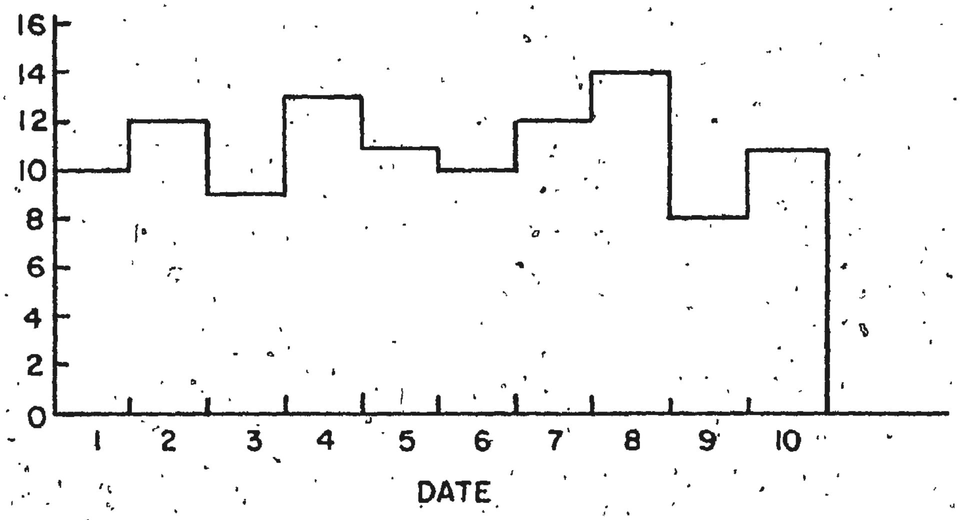
2.1. Theoretical Approach to Resource levelling problem

For a project with fixed duration and a variable daily resource level there can be a large number of resource profiles. Consider a project with a total resource requirement of 110 and a project duration of 10 days. Two, out of many possible resource profiles are shown in Figure II-1 for this project.

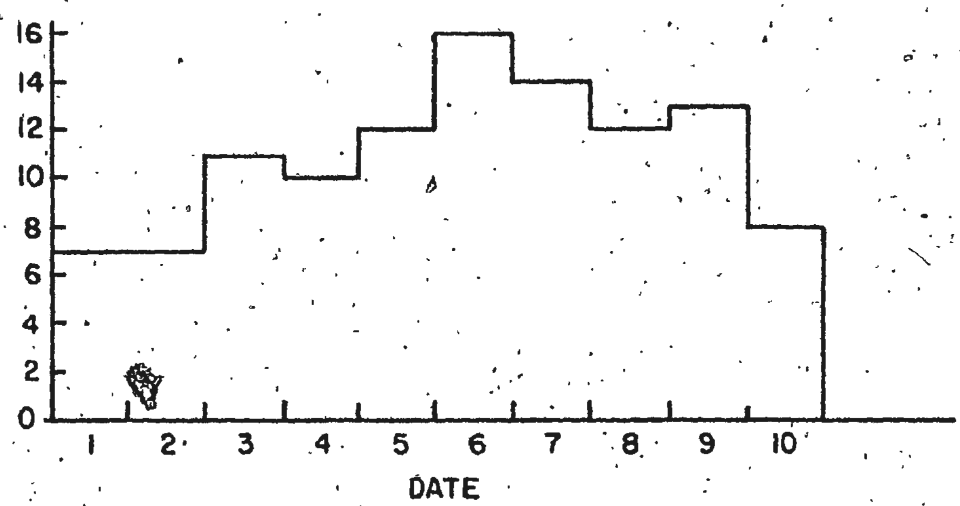
Two questions that immediately arise are:

1. What is the optimum resource profile for a project, given its duration and total resource requirement?
2. Which resource allocation profile has the most gradual aquisition and lay-off of resources?

This section will consider these two questions to achieve a mathematical solution.



Resource Allocation Profile
Figure II-1(a)



Resource Allocation Profile
Figure II-1(b)

2.1.1 The Concept of the Objective Function

The present objective is to achieve a gradual acquisition and lay-off of resources in a project. Suppose F(t) is the resource requirement function of date t. Then the slope of the curve F(t), which is the differential function dF(t)/dt, is a measure of whether or not the acquisition and lay-off of resources is gradual. Therefore to achieve the most gradual curve of the resource requirement function [F(t)] throughout the project duration, a minimization of the fluctuation of dF(t)/dt is required.

The function dF(t)/dt is defined as the resource change function and then the variance of dF(t)/dt can be taken as a measure of fluctuation. Hence the objective function can be achieved by minimizing the variance of the resource change function.

2.1.2 Formulation of the Objective Function

The following definitions are made:

N = project duration,

W = total resource requirement,

x(k) = resource change on date k,

r(k) = resource requirement on date k.

Total resource requirement (W) can now denoted as[14]

$$W = \sum_{k=1}^{N+1} r(k). \tag{2.1}$$

If the resource requirement is defined as

$$r(k) = \sum_{j=1}^{k} x(j), \tag{2.2}$$

then from Eqns. (2.1) and (2.2), the total resource requirement (W) can be rewritten as

$$W = \sum_{k=1}^{N+1} \sum_{j=1}^{k} x(j). \tag{2.3}$$

As an example, Table II-1 shows the values of x(k), r(k) and W obtained from Figure II-1(a).

k	1	2	3	4	5	6	7	8	9	10	11
$x(k)$	10	2	-3	4	-2	-1	2	2	-6	3	-11
$r(k)=\sum_{j=1}^{k} x(j)$	10	12	9	13	11	10	12	14	8	11	0
$W=\sum_{k=1}^{N+1} \sum_{j=1}^{k} x(j)$	10	22	31	44	55	65	77	91	99	110	110

Resource Scheduling

Table II-1

As resources are not to be left over at the end of the project,

$$r(N+1) = \sum_{k=1}^{N+1} x(k) = 0 \qquad (2.4)$$

Equations (2.3) and (2.4) can now be considered as constraints in the present model.

Recall earlier, the objective was defined as the minimization of the fluctuation of the resource change function [dF(t)/dt]. The objective function may now be defined as the minimization of the variance of the resource change function, which is designated as,

$$Z = \frac{\sum_{k=1}^{N=1} [x(k) - \bar{x}]^2}{N}, \tag{2.5}$$

where $\bar{x}$ is the mean of the resource changes and is formulated as,

$$\bar{x} = \frac{\sum_{k=1}^{N+1} x(k)}{N}. \tag{2.6}$$

By substituting the value of $\sum_{k=1}^{N+1} x(k)$ from Eqn. (2.4) into Eqn. (2.6), $\bar{x}$ is obtained as equal to zero. As N, the project duration, is constant and the object is to minimize Eqn. (2.5), N can be dropped from the equation without effecting the results. Now the objective function can be written as,

$$Z = \sum_{k=1}^{N+1} x(k)^2 \qquad (2.7)$$

By changing (N+1) to N, the following may be obtained

Minimize

$$Z = \sum_{k=1}^{N} x(k)^2 \qquad (2.8)$$

$$= x(1)^2 + x(2)^2 + ---- + x(N)^2,$$

subject to

$$W = \sum_{k=1}^{N} \sum_{j=1}^{k} x(j) \qquad (2.9)$$

$$= N\, x(1) + (N-1)\, x(2) + ------$$

$$------- + 2x(N-1) + x(N),$$

and

$$0 = \sum_{k=1}^{N} x(k) \qquad (2.10)$$

$$= x(1) + x(2) + ---- + x(N-1) + x(N).$$

Notice that the Objective function (2.8) is a quadratic and both constraints eqns. (2.9) and (2.10) are equality linear functions.

2.1.3 Optimization of Resource Changes

The solution of the objective function is based upon the use of Lagrange multipliers. Since the principal provides an important theoretical background to the present discussion, a brief introduction into the use of Lagrange multipliers follows.

The use of Lagrange multipliers can be illustrated by considering the general problem of finding the extreme points of $Z = f(x,y)$ subject to the constraint $g(x,y)$ There are three equations,

$$\frac{\partial}{\partial x} f(x,y) = f_x(x,y) = 0 \qquad (2.11)$$

$$\frac{\partial}{\partial y} f(x,y) = f_y(x,y) = 0 \qquad (2.12)$$

and

$$g(x,y) = 0 \qquad (2.13)$$

This system of equations is somewhat difficult to solve since there are more equations than the unknowns. Using Lagrange multipliers, and unknown λ (artificial), the Lagrange expression for the general problem is

$$L(x,y,\lambda) = f(x,y) + \lambda\, g(x,y) \qquad (2.14)$$

Inspection of Eqn. (2.14) indicates that L is a function of the three variables x, y, λ. The necessary conditions for maximum or minimum points of this function are the three equations

$$\frac{\partial L}{\partial x} = 0$$
$$\frac{\partial L}{\partial Y} = 0 \qquad (2.15)$$
$$\frac{\partial L}{\partial \lambda} = 0$$

These three equations can be rewritten as,

$$\frac{\partial L}{\partial X} = \frac{\partial}{\partial X}(f+\lambda g)$$
$$= f_x + \lambda g_x ,$$
$$\frac{\partial L}{\partial Y} = \frac{\partial}{\partial Y}(f+\lambda g)$$
$$= f_y + \lambda g_y ,$$
$$\frac{\partial L}{\partial \lambda} = \frac{\partial}{\partial \lambda}(f+\lambda g)$$
$$= g .$$

The last equation above is actually the constraint equation. If an extreme point (x_m, y_m) of L is found, it will satisfy the constraint equation. Equation (2.14) can now be rewritten as,

$$L(x_m, y_m, \lambda) = f(x_m, y_m) + \lambda g(x_m, y_m)$$
$$= f(x_m, y_m) , \qquad (2.16)$$

where $g(x_m, y_m) = 0$. The values of L and f are the same at the maximum and minimum points of L.

Applying Lagrange multipliers to the objective function (2.8) and its constraining equations (2.9) and (2.10), and using the unknown λ (artificial variable), the Lagrangian expression can be written as, where $\bar{x}$ and $\bar{\lambda}$ are vectors:

$$L(\bar{x}, \bar{\lambda}) = \sum_{k=1}^{N} x(k)^2 + \lambda_1 [\sum_{k=1}^{N} \sum_{j=1}^{k} x(j) - W] + \lambda_2 \sum_{k=1}^{N} x(k). \quad (2.17)$$

The necessary condition for minimum points of this function are the following equation sets:

$$\begin{aligned}
\frac{\partial L}{\partial x(1)} &= 2x(1) + N\lambda_1 + \lambda_2 = 0 \\
\frac{\partial L}{\partial x(2)} &= 2x(2) + (N-1)\lambda_1 + \lambda_2 = 0 \\
&\vdots \\
\frac{\partial L}{\partial x(i)} &= 2x(i) + [N-(i-1)]\lambda_1 + \lambda_2 = 0 \\
&\vdots \\
\frac{\partial L}{\partial x(N)} &= 2x(N) + \lambda_1 + \lambda_2 = 0
\end{aligned} \quad (2.18)$$

and

$$\frac{\partial L}{\partial \lambda_1} = W - \sum_{k=1}^{N} \sum_{j=1}^{k} x(j) = 0$$

$$\frac{\partial L}{\partial \lambda_2} = \sum_{k=1}^{N} x(k) = 0 \qquad (2.19)$$

Using Eqn. set (2.18), $x(k)$, where $k = 1$ to N, can be solved in terms of λ_1 and λ_2.

$$x(1) = -1/2 \; [\; N \lambda_1 + \lambda_2 \;]$$

$$x(2) = -1/2 \; [\; (N-1) \lambda_1 + \lambda_2 \;]$$

$$\vdots \qquad (2.20)$$

$$x(i) = -1/2 \; [\; (N-(i-1) \lambda_1 + \lambda_2 \;]$$

$$\vdots$$

$$x(N) = -1/2 \; [\; \lambda_1 + \lambda_2 \;]$$

Now from Equations (2.18) and (2.19), using simple calculations, λ_1 and λ_2 can be written as,

$$\lambda_1 = \frac{-24\ W}{(N-1)\ N\ (N+1)},$$

$$\lambda_2 = \frac{12\ W}{(N-1)\ N}. \tag{2.21}$$

Substituting the values of λ_1 and λ_2 from (2.21) into Equation set (2.20), x(k) can be rewritten as,

$$x(1) = \frac{6W(N-1)}{(N-1)\ N\ (N+1)},$$

$$x(2) = \frac{6W(N-3)}{(N-1)\ N\ (N+1)},$$

$$\vdots$$

$$x(i) = \frac{6W(N+1-2i)}{(N-1)\ N\ (N+1)}, \tag{2.22}$$

$$\vdots$$

$$x(N) = \frac{6W(1-N)}{(N-1)\ N\ (N+1)}.$$

Recall, earlier for simplicity N+1 was set equal to N, now N is set to its true value N+1. Then the general form for the optimum resource change for each date can be written as,

$$x(k) = \frac{6W\,(N+2-2k)}{N(N+1)(N+2)} \qquad (2.23)$$

And the optimum resource requirement on each date is a quadratic function of the form,

$$r(k) = \sum_{j=1}^{k} x(j) = \frac{6W\,[(N+1)k - k^2]}{N(N+1)(N+2)} . \qquad (2.24)$$

2.1.4 Numerical Example

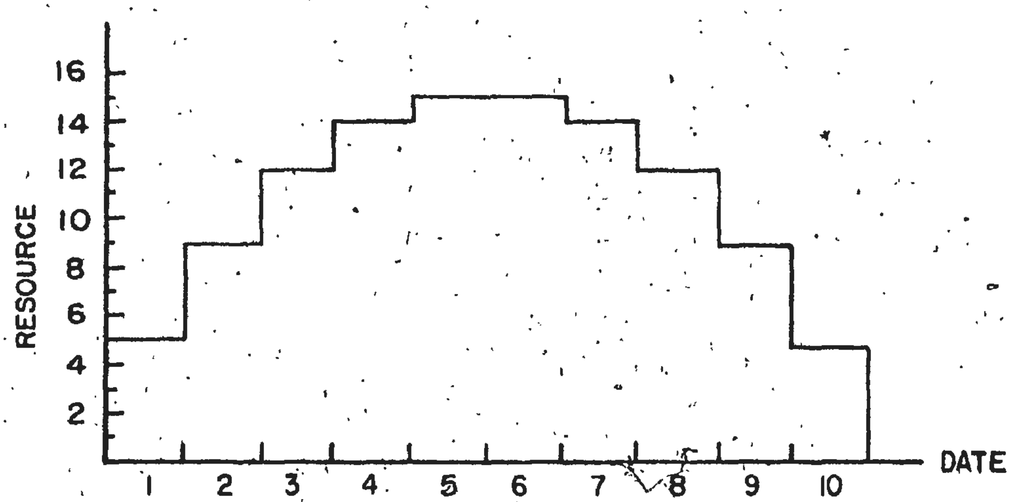
Suppose it is desired to obtain the optimum resource scheduling when the total resource requirement (W) is 110 and the duration (N) is 10 days. This is the same as the example at the begining of this section.

Using the developed equations (2.23) and (2.24), x(k), the resource change on date k, r(k), the resource requirement on date k, and W, the total resource requirement, are calculated. The optimum resource schedule for this example is shown in Table II-2. Figure II-2 is the optimum resource profile.

k	date	1	2	3	4	5	6	7	8	9	10	11
x(k)	resource change	5	4	3	2	1	0	-1	-2	-3	-4	-5
r(k)	resource requirement	5	9	12	14	15	15	14	12	9	5	0
W	total resource	5	14	26	40	55	70	84	96	105	110	110

Optimum Resource Schedule

Table II-2



Optimal Resource Profile

Figure II-2

2.2 Project Model Based on CPM Network Analysis

This model will include CPM network constraints which were not applied in the previous model. It is therefore necessary to introduce the following symbols.

A(j) = activity j not on the critical path (non-critical activity [1 j K])

a(i,j) = resource change for non-critical activity number j on date i [1 i N+1]

D(j) = duration of activity A(j)

EF(j) = earliest finish of activity A(j)

ES(j) = earliest start of activity A(j)

FL(j) = total float for activity A(j), ie, the duration between its ES(j) and the LS(j)

K = number of non-critical activities

L(i) = resource change on date i for a critical activity (This change cannot be varied.)

LS(j) = latest start date of activity A(j)

N = total project duration

R(j) = resource requirement of activity A(j)

S(j) = scheduled start date of activity A(j)

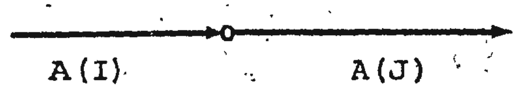
X(j) = start date of activity A(j) relative to the earliest start date [ES(j)], for example, when X(j)=0, activity A(j) is started on ES(j) and when X(j) = LS(j) - ES(j), activity A(j) is started on the latest start date LS(j). This is called the <u>lag</u>.

Y(i) = resource change on date i caused by both critical and non-critical activities.

In all project models based on CPM network theory it is essential to consider the activity constraints of the project. Now the mathematical model will be discussed taking into account these activity constraints, to produce a gradual aquisition and lay-off of resources. To achieve this, the loading or scheduling of non-critical activities are varied. The non-critical activities:

1) are finite,
2) are of fixed integer duration,
3) require a fixed amount of resources,
4) are subject to a set of precedence relationship which satisfies permissible activity ordering,
5) once started are not interrupted till completion.

Now formulating this into the problem, let A(I) and A(J) be sequential activities.



Then the earliest starting date of activity A(J) is

$$ES(J) = ES(I) + D(I), \tag{2.25}$$

and the schedule starting date of activity A(J) is

$$S(J) \geq S(I) + D(I). \tag{2.26}$$

Consider for the moment the starting date of activity A(j) relative to the earliest start date, X(j), which can be written as,

$$X(j) = S(j) - ES(j), \tag{2.27}$$

now by definition,

$$X(j) \leq LS(j) - ES(j). \tag{2.28}$$

The expression in the right half side of equation (3.4) is the total float, i.e.,

$$FL(j) = LS(j) - ES(j). \tag{2.29}$$

So that the duration of the lag between the schedule start date and the earliest start date, X(j), must be between 0 and the total float written as,

$$0 \leq X(j) \leq FL(j). \tag{2.30}$$

Now a relationship between the lag of activity A(I) and the lag of activity A(J) can be formulated as,

$$X(I) \leq X(J). \tag{2.31}$$

Leaving this, the generalized expression for resource changes for non-critical activities A(j) and critical path resource changes for days i=1 to N+1, including resource requirements and duration for activities A(j)'s are tabulated in Table II-3.

i \ A(j)	A(1)	A(2)	-	A(K)	L(i)
1	a(1,1)	a(1,2)	-	a(1,K)	L(1)
2	a(2,1)	a(2,2)	-	a(2,K)	L(2)
-	-	-	-	-	-
-	-	-	-	-	-
N	a(N,1)	a(N,2)	-	a(N,K)	L(N)
N+1	a(N+1,1)	a(N+1,2)	-	a(N+1,K)	L(N+1)
R(j)	R(1)	R(2)	-	R(K)	
D(j)	D(1)	D(2)	-	D(K)	

Generalized Expression for Resource Change

Table II-3

The total resource cahnge for each date is given by,

$$Y(i) = \sum_{j=1}^{K} a(i,j) + L(i) \qquad (2.32)$$

and

$$\sum_{i=1}^{N+1} Y(i) = 0, \qquad (2.33)$$

this is the same as constraint (2.10) of the mathematical model in Section 1.

For each activity A(j), there are only two resource changes,

$$a[\, S(j),\, j \,] = R(j), \qquad (2.34)$$

$$a[\, S(j)+D(j),\, j \,] = - R(j) \qquad (2.35)$$

The total resource requirement for the project is

$$\sum_{j=1}^{K} R(j)D(j) + \sum_{p=1}^{N} \sum_{i=1}^{p} L(i) = W \qquad (2.36)$$

this is the same as constraint (2.9) of the mathematical model in Section 1.

The objective function is the minimization of the variance of resource change, that is, the minimization of the sum of squares of resource change for each day. Thus the following relation can be written,

$$\hat{Z} = \sum_{j=1}^{K} Y(i)^2$$

$$= \sum_{i=1}^{N+1} \left[\sum_{j=1}^{K} a(i,j) + L(i) \right]^2 . \quad (2.37)$$

This can be summarized taking into account the activity constraints in mathematical form as,

Minimize

$$Z = \sum_{i=1}^{N+1} \left[\sum_{j=1}^{K} a(i,j) \quad + L(i) \right]^2 \qquad (2.37)$$

subject to,

$$0 \leq X(j) \leq FL(j) \qquad (2.30)$$

and

$$X(I) \leq X(J) \qquad (2.31)$$

under Equations (2.26), (2.27), (2.29), (2.30), (2.32), (2.34), and (2.35).

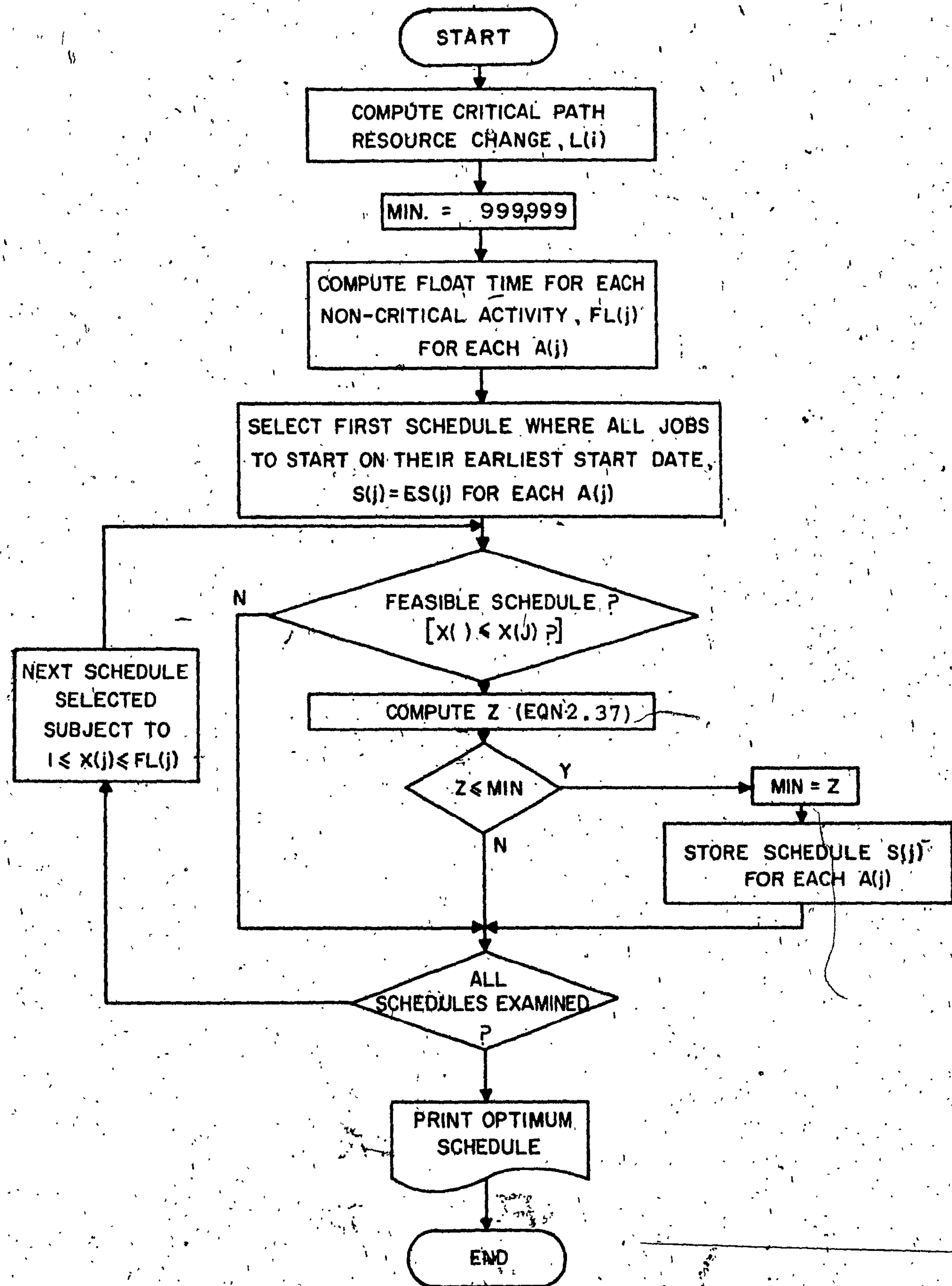
2.2.1 Optimum Resource Levelling on a Simple CPM Network

As an illustration of the procedure of resource levelling to achieve the optimum gradual resource profile a Flow Chart is shown in Figure II-3.

To illustrate the method of minimizing variance of resource change as explained in the previous section and illustrated in the flow diagram, a small computer programme, BASS (Build All Starts Schedule), included in Appendix A, has been developed to achieve the optimum resource profile for this project. It has been applied to a small project, CPM network for which is shown in Figure II-4.

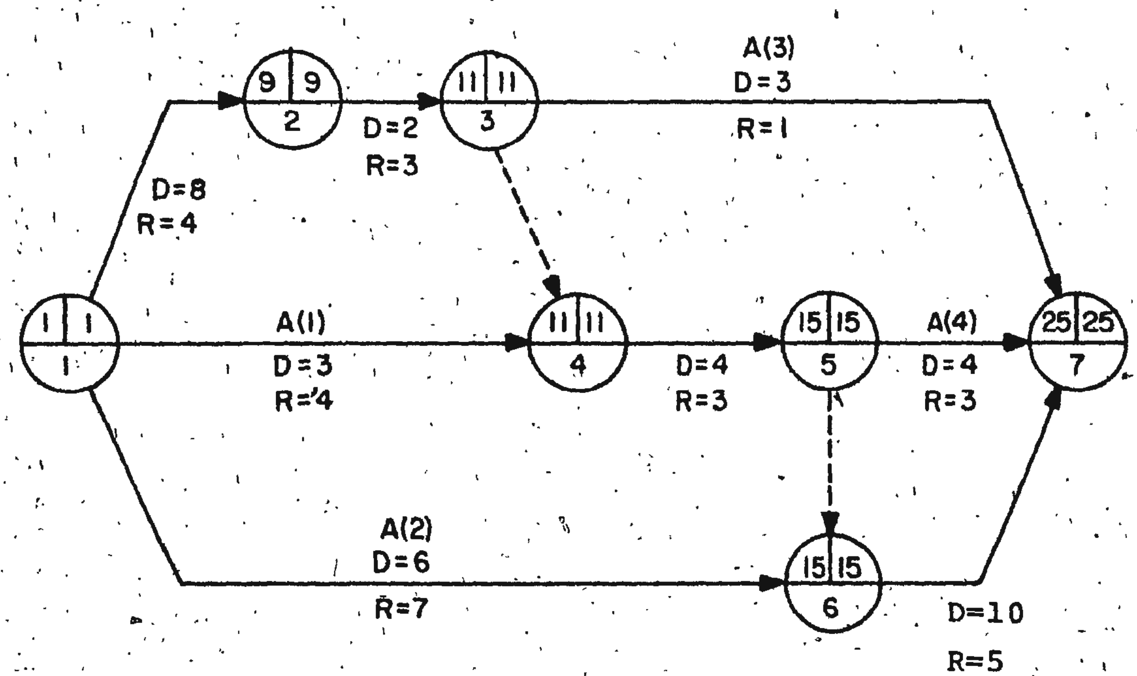
Table II-4 is the bar chart for this network. The firm lines denote the earliest start date [ES(j)] and the dotted lines show the latest start date [LS(j)] for each non-critical activity [A(j)]. Floats [FL(j)'s] of the activities A(1), A(2), A(3), and A(4) are 7, 8, 11, and 6 days respectively.

These non-critical activities can start at any time between their earliest and latest starts and their start times can result in 3696 combinations which are made using the generalized expressions of Table II-4. After developing a resource change table and examining all the combinations for minimum variance of resource change, an optimum solution is determined by the computer programme.



Flow Chart for Bass

Figure II-3



Numerical Example Network

Figure II-4

i \ A(j)	A(1)	A(2)	A(3)	A(4)	L(i)
1					4
2					
3					
4					
5					
6					
7					
8					
9					-1
10					
11					
12					
13					
14					
15					2
16					
17					
18					
19					
20					
21					
22					
23					
24					
25					-5
D(j)	3	6	3	4	

Bar Chart

Table II-4

i \ A(j)	A(1)	A(2)	A(3)	A(4)	L(i)	Y(i)	$Y(i)^2$
1					4	4	16
2							
3							
4							
5							
6	4					4	16
7							
8							
9	-4	7			-1	2	4
10							
11							
12							
13							
14							
15		-7		3	-2	-2	4
16							
17							
18							
19			1	-3		-2	4
20							
21							
22			-1			-1	1
23							
24							
25					-5	-5	25
R(j)	4	7	1	3			
D(j)	3	6	3	4			

Optimum Distribution

Table II-5

A(j)	D(j)	R(j)	ES(j)	SCHEDULED DATE
A(1)	3	4	1	6
A(2)	6	7	1	9
A(3)	3	1	11	19
A(4)	4	3	15	15

Optimum Schedule

Table II-6

Table II-5 shows the resource change for the optimum solution. The value of Z (Equation (2.37)) is 70 which is obtained from $4^2 + 4^2 + 2^2 + (-2)^2 + (-2)^2 + (-1)^2 + (-5)^2$. Table II-6 is the optimum schedule. The critical activities are not included in the table, the reason being that each of these activities has only one start date which has to be its scheduled date.

Figure II-5 is the resource profile for the optimum solution. The resources are built up to a peak of 10 and then laid away as their need no longer exists. There are no valleys in the solution.

Optimum Resource Profile

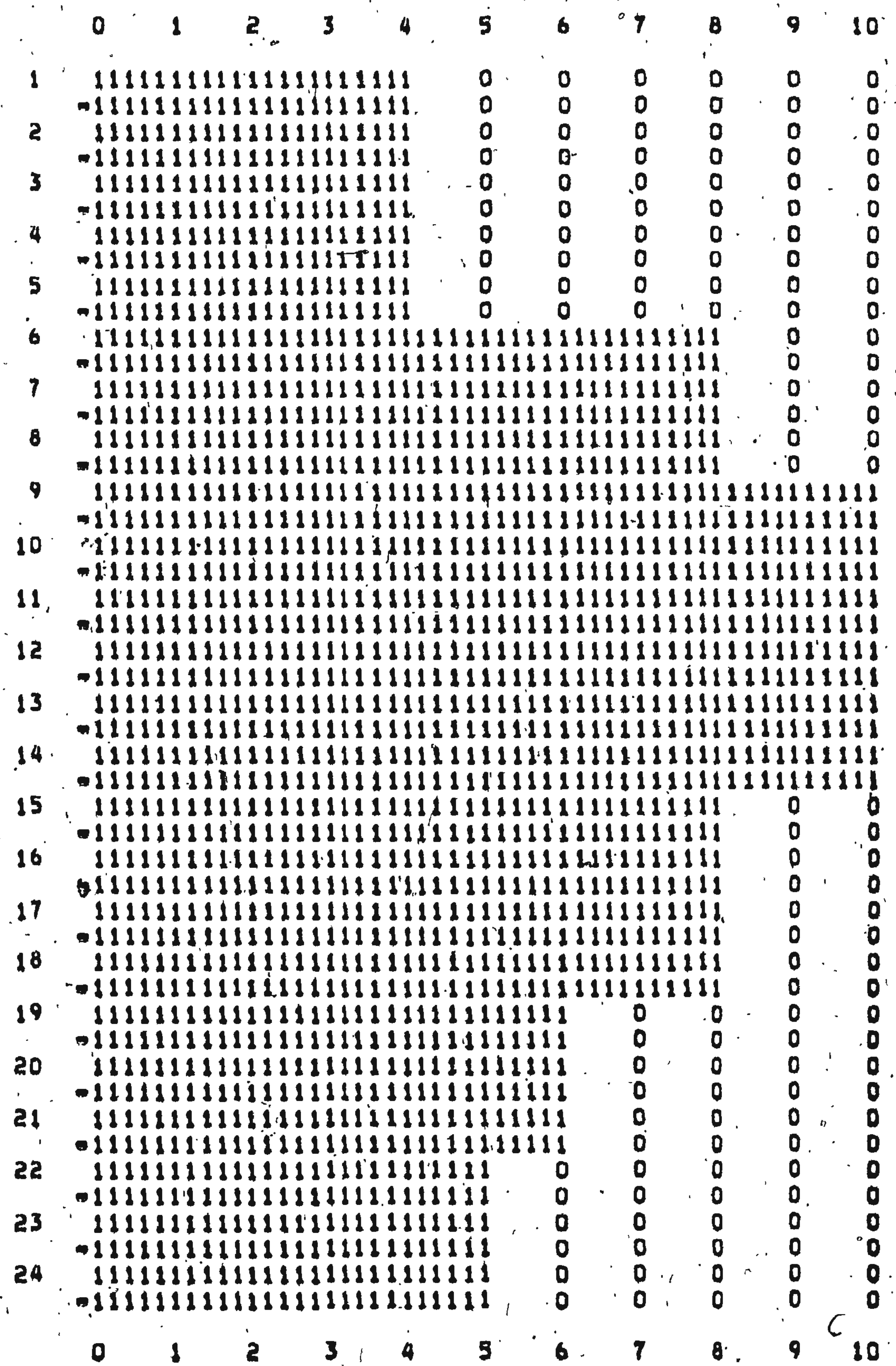


Figure II-5

2.2.2 Impracticality of Achieving the Optimum Resource Profile

Section 2.1 presented the conceptual basis for minimizing variance of resource change to obtain an optimum solution to a general resource levelling problem. This section presented a mathematical model for the network based projects. The model uses the objective function developed in Section 2.1. Resource levelling for network based projects is a large combinatorial problem in which a large number of combinations are required to be investigated to arrive at the final solution. As the size of the network becomes large, it becomes infeasible even using a very large computer to solve such a problem.

An heuristic algorithm is therefore required to obtain a level resource profile for large network projects within an economical computation time. The constraint on computation time would cut down the process of search for the optimum solution by use of the heuristics but in such a case, an optimal (closest to the optimum rather than the mathematically established optimum) instead of an optimum solution will have to be accepted.

2.3 Heuristic Algorithm for Large Networks

The problem to be considered in an heuristic algorithm approach for resource levelling for large scale networks is in the reduction of the number of combinations which are formed by the many possible start times of a large number of non-critical activities in a network. A step by step explanation of the algorithm follows:

STEP 1: All activity chains in the network are identified. An activity chain is defined as the set of sequential non-critical activities whose primary slacks are equal. Such a set of activities is regarded as a single unit in the system. Inside the activity chain, a succeeding activity immediately follows its preceding activity. This means that there is no time-lag between two sequential activities.

STEP 2: All critical activities are scheduled first and their resource change is recorded.

STEP 3: Priority rules* are applied to select the top priority activity chain to be scheduled.

STEP 4: For each day from the earliest start of the chain to its latest start, the variance of resource change (Equation 2.37) is calculated for the combined resource profile of the critical activities, previous scheduled chains if any, and

* Priority rules are explained later.

this chain. The date which gives the minimum variance is selected and the resource change of this activity chain is recorded.

STEP 5: STEP 3 and STEP 4 are repeated for all the chains, always computing the total variance for the critical activities, already scheduled and the chains under consideration, until all activity chains are scheduled.

STEP 6: After scheduling all activity chains, the schedule is looked over again and if some peaks and valleys are encountered in the resource profile, an effort is made to eliminate them with the objective of reducing the variance of resource change.

<u>Priority Rules</u> are defined to obtain a priority whereby the scheduling of activity chains can be ordered to achieve a first resource profile near to an optimal profile. Priority rules are as follows:

P1 = earliest start ES(j)
[high priority is assigned to the chain having the lowest ES(j) value]

P2 = Primary slack of the chain
[high priority is assigned to the chain having the lowest primary slack]

P3 = Total duration of the chain
[high priority is assigned to the chain of the longest duration]

P4 = activity chain number
[high priority is assigned to the chain having the lowest chain number]

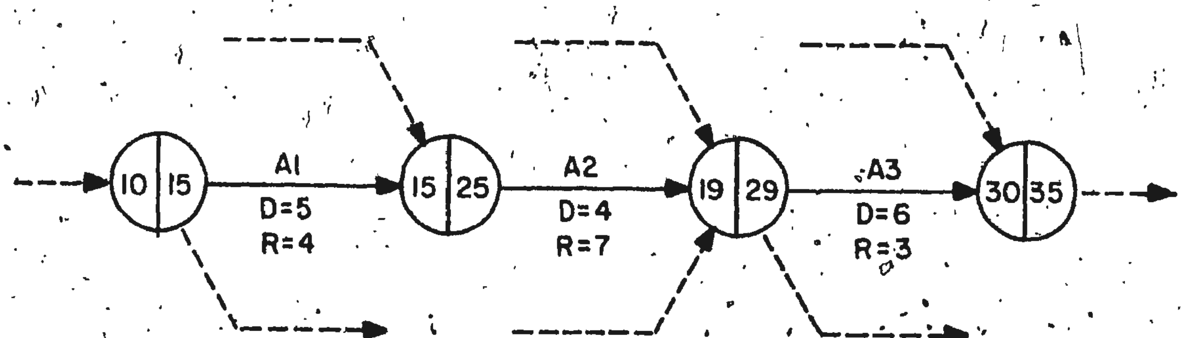
From these priority rules a priority value is defined as:

$$PR = P1 * 10^9 + P2 * 10^6 + (999-P3) * 10^3 + P4. \quad (2.38)$$

The exponents of the multipliers for different priorities are arbitrarily selected to obtain greater numerical value for higher priority and a lower value for lower priority. There is no other significance of the exponent which can be changed if this requirement is met.

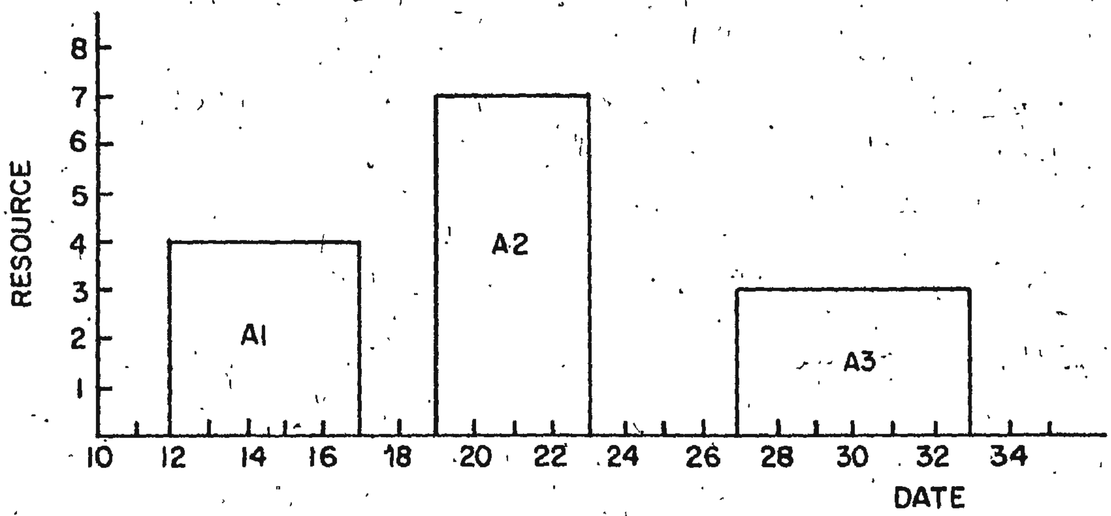
2.3.1 Activity Chain

Figure II-6 is an example of an activity chain for which two types of scheduling are considered within the limits of the earliest and the latest starts. The first alternative is scheduled with time lags between the activities (Figure II-7(a)) whereas the second alternative does not have such time lags (Figure II-7(b)).



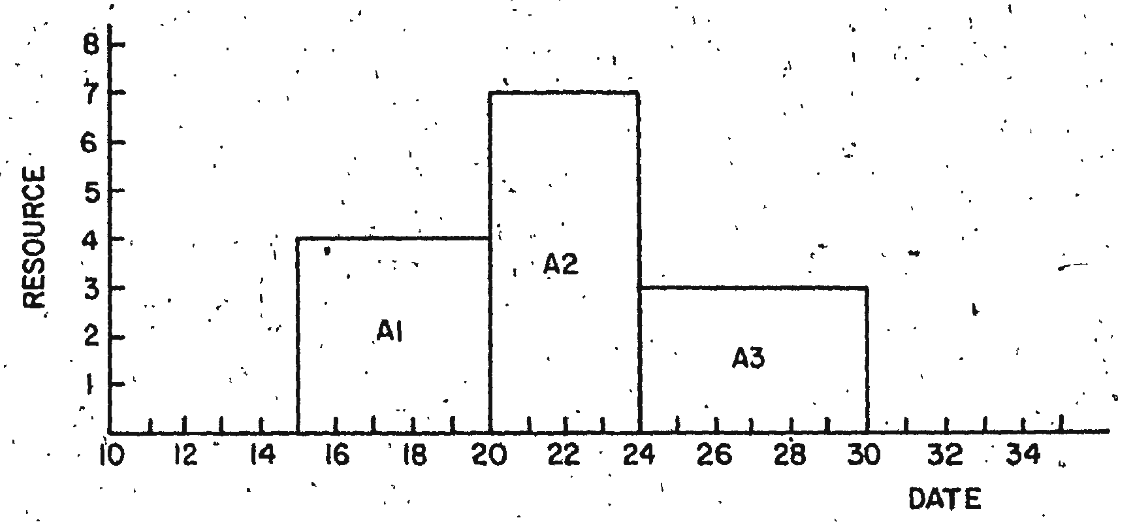
Activity Chain

Figure II-6



Schedule with Time-lags in Activity Chain

Figure II-7 (a)



Schedule without Time-lags in Activity Chain

Figure II-7 (b)

The sum of squares of resource change in Figure II-7(a) equals 148 which is obtained from $4^2 + 4^2 + 7^2 + 7^2 + 3^2 + 3^2$. The sum of squares of resource change in Figure II-7(b) equals 50 as obtained from $4^2 + 3^2 + 4^2 + 3^2$. By varying the time-lags between the activities and comparing them with the sum of squares of resource change in Figure II-7(b) it can be demonstrated that sequential scheduling results in the minimum sum of squares of resource change. This is the reason why an activity chain is defined as a set of sequential non-critical activities, in STEP 1.

2.3.2 Numerical Example

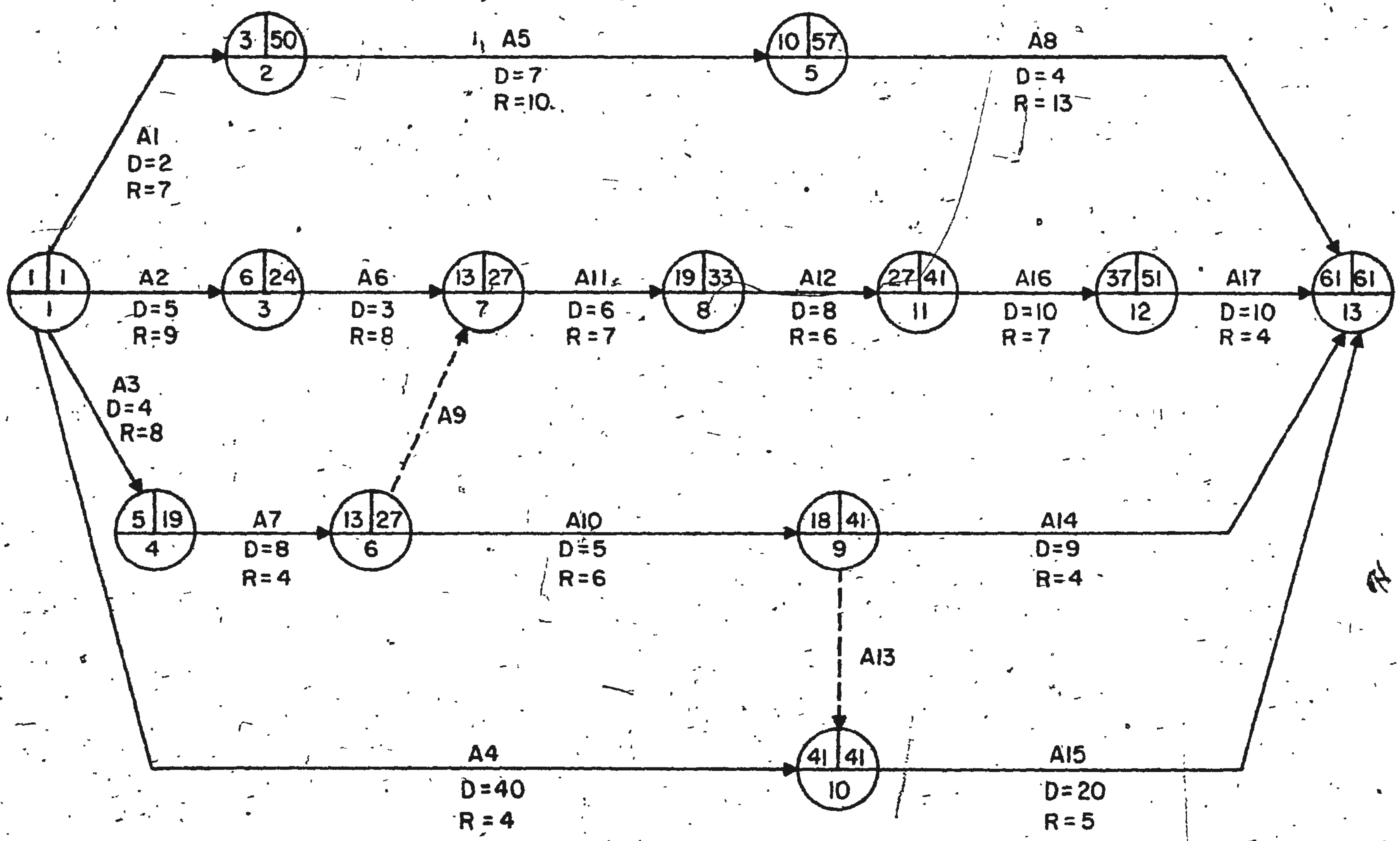
Figure II-8 shows an example of a small project. There are four activity chains in the network. These are listed in Table II-7.

chain number	activities
1	A1, A5, A8
2	A2, A6,
3	A3, A7, A9, A11, A12, A16, A17
4	A10, A14

Activity Chains

Table II-7

Figure II-9 (I) shows resource profile for the critical activities A4, A15. Figure II-9 (II) shows alternative resource profiles for the first chain, the earliest and latest start alternatives are denoted by firm and dotted lines respectively. Similarly, the alternative profile for the second, third and fourth chains are plotted in Figure II-9(III), II-9(IV), and II-9 (V) respectively.



CPM Network

Figure II-8

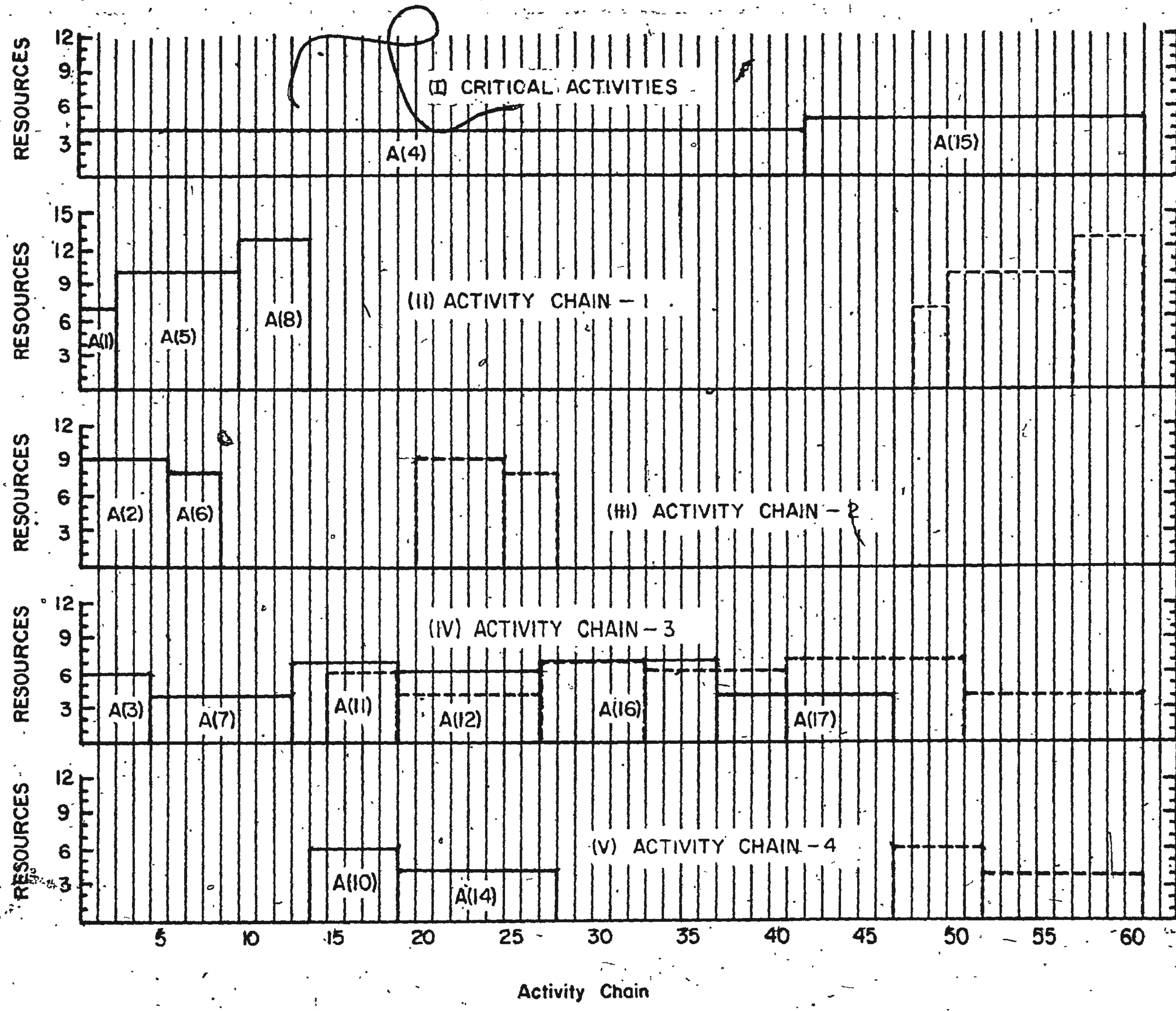

(I) CRITICAL ACTIVITIES
A(4)
A(15)
(II) ACTIVITY CHAIN - 1
A(1)
A(5)
A(8)
(III) ACTIVITY CHAIN - 2
A(2)
A(6)
(IV) ACTIVITY CHAIN - 3
A(3)
A(7)
A(11)
A(12)
A(16)
A(17)
(V) ACTIVITY CHAIN - 4
A(10)
A(14)
RESOURCES
5 10 15 20 25 30 35 40 45 50 55 60
Activity Chain

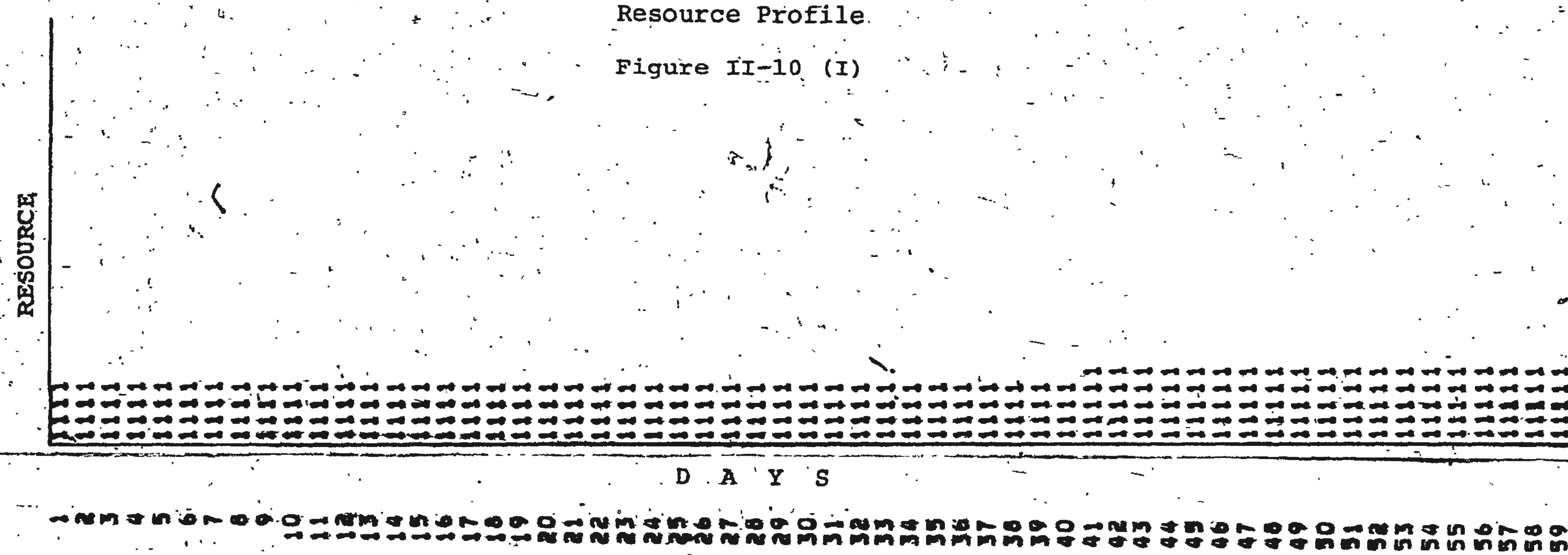
Figure II-9

Following the steps outlined in the beginning of this section, the scheduling is carried out. The profile obtained are plotted in Figure II-10 Figure II-10(I) shows the fixed resource allocation for the critical activities. Figure II-10(II) shows the resource allocation for the third activity chain superimposed over the critical activities resource profile. To this profile are subsequently added the second, first, and fourth activity chains which are shown in Figure II-10 (III), II-10 (IV) and II-10(V) respectively.

It will be observed in Figure II-10(V) that the resource profile has peaks and valleys. The start of the activities (not activity chains) is moved back and forth in an attempt to eliminated these peak. Figure II-11 is the final optimal solution. The optimal resource schedule is shown in Table II-8.

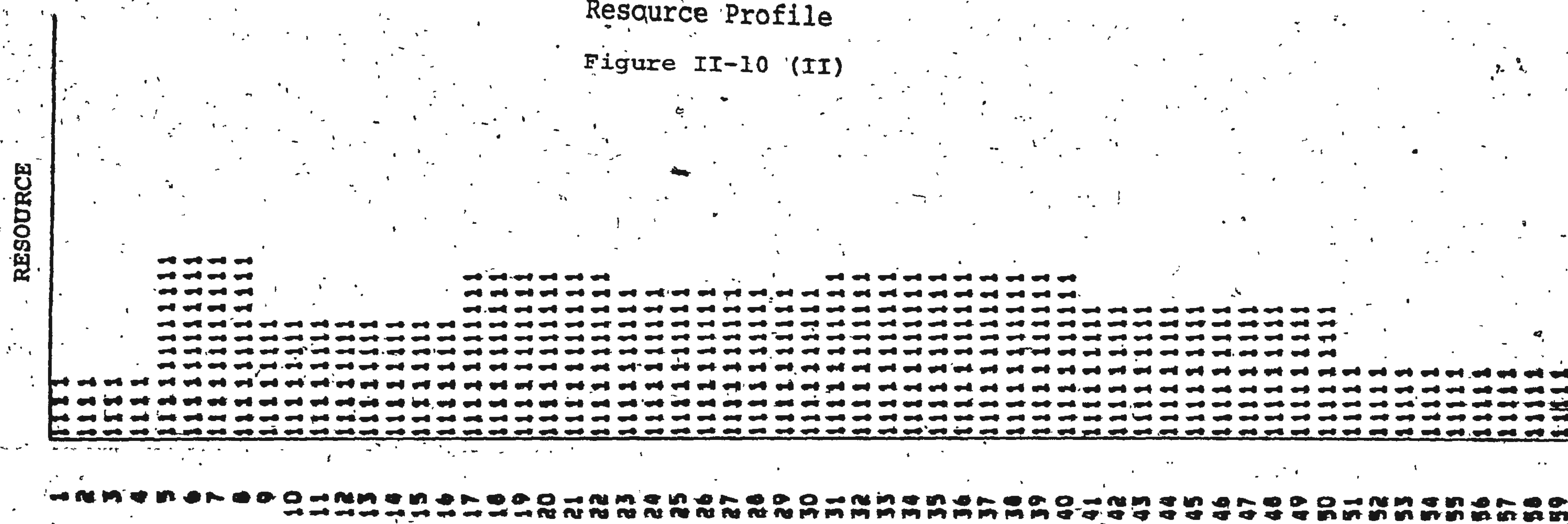
Resource Profile

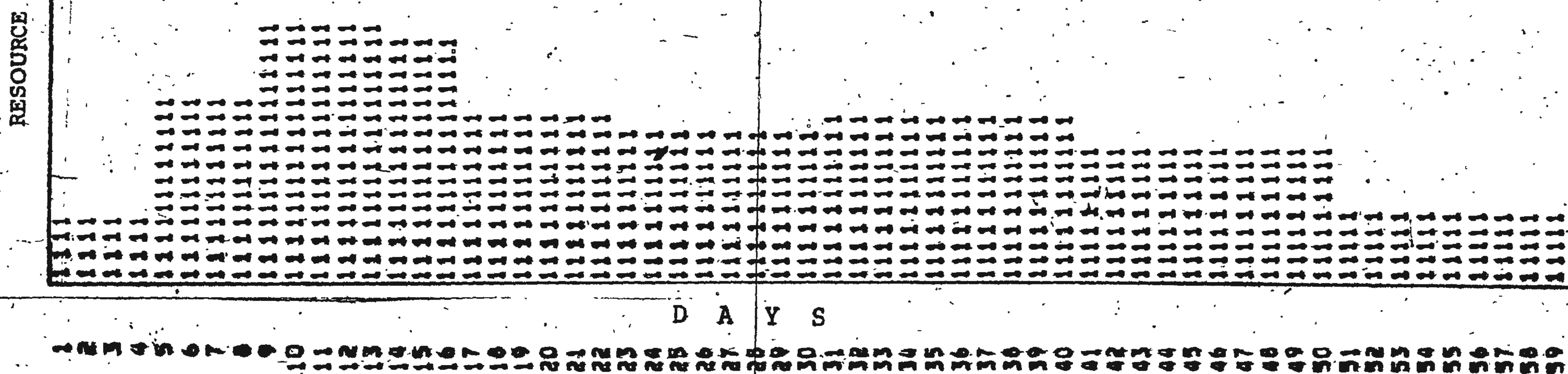
Figure II-10 (I)



Resource Profile

Figure II-10 (II)



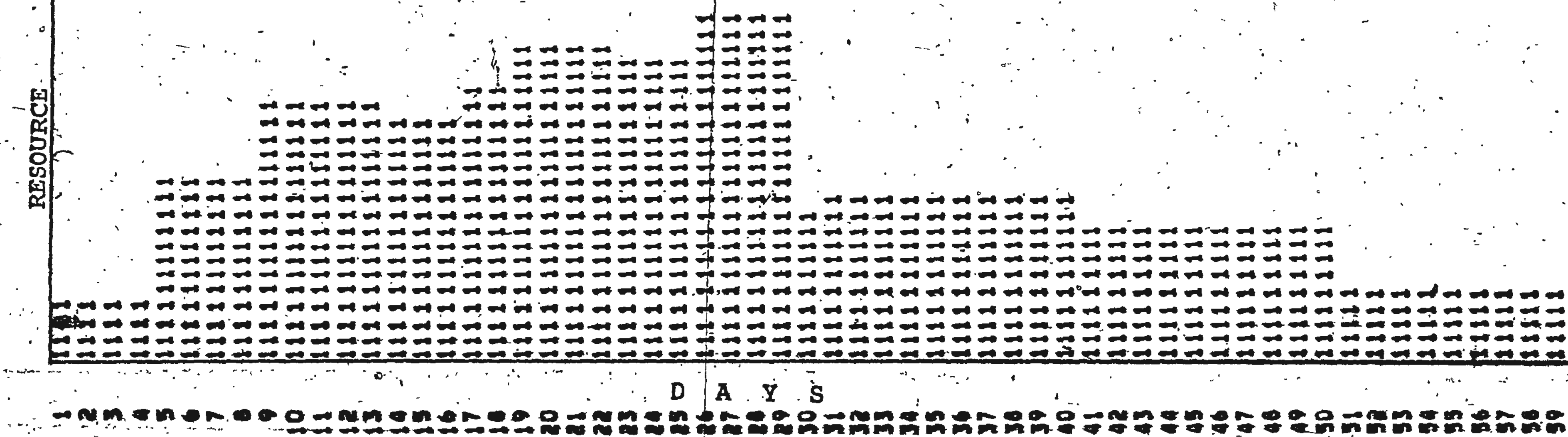


Resource Profile

Figure II-10 (III)

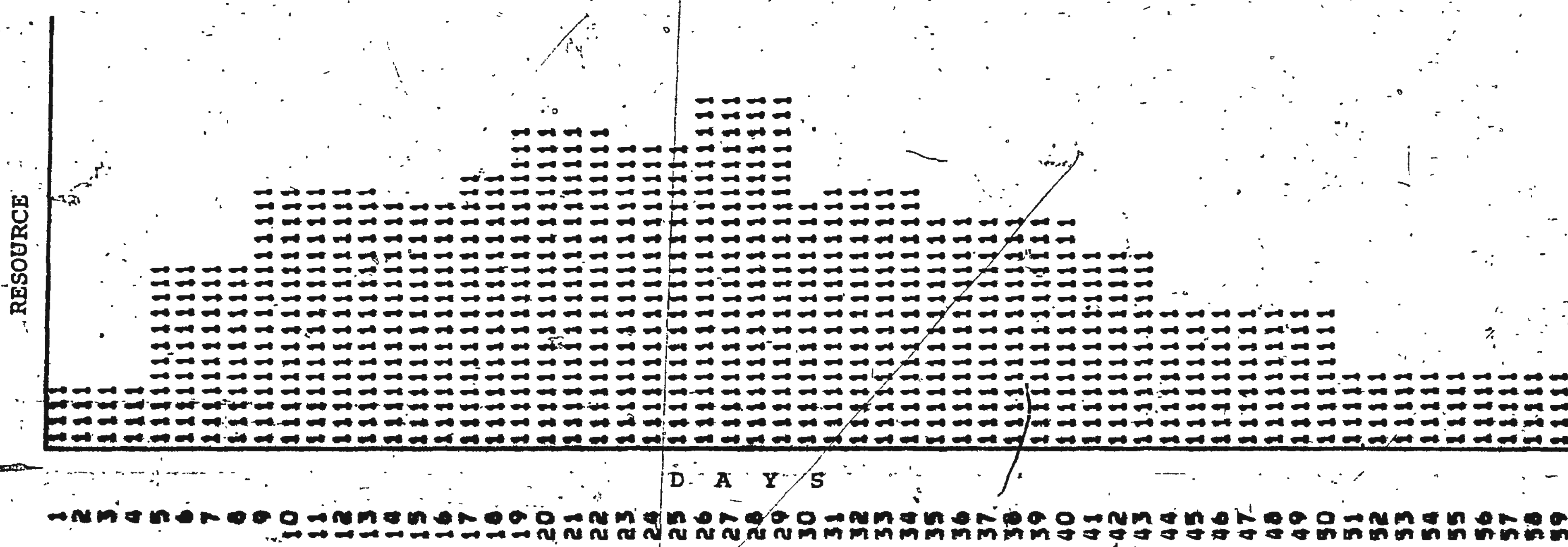
Resource Profile

Figure II-10 (IV)



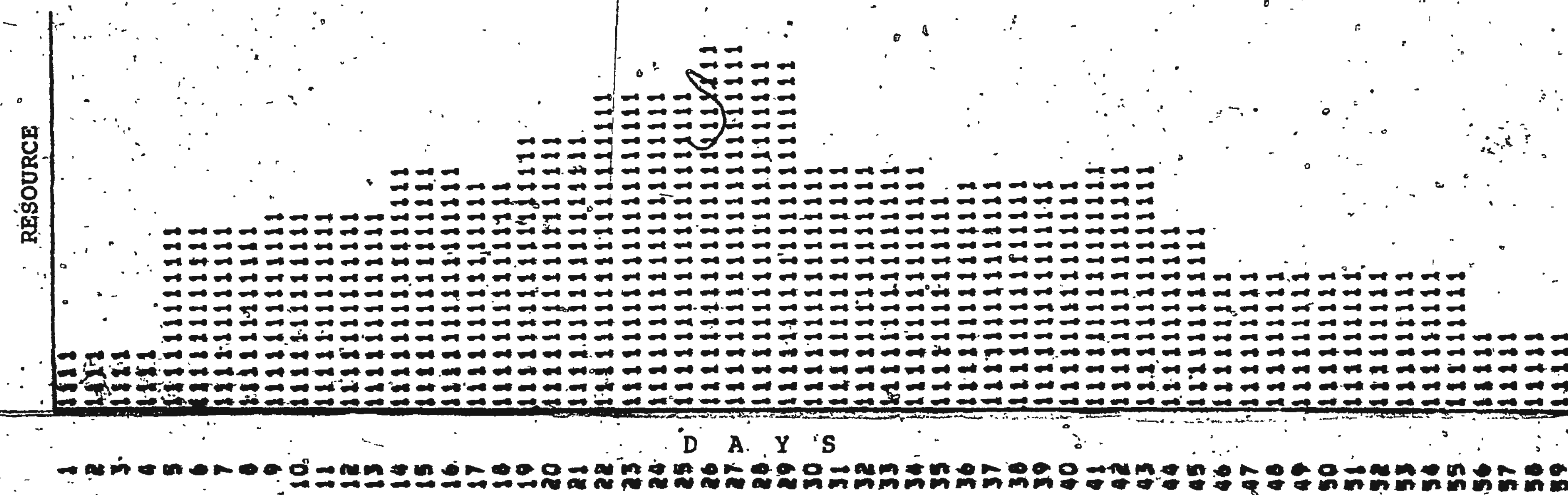
Resource Profile

Figure II-10 (V)



Final Optimal Resource Profile

Figure II-11



Activity Number	Activity Prec. Succ.	Duration	Resources	Scheduled Start
A1	1 - 2	2	7	17
A2	1 - 3	5	9	9
A3	1 - 4	4	8	5
A4	1 - 10	40	4	1
A5	2 - 5	7	10	19
A6	3 - 7	3	8	14
A7	4 - 6	8	4	14
A8	5 - 13	4	13	26
A9	6 - 7	0	0	22
A10	6 - 9	5	6	30
A11	7 - 8	6	7	22
A12	8 - 11	8	6	28
A13	9 - 10	0	0	41
A14	9 - 13	9	4	35
A15	10 - 13	20	5	41
A16	11 - 12	10	7	36
A17	12 - 13	10	4	41

Optimal Resource Schedule

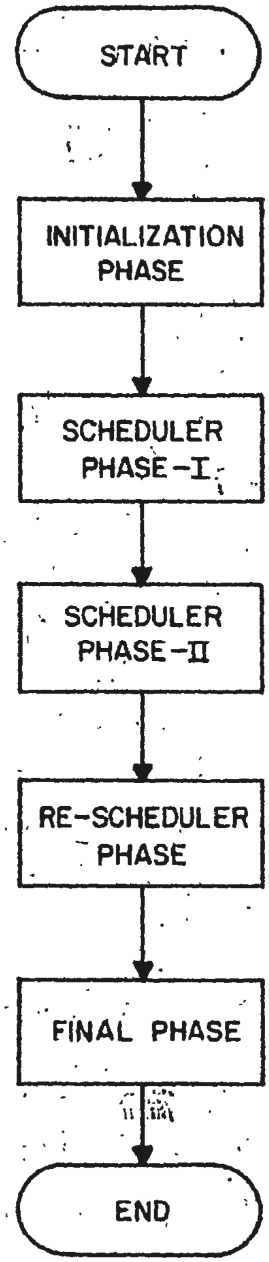
Table II-8

CHAPTER III

COMPUTER PROGRAMME OF THE HEURISTIC ALGORITHM

This chapter presents a computer programme called BACS (Build-up Activity Chain Schedule) programmed with FORTRAN IV IBM 370 for the heuristic algorithm developed in Section 3 Chapter II. The BACS programme is divided into five main phases as shown in Figure III-1. These phases are executed as a single job step and the function of each phase is as follows:

1. Initialization Phase reads CPM network input data and writes the information on the line-printer.
2. Scheduler Phase I makes activity chains and stores them in a file.
3. Scheduler Phase II schedules the activity chains to minimize the variance of resource change.
4. Re-scheduler Phase analyses the resource profile which has been built up by Scheduler Phase II and re-schedules activities to remove peaks, other than the maximum resource peak, if they are encountered.
5. Final Phase writes the optimal schedule.



System Flow Chart for BACS

Figure III-1

3.1 Initialization Phase

It inputs the necessary data for BACS from network information supplied by CPM analyses of the project. The flow diagram is shown in Figure III-2.

The following is a list of network data which BACS reads for each activity (input format is shown in Appendix B-1).

1. preceding node .... (I)
2. succeeding node .... (J)
3. duration .... (D)
4. resource requirement .... (R)
5. early start date and late finish date for preceding node .... (ES1,LF1)
6. early start date and late finish date for succeeding node .... (ES2,LF2)

The input data for each activity is stored in core and assigned an Activity Code as it is received. These Activity Codes are as follows:

1. Activity Code = 10 (non-critical activity which requires resources)

when	ES1	≠	LF1
or	ES2	≠	LF2
or	ES1 + D	≠	ES2
or	LS1 + D	≠	LS2
and	R	≠	0

2. Activity Code = 11 (non-critical activity which does not require resources)

when	ES1	≠	LF1
or	ES2	≠	LF2
or	ES1 + D	≠	ES2
or	LS1 + D	≠	LS2
and	R	=	0

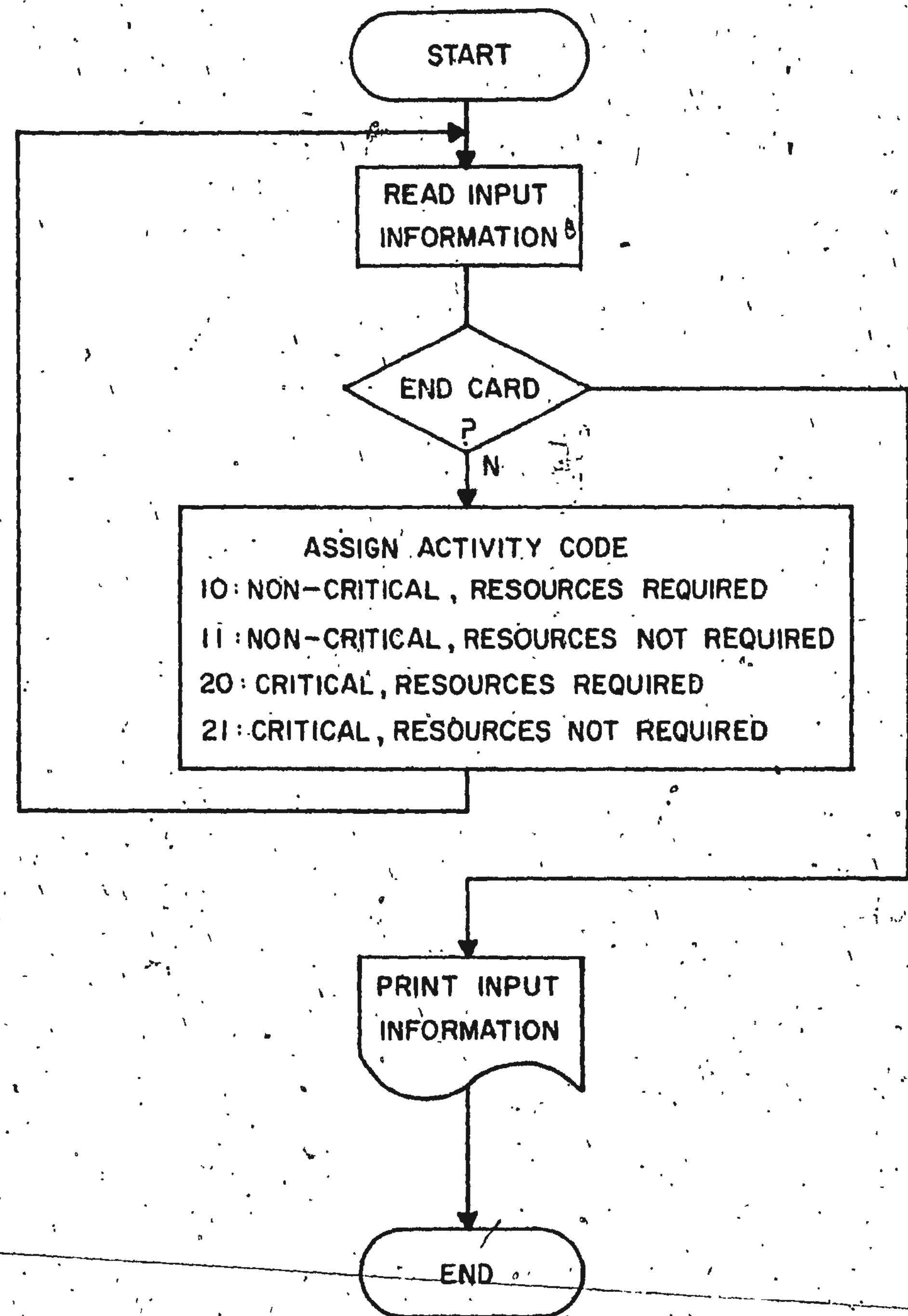
3. Activity Code = 20 (critical activity which requires resources)

when	ES1	=	LF1
and	ES2	=	LF2
and	ES1 + D	=	ES2
and	R	≠	0

4. Activity Code = 21 (critical activity which does not require resources)

when	ES1	=	LF1
and	ES2	=	LF2
and	ES1 + D	=	ES2
and	R	=	0

After reading all input data BACS outputs the input information on the line-printer.



Initialization Phase Flow Chart

Figure III-2

3.2 Scheduler Phase I

This phase analyses network information and forms activity chains in the project as showing in the flow chart in Figure III-6 The definition of an activity chain is : <u>the set of sequential non-critical activities whose primary slacks are equal.</u> The method to identify activity chains is as follows:

STEP 1: BACS assigns Job Code [JC(j), which means Job Code of activity j] as the first step towards the identification of activity chain.

JC(j) = 1 when ES1 + D = ES2 and LS1 + D ≠ LS2,

i.e. no preceding activity has the same primary slack.

JC(j) = 2 when ES1 + D = ES2 and LS1 + D = LS2,

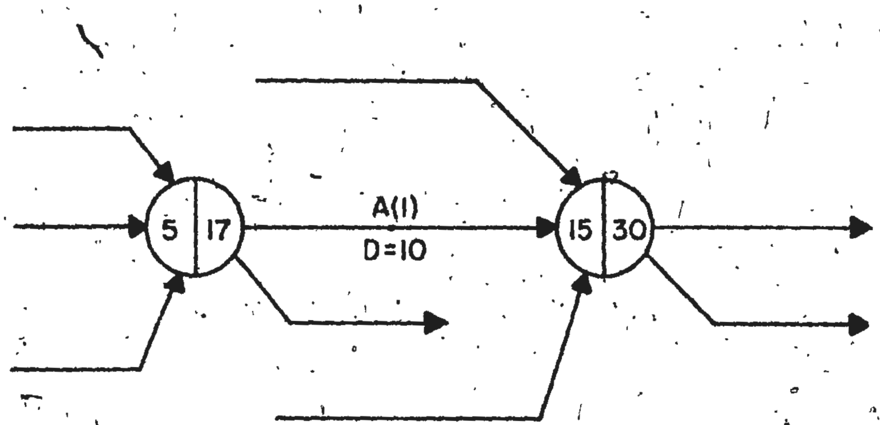
i.e. at least one preceding activity has the same primary slack and at least one following activity has the same primary slack.

$JC(j) = 3$ when $ES1 + D \neq ES2$

and $LS1 + D = LS2$,

i.e. no following activity has the same primary slack.

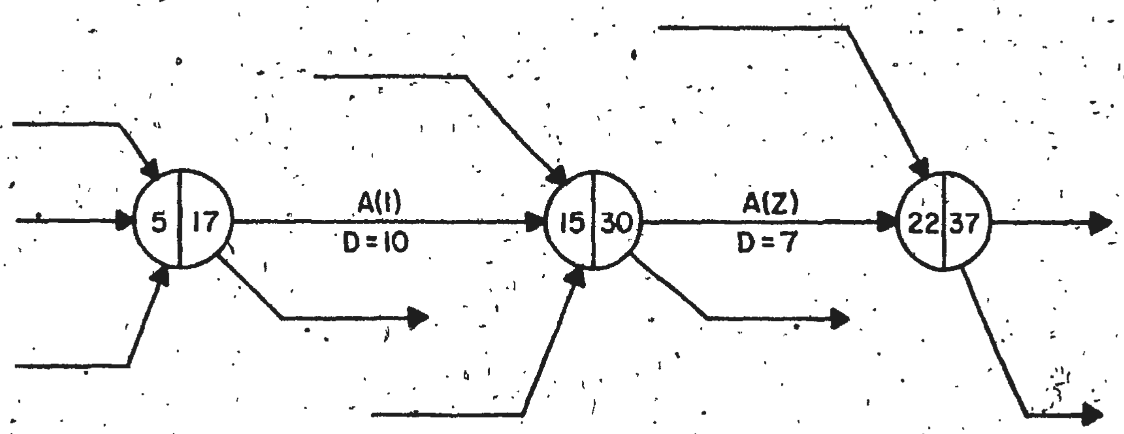
STEP 2: Select a non-critical activity, say A(1) whose JC(1)=1. This A(1) is a first activity of the chain. Figure III-3 is an example where JC(1) = 1.



The First Activity of the Activity Chain

Figure III-3

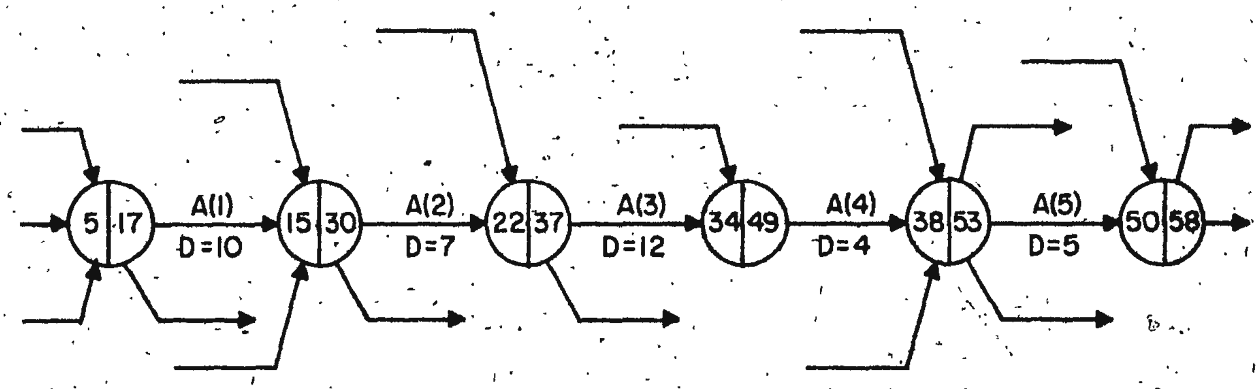
STEP 3 Find the following activity A(2) of A(1) whose JC(2)=2. This A(2) is a intermediate activity of the chain. Figure III-4 shows an example where JC(2) = 2.



Intermediate of the Activity Chain

Figure III-4

STEP 4: Repeat STEP 3 to find A(3), A(4),---- A(j) where each activity's JC = 2 and activity A(3) follows activity A(2), activity A(4) follows activity A(3), ---- and activity A(j) follows activity A(j-1). This step is repeated until no following activity is found,* or until A(j), the last activity of the chain whose JC = 3 is met. Figure III-5 shows an example of activity chain where A(1), A(2), A(3), A(4), and A(5) form a chain.



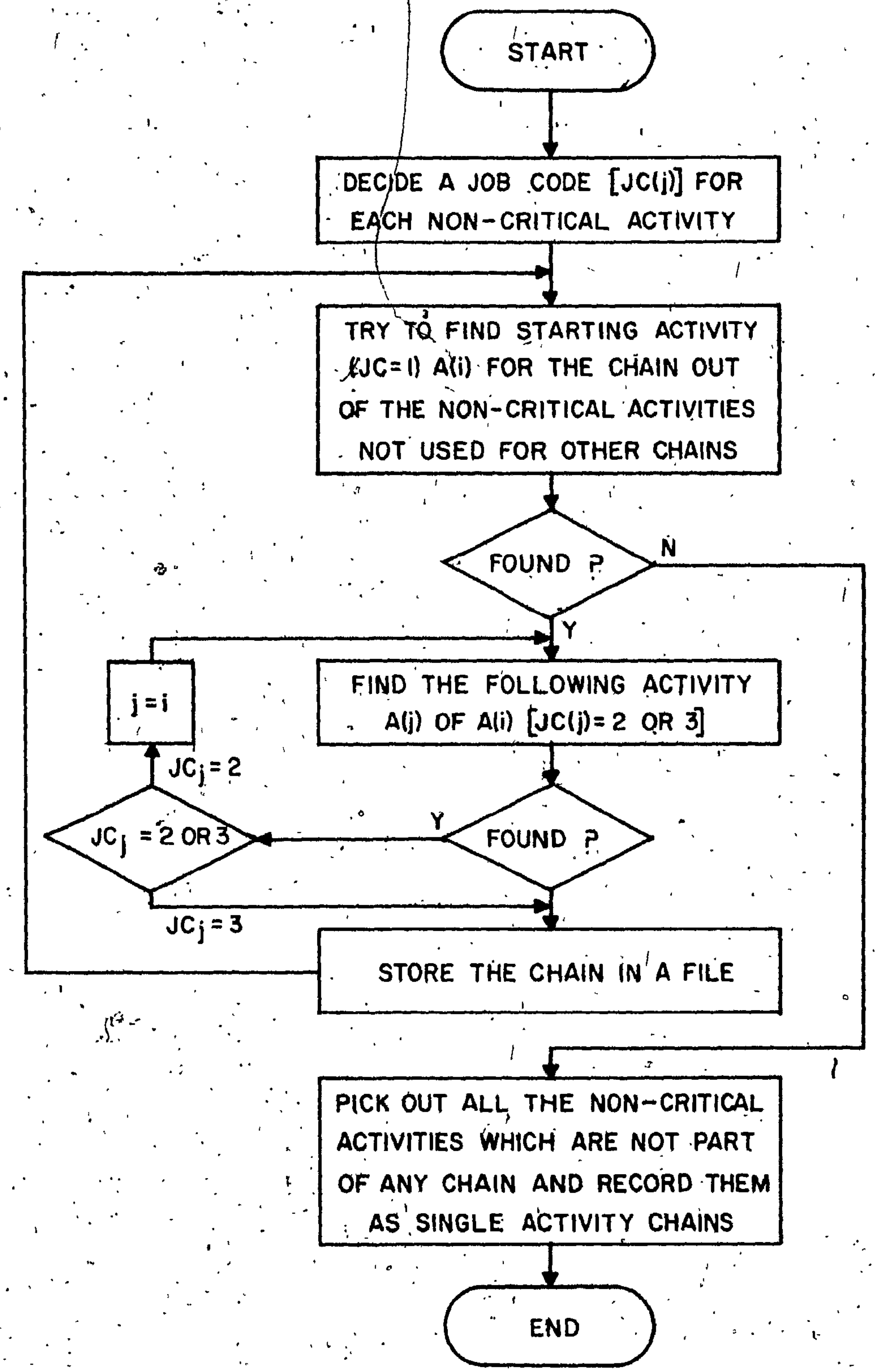
An Activity Chain

Figure III-5

* This happens when another activity chain has already used the activity A(j+1) whose JC=2 and which is a following activity of A(j).

STEP 5: Repeat again from Step 1 to make other activity chains until all non-critical activities are checked. The activity chains are stored in a file.

STEP 6: Pick up all non-critical activities which have not been involved in any activity chain. Each of these activities is considered as an activity chain in BACS although just one activity is involved in each case.



Scheduler Phase I – Flow Chart

Figure III – 6

3.3 Scheduler Phase-II

This phase schedules activity chains step by step to gradually build up resources to a peak and gradually lay them off (i.e. a parabolic resource profile). The procedure is flow charted in Figure III-7 and is described as follows:

STEP 1: Critical activities are scheduled and the resource change is recorded in a Resource Change File.

STEP 2: Priority rules are applied for each activity chain. The activity chain (C1) having the minimum priority value (Eqn.2.38) is scheduled first.

STEP 3: The start date of activity chain C1 is varied from the earliest start to its latest start, checking for each date, if the start date is feasible,* and computing the variance. Activity chain C1 is then scheduled on the starting date which provides minimum variance and the Resource Change File is incremented by the resource change of activity chain C1.

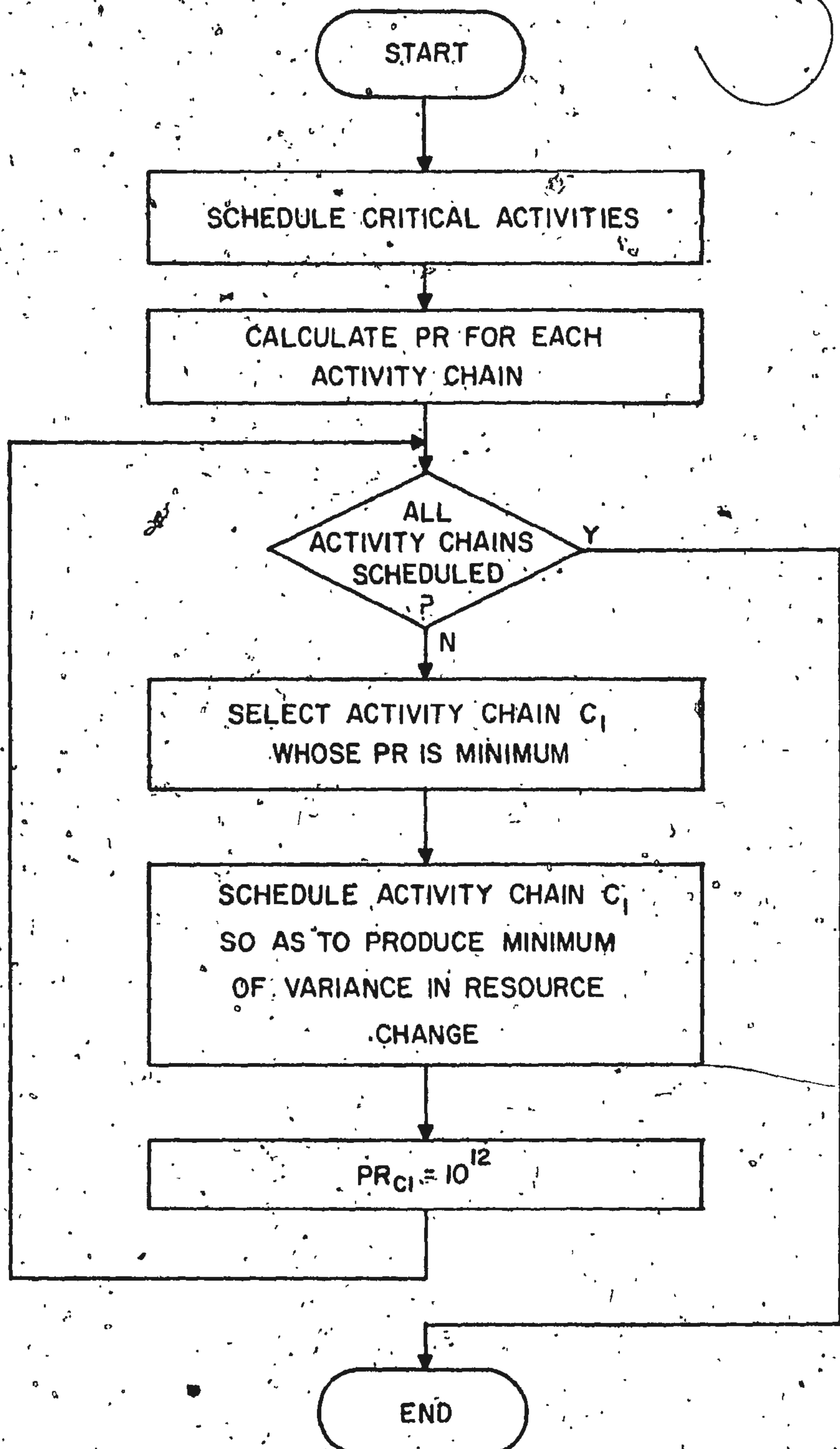
---

* The feasibility check implies that all the preceding activities are already scheduled.

STEP 4: PR value of the scheduled activity chain is incremented to 10^{12} to prevent a repetitive consideration.

STEP 5: Step 2 to 5 are repeated until all activity chains are scheduled.

STEP 6: Print a resource profile.



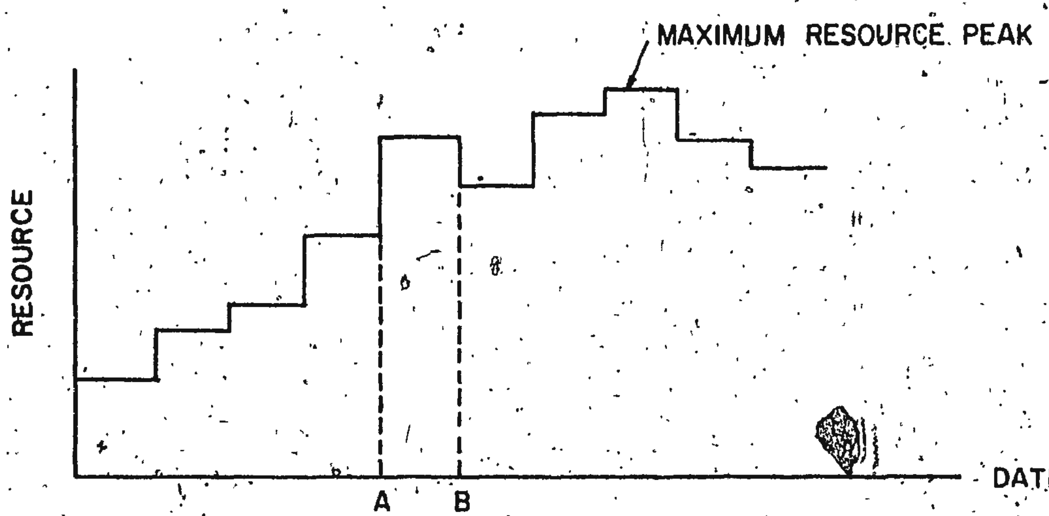
Scheduler Phase II – Flow Chart

Figure III-7

3.4 Re-scheduler Phase

This phase looks over the resource profile which has already been built up by Scheduler Phase II and if any peak other than its maximum resource peak, is encountered, BACS tries to remove it by re-scheduling activities not activity chains. The procedure shown in Figure III-14 can be divided into two parts.

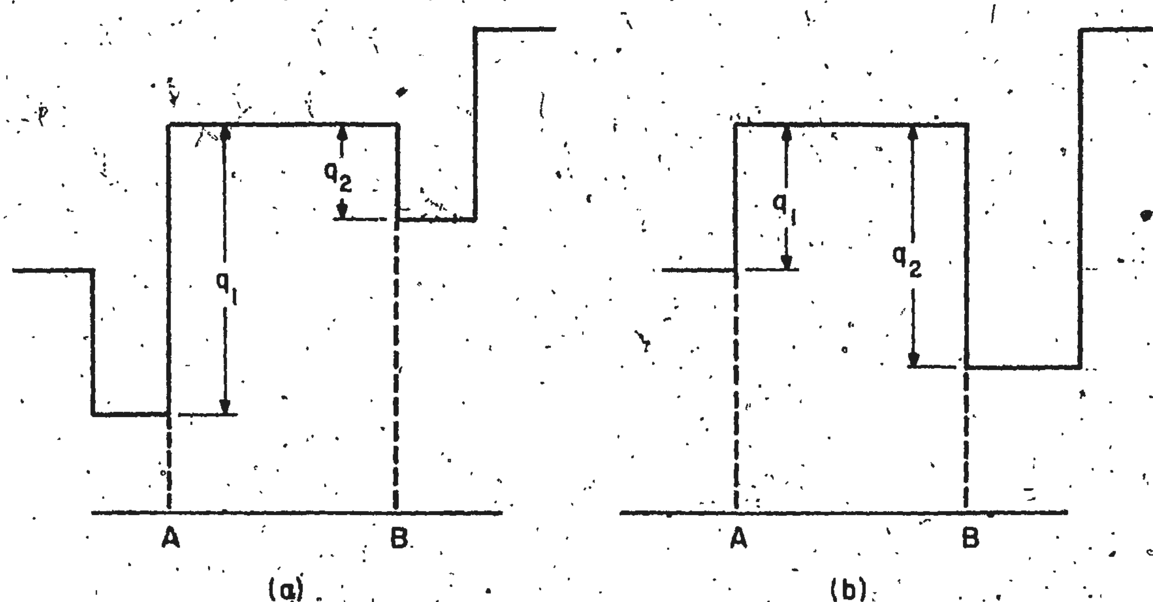
[PART-I] A peak is encountered prior to the maximum resource peak in the resource profile.



Resource Profile

Figure III-8.

In Figure III-8 the resource profile has a peak between dates A and B. Two cases are considered about this peak. First, in Figure III-9 (a) the length of plus resource change, q1 is larger than that of minus resource change, q2 and second, in Figure III-9 (b) the length of minus resource change, q2 is larger than that of plus resource change, q1.



Peak in Resource Profile

Figure III – 9

STEP 1 BACS looks for a peak in the resource profile.

STEP 2 If a peak is found, BACS tries to find all activity A(j)'s, which start on date A and whose resource requirement R is

$$q2 \leq R \leq q1 \quad \text{when } q1 \geq q2$$

or

$$q1 \leq R \leq q2 \quad \text{when } q1 < q2$$

if no activity was found, BACS continues with STEP 1.

STEP 3 If there are one or more activities satisfying the above restriction, BACS reschedules each activity A(j) forward a length P defines as,

$$P = \overline{A - B}^{*} \quad \text{when } LS(j) - A \geq \overline{A - B},$$

or

$$P = LS(j) - A \quad \text{when } LS(j) - A < \overline{A - B}.$$

Then all the activities following activity A(j) are examined to select the activities which are affected by delaying the start of activity A(j). The start times of such activities must change, this start is moved forward equal to or less than the forward push of the activity A(j) causing the unwanted peak. The extent of movement of any activity is determined by the technological sequence (i.e. it is scheduled contiguously with the completion of its immediately preceding activity). As an example

---

* $\overline{A - B}$ = the absolute difference of time between A and B.

suppose Activity A(1) in Figure III-10 is rescheduled 2 days later, then the following activities A(2), A(3) and A(4) are rescheduled 2 days later, A(5) 1 day later, and A(6) is unaffected. This is done to minimize the effect of removing the peak.

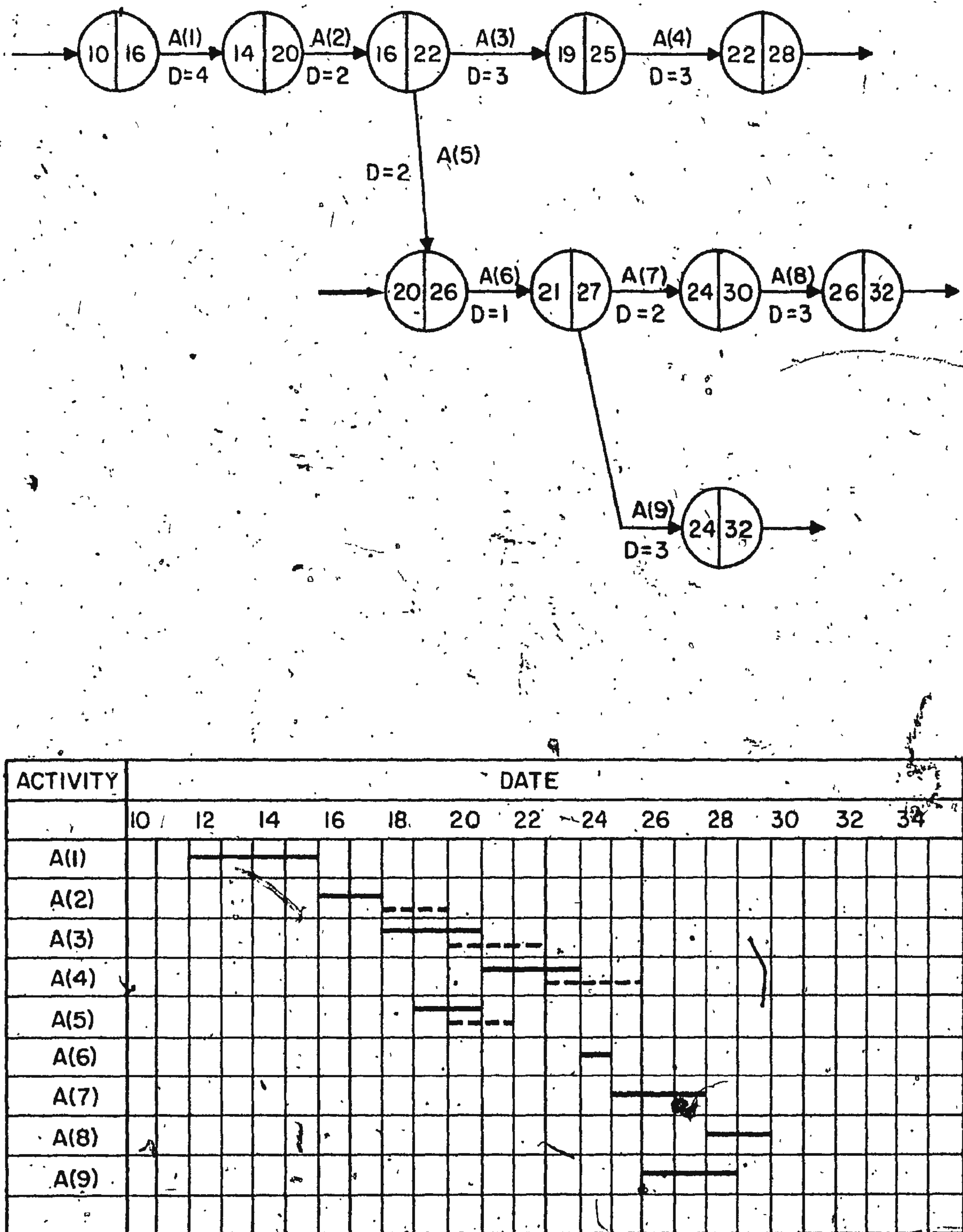
STEP 4 BACS calculate the variance of resource change for the new schedule or schedules if more than one activity was found which, satisfied the criteria of STEP 2 and if the lowest variance of the new schedules is less than the previous variance*, the new schedule is accepted.

STEP 5 Repeat from STEP 1 until the maximum resource peak (MRP) of the project is reached.

STEP 6 Go to PART-II of this phase.

---

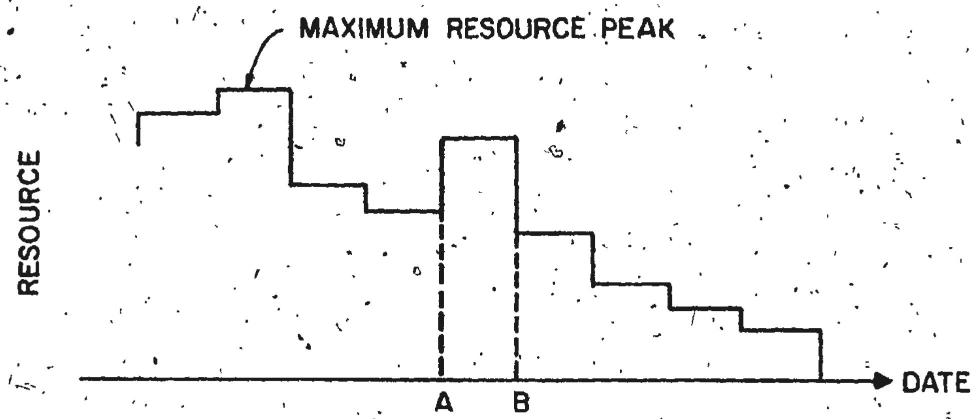
* Suppose the new variance is same as the old one, BACS checks the level of highest resource requirement of the two schedules and selects the schedule whose peak resource requirement is lower.



Activity Schedule Chart

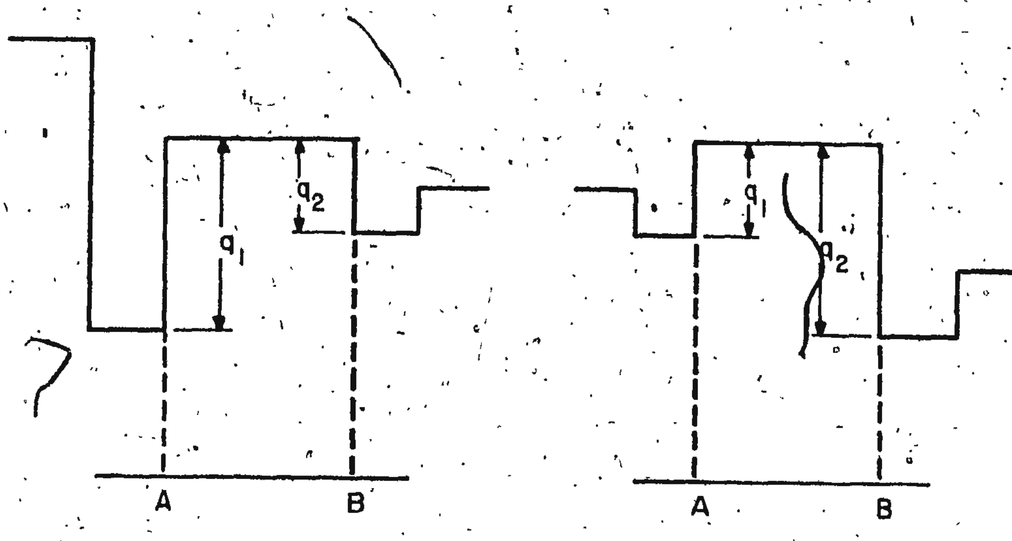
Figure III-10

[PARTS-II] A peak is encountered after the maximum resource peak in the resource profile.



Peak in Resource Profile
Figure III-11

In Figure III-11 the resource profile has a peak between dates A and B. Again two cases are considered for the peak. First, in Figure III-12 (a) the length of plus resource change, q1 is larger than that of minus resource chage, q2 and second, in Figure III-12 (b) the length of minus resource change, q2 is larger than that of plus resource change, q1.



Peak in Resource Pool

Figure III-12

STEP 1 BACS continues looking for a peak in the second half of the resource profile.

STEP 2 If a peak is found, BACS tries to find all activities A(j)'s which ends on date B and whose resource requirement R is

$$q2 \leq R \leq q1 \qquad \text{when } q1 \geq q2$$

or $$q1 \leq R \leq q2 \qquad \text{when } q1 < q2.$$

If no activity is found, BACS continues with STEP 1.

STEP 3 If there is one or more activities satisfying the above restriction, BACS reschedules each activity backwards a length P defined as,

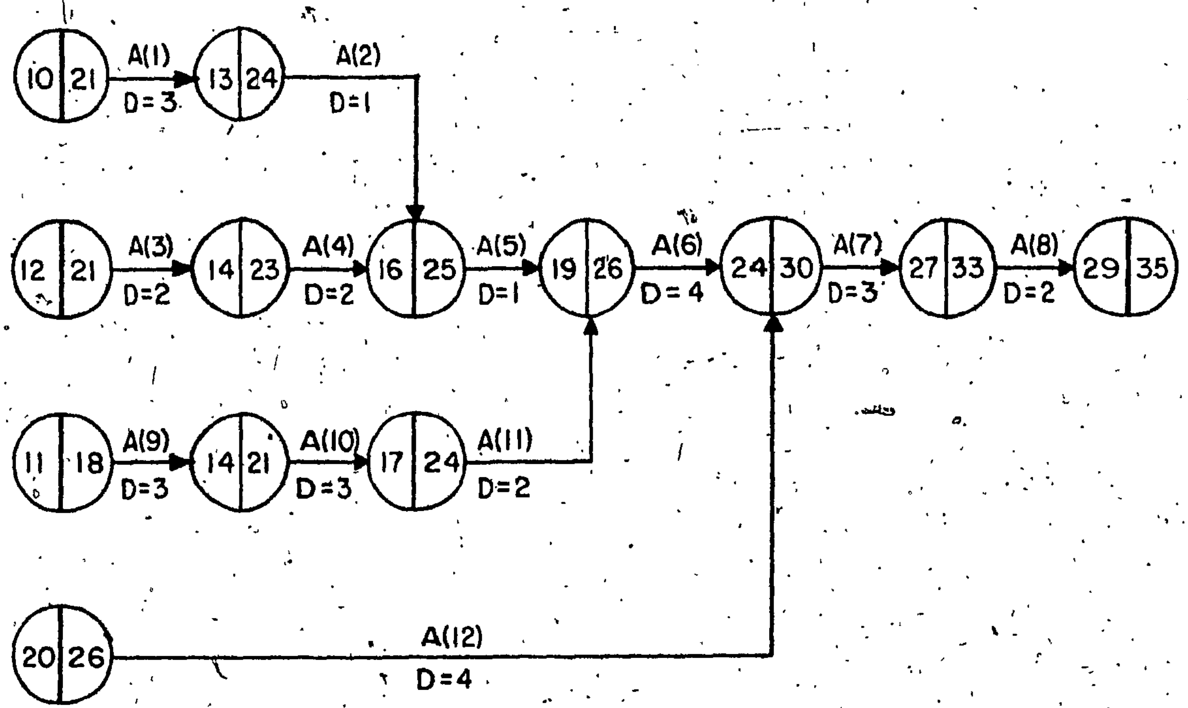
$$P = \overline{A - B} \qquad \text{when } EF(j)-B \geq \overline{A - B}$$

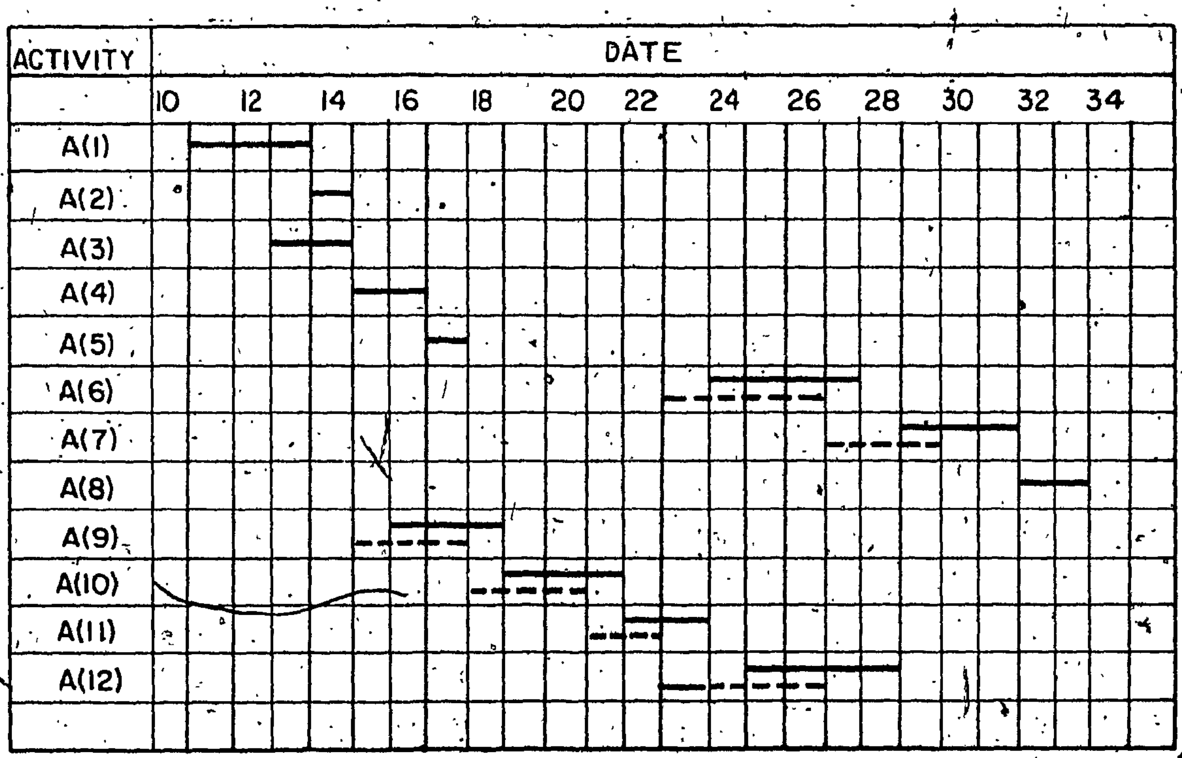
$$P = EF(j) - B \qquad \text{when } EF(j)-B < \overline{A - B}.$$

Then all the activities preceding activity A(j) are examined to select the activities which are affected by rescheduling the start of activity A(j) earlier. The start times of such activities must be changed, the start is moved backwards equal to or less than the backward push of activity A(j) causing the unwanted peak. The extent of movement of any activity is determined by the technological sequence. As an example suppose A(7) in Figure III-13 is rescheduled 2 days earlier, then the preceeding activity A(12) is rescheduled 2 days earlier, and A(6), A(9), A(10) and A(11) 1 day earlier.

STEP 4 BACS calculates variance of resource change for the new schedule or schedules if more than one activity was found which satisfied the criteria of STEP 2 and if the lowest variance of the new schedule is less than the previous variance, the new schedule is accepted. BACS then returns to STEP I of PART I. If the variance is not reduced, it goes to next step.

STEP 5 BACS repeats from STEP 1 of PART-II until project duration is reached.



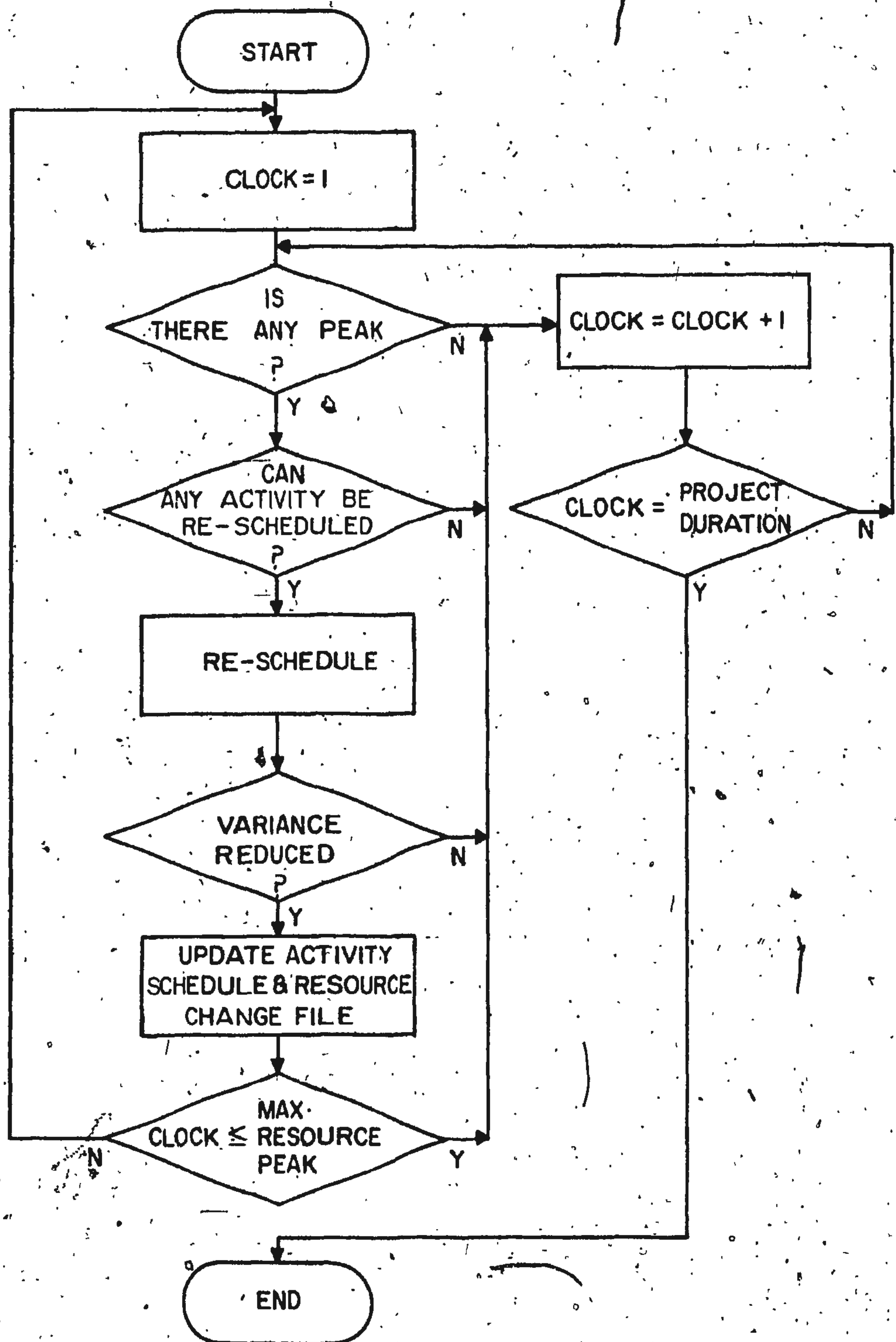


Activity Schedule Chart

Figure III-13

The criteria for achieving an optimal solution has been the minimization of variance of resource change. First, the resource profile for the critical activities was found. This part of the resource profile is fixed. Then, activity chains were built-up and priority rules applied so that chains were considered in order of importance. The resource profile of these chains was then added to the previous resource profile on starting positions which gave the minimum variance. Next, a look back over the resource profile was made and if any unwanted peaks were encountered, activities starting on the date of the peak were moved back and forth in an attempt to minimize the variance by removing the peak. All the affected activities were considered even if they were not part of the chain. Other activities were not moved as their shifting back and forth would not minimize variance. Because of the constraints of fixed duration and fixed resources on the activities and the technological sequence imposed by the CPM network, it will not always be possible to remove these peaks and thereby further reduce the variance.

The heuristic procedure has attempted to eliminate only those alternatives which will not help in minimizing the variance, so that the final optimal resource distribution achieved is as close as possible to the optimum resource distribution, taking into account the constraint of computational time.

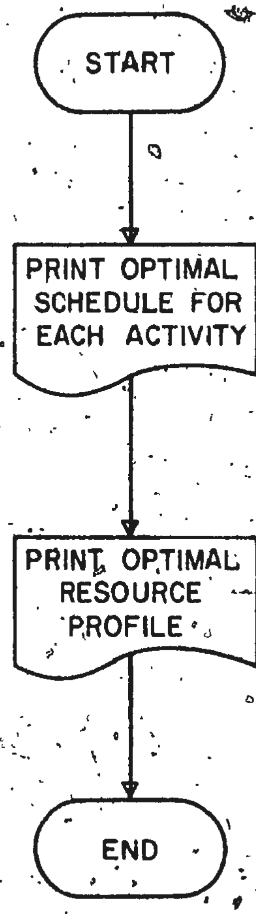


Re-Scheduler Phase Flow Chart

Figure III-14

3.5 Final Phase

BACS writes the optimal schedule on line printer following the optimal resource profile. Figure III-15 shows the flow chart of this phase.



Final Phase Flow Chart

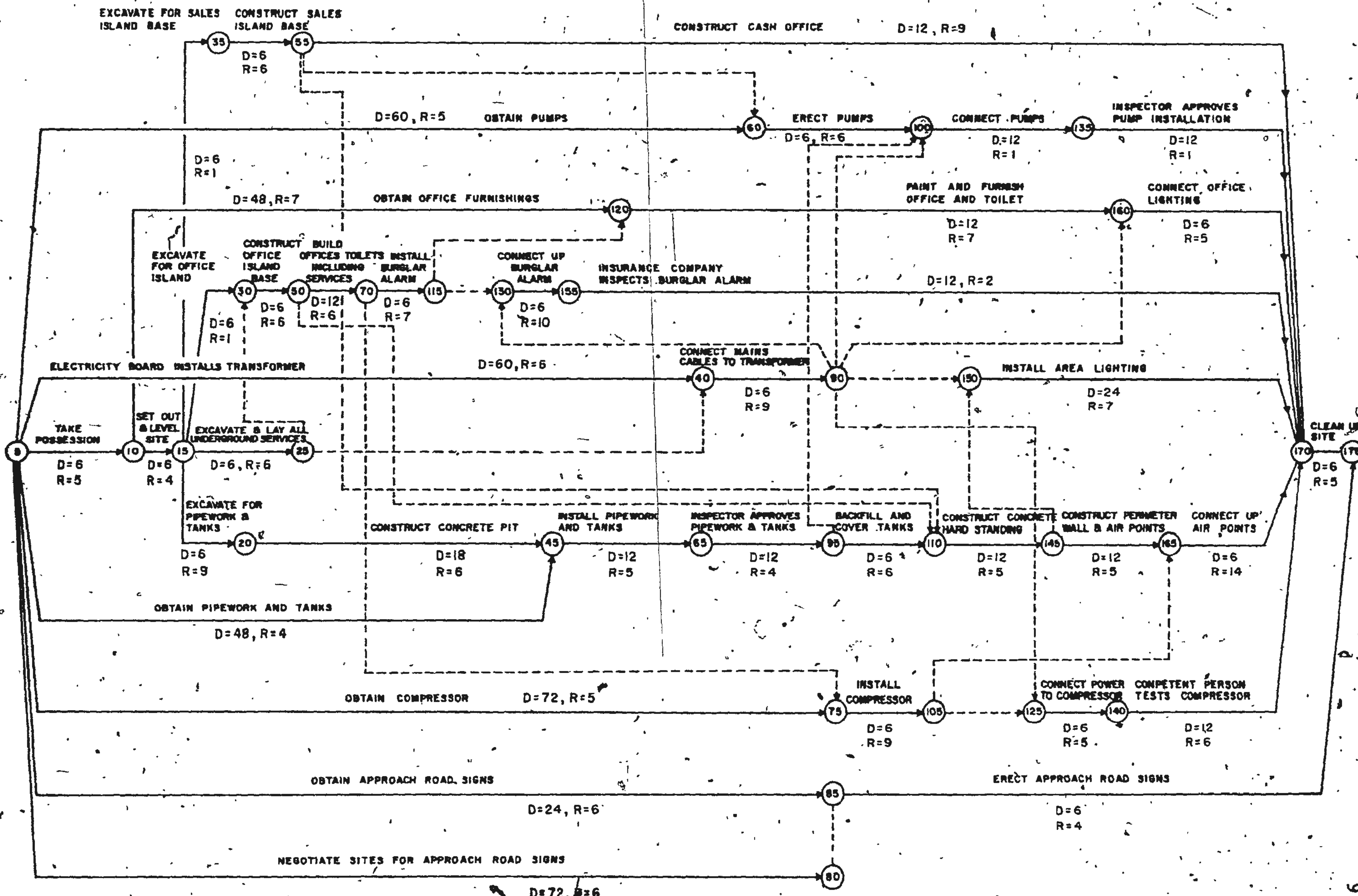
Figure III-15

CHAPTER IV

NUMERICAL EXAMPLE USING A COMPUTER PROGRAMME

The computer programme BACS as described in Chapter III is listed in Appendix B(1). This example demonstrates the use and output analysis of BACS and is divided into the phases outlined in Chapter III.

Figure IV-1 is the CPM network for a gas station project (57 activities). The Initialization Phase accepts data from the CPM network diagram for each activity in the format shown in Appendix B(2). Network data required for each activity is i) preceding node number, I(j), ii) succeeding node number, J(j), iii) duration, D(j), iv) resource requirement, R(j), v) early start date and late finish date for preceding node, ES1(j), LF1(j), vi) early start date and late finish date for succeeding node, ES2(j), LF2(j). For example for the first activity take possession I(1)=5, J(1)=10, D(1)=6, R(1)=6, ES1(1)=1, LF1(1)=1, ES2(1)=7, and LF2(1)=19. Since ES2(1) $\neq$ LF2(1) and R(1) $\neq$ 0, the computer assignes an Activity Code of 10 (non-critical activity, required resources). Figure IV-2 lists the CPM network data and computer assigned Activity Code, printed at the completion of the Initialization Phase.



Gas Station
Figure IV-1

NETWORK INFORMATION

NO	I	J	D	R	ES1	LF1	ES2	LF2	CODE
1	5	10	6	5	1	1	7	19	10
2	5	40	60	6	1	1	61	85	10
3	5	45	48	4	1	1	49	49	20
4	5	60	60	5	1	1	61	85	10
5	5	75	72	5	1	1	73	91	10
6	5	80	72	6	1	1	73	115	10
7	5	85	24	6	1	1	73	115	10
8	10	15	6	4	7	19	13	25	10
9	10	120	48	7	7	19	55	97	10
10	15	20	6	9	13	25	19	31	10
11	15	25	6	6	13	25	19	73	10
12	15	30	6	1	13	25	19	73	10
13	15	35	6	1	13	25	19	73	10
14	20	45	18	6	19	31	49	49	10
15	25	30	0	0	19	73	19	73	11
16	25	40	0	0	19	73	61	85	11
17	30	50	6	6	19	73	25	79	10
18	35	55	6	6	19	73	25	79	10
19	40	90	6	9	61	85	67	91	10
20	45	65	12	5	49	49	61	61	20
21	50	70	12	6	25	79	37	91	10
22	50	110	0	0	25	79	79	79	11
23	55	60	0	0	25	79	61	85	11
24	55	110	0	0	25	79	79	79	11
25	55	170	12	9	25	79	115	115	10
26	60	100	6	5	61	85	73	91	10
27	65	95	12	4	61	61	73	73	20
28	70	75	0	0	37	91	73	91	11
29	70	115	6	9	37	91	43	97	10
30	75	105	6	9	73	91	79	97	10
31	80	85	0	0	73	115	73	115	11
32	85	175	6	4	73	115	121	121	10
33	90	100	0	0	67	91	73	91	11
34	90	125	0	0	67	91	79	97	11
35	90	130	0	0	67	91	67	97	11
36	90	150	0	0	67	91	91	91	11
37	90	160	0	0	67	91	67	109	11
38	95	100	0	0	73	73	73	91	11
39	95	110	6	6	73	73	79	79	20
40	100	135	12	1	73	91	85	103	10
41	105	125	0	0	79	97	79	97	11
42	105	165	0	0	79	97	103	109	11
43	110	145	12	5	79	79	91	91	20
44	115	120	0	0	43	97	55	97	11
45	115	130	0	0	43	97	67	97	11
46	120	160	12	7	55	97	67	109	10
47	125	140	6	5	79	97	85	103	10
48	130	155	6	10	67	97	73	103	10
49	135	170	12	1	85	103	115	115	10
50	140	170	12	6	85	103	115	115	10
51	145	150	0	0	91	91	91	91	21
52	145	165	12	5	91	91	103	109	10
53	150	170	24	7	91	91	115	115	20
54	155	170	12	2	73	103	115	115	10
55	160	170	6	5	67	109	115	115	10
56	165	170	6	14	103	109	115	115	10
57	170	175	6	5	115	115	121	121	20

Figure IV-2

Next, Scheduler Phase I assigns Job Codes, JC(j), to each activity. For example for the activity "take possession", since

$$ES1(1) + D(1) = ES2(1) \quad \text{and}$$

$$LS1(1) + D(1) \neq LS2(1),$$

a JC(1) = 1 is assigned (no preceding activity has the same primary slack).

Scheduler Phase I schedules critical activities first. Next, priority values are calculated for each activity chain from priority rules and Equation 3.38. Activity chains are now scheduled starting with the chain having the lowest priority value. Figure IV-5 is the resulting resource profile, once all activity chains have been scheduled.

ACTIVITY CHAINS

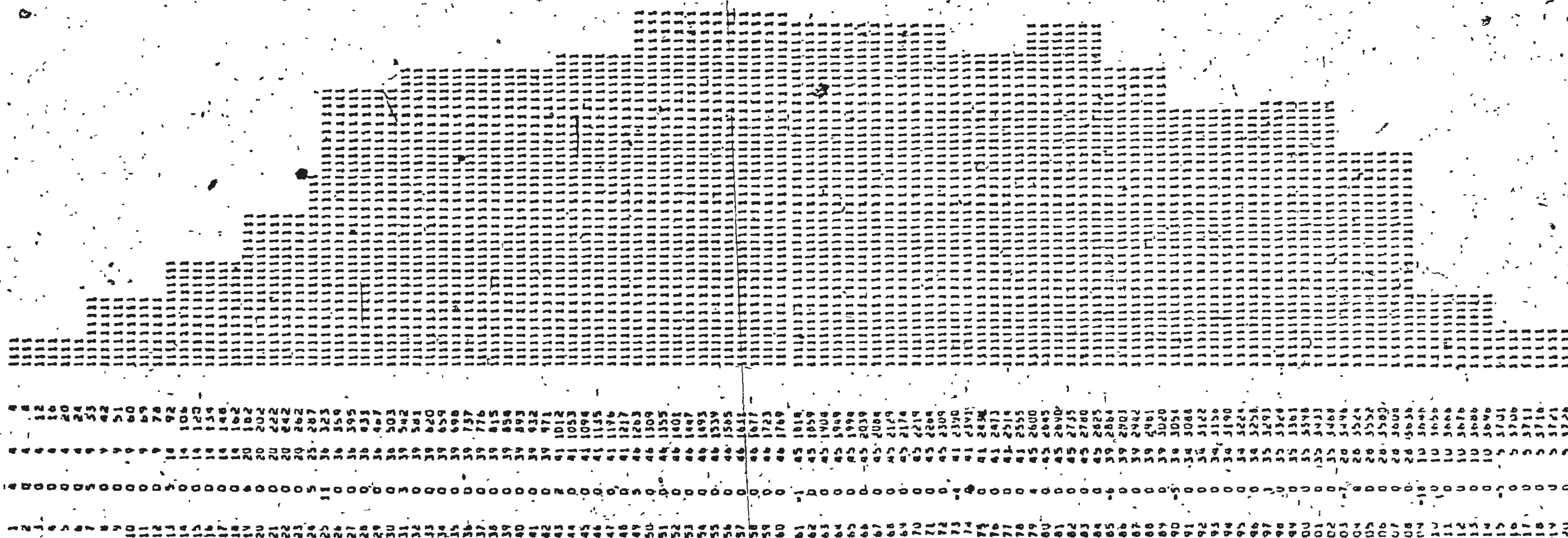
NO	ACTIVITY NUMBER											
1	1	8	10	14	0	0	0	0	0	0	0	0
2	2	19	33	0	0	0	0	0	0	0	0	0
3	4	26	0	0	0	0	0	0	0	0	0	0
4	5	30	41	47	50	0	0	0	0	0	0	0
5	6	31	32	0	0	0	0	0	0	0	0	0
6	9	46	55	0	0	0	0	0	0	0	0	0
7	11	15	17	21	28	0	0	0	0	0	0	0
8	12	0	0	0	0	0	0	0	0	0	0	0
9	13	18	24	0	0	0	0	0	0	0	0	0
10	35	48	54	0	0	0	0	0	0	0	0	0
11	37	0	0	0	0	0	0	0	0	0	0	0
12	38	40	49	0	0	0	0	0	0	0	0	0
13	52	56	0	0	0	0	0	0	0	0	0	0
14	29	0	0	0	0	0	0	0	0	0	0	0
15	7	0	0	0	0	0	0	0	0	0	0	0
16	16	0	0	0	0	0	0	0	0	0	0	0
17	22	0	0	0	0	0	0	0	0	0	0	0
18	23	0	0	0	0	0	0	0	0	0	0	0
19	25	0	0	0	0	0	0	0	0	0	0	0
20	34	0	0	0	0	0	0	0	0	0	0	0
21	36	0	0	0	0	0	0	0	0	0	0	0
22	42	0	0	0	0	0	0	0	0	0	0	0
23	44	0	0	0	0	0	0	0	0	0	0	0
24	45	0	0	0	0	0	0	0	0	0	0	0

Figure IV-3

1	....	1013963
2	....	1025933
3	....	1025933
4	....	1019903
5	....	1043921
6	....	7043933
7	....	13055975
8	....	13055993
9	....	13055987
10	....	67031981
11	....	67043999
12	....	73019975
13	....	91007981
14	....	37055993
15	....	1091975
16	....	19067999
17	....	25055999
18	....	25061999
19	....	25079987
20	....	67031999
21	....	67025999
22	....	79031999
23	....	43055999
24	....	43055999

Priority Value of Each Chain

Figure IV-4

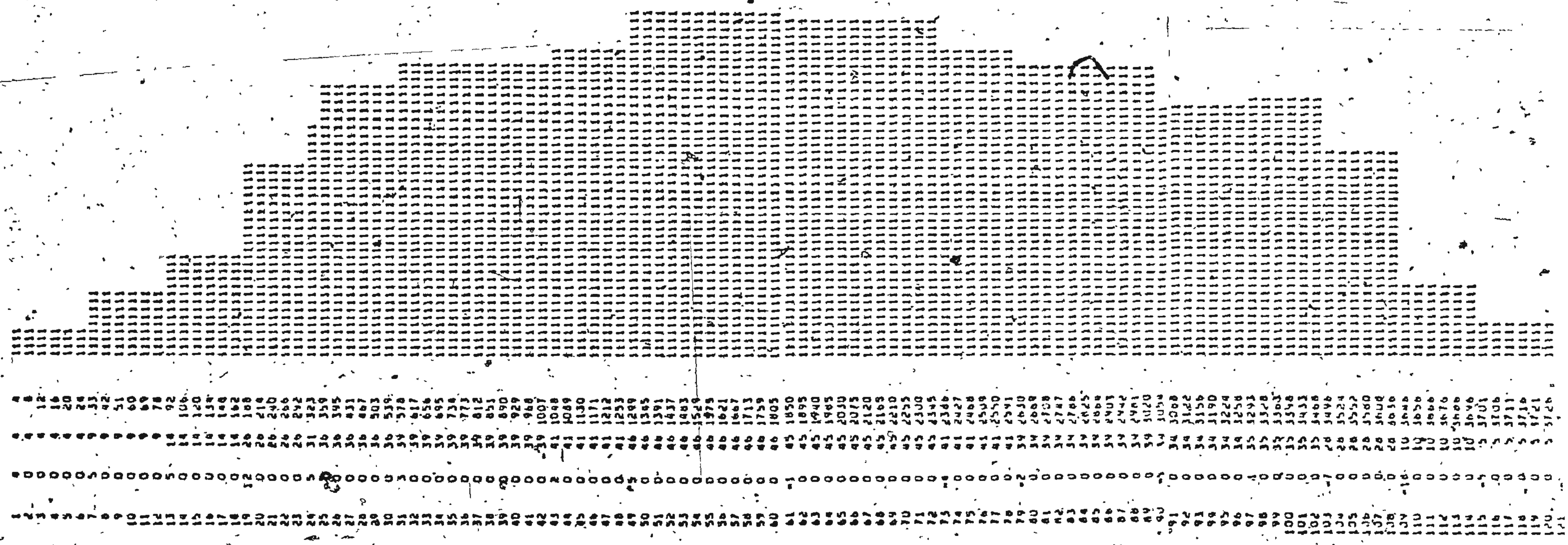


Initial Resource Profile

Figure IV - 5

Re-scheduler Phase looks over the resource profile developed by Scheduler Phase II. If any unwanted peaks are encountered, this phase attempts to reschedule activities to remove these peaks. Figure IV-6 is the final optimal resource profile in which row 1 (the bottom row) depicts the date, row 2 (above row 1) shows resource change for each day, row 3 represents resource requirement for each day, and row 4 indicates cumulative resources used to date.

In Figure IV-5 then are two unwanted peaks which occur for day 79 to day 84 and for day 97 to day 102. The rescheduler Phase removes the first peak as shown in Figure IV-5. The attempt to remove the second peak has not been successful. The optimal schedule, therefore has an unwanted peak. The reason is that this peak could not be removed without increasing the total variance of resource change. Activity A(6) starting on day 31 and having resource requirement of 6 units and duration of 72 days was moved backwards to start on day 25, this forced a larger peak in the first half of the resource profile so that this new schedule was unacceptable. Table IV-1 is the final optimal schedule.



Final Optimal Resource Profile

Figure IV - 6

NO	I	J	D	R	ES1	LF1	ES2	LF2	CODE	SCHEDULED DATE
1	5	10	6	5	1	1	7	19	10	13
2	5	40	60	6	1	1	61	85	10	19
3	5	45	48	4	1	1	49	49	20	1
4	5	60	60	5	1	1	61	85	10	24
5	5	75	72	5	1	1	73	91	10	7
6	5	80	72	6	1	1	73	115	10	31
7	5	85	24	6	1	1	73	115	10	49
8	10	15	6	4	7	19	13	25	10	19
9	10	120	48	7	7	19	55	97	10	19
10	15	20	6	9	13	25	19	31	10	25
11	15	25	6	6	13	25	19	73	10	55
12	15	30	6	1	13	25	19	73	10	43
13	15	35	6	1	13	25	19	73	10	43
14	20	45	18	6	19	31	49	49	10	31
15	25	30	0	0	19	73	19	73	11	61
16	25	40	0	0	19	73	61	85	11	85
17	30	50	6	6	19	73	25	79	10	61
18	35	55	6	6	19	73	25	79	10	49
19	40	90	6	9	61	85	67	91	10	85
20	45	65	12	5	49	49	61	61	20	49
21	50	70	12	6	25	79	37	91	10	67
22	50	110	0	0	25	79	79	79	11	79
23	55	60	0	0	25	79	61	85	11	84
24	55	110	0	0	25	79	79	79	11	55
25	55	170	12	9	25	79	115	115	10	85
26	60	100	6	5	61	85	73	91	10	84
27	65	95	12	4	61	61	73	73	20	61
28	70	75	0	0	37	91	73	91	11	79
29	70	115	6	9	37	91	43	97	10	79
30	75	105	6	9	73	91	79	97	10	79
31	80	85	0	0	73	115	73	115	11	103
32	85	175	6	4	73	115	121	121	10	103
33	90	100	0	0	67	91	73	91	11	91
34	90	125	0	0	67	91	79	97	11	97
35	90	130	0	0	67	91	67	97	11	97
36	90	150	0	0	67	91	91	91	11	91
37	90	160	0	0	67	91	67	109	11	97
38	95	100	0	0	73	73	73	91	11	91
39	95	110	6	6	73	73	79	79	20	73
40	100	135	12	1	73	91	85	103	10	91
41	105	125	0	0	79	97	79	97	11	85
42	105	165	0	0	79	97	103	109	11	85
43	110	145	12	5	79	79	91	91	20	79
44	115	120	0	0	43	97	55	97	11	85
45	115	130	0	0	43	97	67	97	11	85
46	120	160	12	7	55	97	67	109	10	67
47	125	140	6	5	79	97	85	103	10	85
48	130	155	6	10	67	97	73	103	10	97
49	135	170	12	1	85	103	115	115	10	103
50	140	170	12	6	85	103	115	115	10	91
51	145	150	0	0	91	91	91	91	21	91
52	145	165	12	5	91	91	103	109	10	91
53	150	170	24	7	91	91	115	115	20	91
54	155	170	12	2	73	103	115	115	10	103
55	160	170	6	5	67	109	115	115	10	79
56	165	170	6	14	103	109	115	115	10	103
57	170	175	6	5	115	115	121	121	20	115

Table IV-1

CHAPTER V

CONCLUSION

This thesis has presented an heuristic procedure for resource levelling problems to achieve an optimal parabolic resource profile using data obtained from CPM analysis of a project.

A mathematical based objective function was developed for this heuristic in Section 1 of Chapter II from a theoretical approach to the resource levelling problem, without considering the activity constraints found in a CPM network. This objective function is based upon the measure of gradualness of resource change and is defined as the minimization of variance of the resource change function. Then, in Section 2 of Chapter II these activity constraints are taken into account and optimization of the activity schedule is achieved yielding an optimum resource profile. This procedure, although not feasible for large project, provides a building block upon which an heuristic procedure achieving an optimal resource profile for large scale project can be developed. The heuristic, outlined in Section 3 of Chapter II, has as its base the reduction of the number of combinations by a build-up of activity chains which are then considered as single activities. These chains are then scheduled to yield an initial resource profile followed by a look over procedure. If any unwanted peaks are present an attempt is made to remove them, by breaking up the chains into constituent activities and rescheduling the single activities. Only such readjustments are accepted, which reduce the total variance of resource change. The cut-off point in the search for the final solution is provided by the minimization function for the variance of resource change.

Based on the computational procedure for this heuristic in Chapter III a computer programme called BACS (Build-up Activity Chain Schedule) has been developed using FORTRAN IV IBM 370. The programme has been tested on a number of projects. Chapter IV shows an example illustrating the use and output analysis of BACS.

The heuristic algorithm and BACS as presented here have achieved the following:

1) A parabolic resource profile based on a mathematical model where the variance of resource change is minimized subject to CPM network constraints.
2) Optimal resource usage within a fixed project duration.
3) The variance of the differential of the resource requirement function provides a measure for the gradualness of resource change.
4) BACS is capable of handling large scale projects within a short computational time, the example in Chapter IV which contains 57 activities has a computational time of 30 seconds.

5) BACS is very easy to use. With minor difficulties it can also be linked to a CPM scheduling programme from which it can obtain part of its input data. The data about resources can be fed directly, thus further facilitating its use.

Future Research

Although the heuristic develops an optimal resource profile, if a smoother or more gradual resource curve would be useful for particular projects, some further refinements in the heuristic which may improve the final resource profile without greatly increasing the computation time are:

1) A new basis for building activity chains . . .
2) Allowing interruption in the activity chains.
3) Changing the priority rules.
4) Allowing the Re-scheduler Phase to push forward or backward more than one activity at a time.
5) Allowing the re-scheduler to look at the height of the steps in an attempt to decrease the height of any steep steps which may occur.
6) Relaxing the constraint of fixed resources and fixed duration for activities (i.e. an activity taking 4 days and 8 resource units can be allowed to take 8 days and 4 resource units)

BIBLIOGRAPHY

1. Ahuja, H.N., Systems Approach to Project and Control, Memorial University of Newfoundland, 1972.

2. Burgess, A.R. and J.B.Killebrew, Variation in Activity Level on a Cyclical Arrow Diagram, The Journal of Industrial Engineering, Vol XIII, no.2.

3. De Ambrogio, W., A Computer for Multiproject Scheduling with Smoothing of Resource Requirements, Paper, Institute of Management Science, Zurich, Sept. 1964.

4. De Witte, L., Manpower Levelling of PERT Networks, Data Processing for Science/Engineering, 2.2,1964, p29.

5. Hooper, P.C., Resource Allocation and Levelling, Proceedings of the Third CPA Symposium, Operation Research Society, London, 1965.

6. IBM, Resource Allocation Processor (RAP), PMS IV, IBM programme Number 5734-xp4.

7. IBM, Resource Allocation (REAL/360), IBM programme number 5736-xp2.

8. Levy, F.K., G.L.Thompson, and J.D.Wiest, Multi-ship, Multi-Shop Workload Smoothing Program, Naval Research Logistics Quartely, 9.1. (March 1962) p37-44.

9. Martino, R.L., Allocating and Scheduling resources, American Management Association, Project Management and Control Vol.III, New York.

10. Moshman, J., J.Johnson, and Madalyn Larsen, RAMPS: A Tecnique for Resource Allocation and Multiproject Scheduling, Proceedings of the 1963 Spring Joint Computer Conference.

11. Petrovic, R., Optimization of Resource Allocation in Project Planning, Operational Research Society, 16, 559.

12. Petrovic, R., On Optimization of Resource Levelling in Project Plans, Proc. Second Int. Cong. of Internet, North-Holland, Amsterdam, p268.

13. Prabhakar, T., Resource Allocation/ Levelling in Engineeing Construction Projects, Paper presented at the 37th Meeting of ORSA 1970.

14. Tamura, H., Dual Algorithm for solving Large Scale Scheduling Control Problem with Hierarchical Structure, International Federation of Automatic Control, World Congress 5th, 1972.

15. Thierauf, R.J., and R.A.Grosse, Decision Making through Operations Research, John Wiley and Sons, Inc., New York, 1970.

16. Wiest, J.D., and F.K.Levy, A Management Guide to PERT/CPM, Prentice-Hall Inc., Englewood Cliffs, New Jersey, 1969.

17. Wilson, R.C., Assembly Line Balancing and Resource Scheduling, University of Michigan Summer Conference on Production and Inventory Control, 1964.

APPENDIX-A

BASS

(Build All Starts Schedule)

A computer programme for optimum resource levelling

The programme listing along with input data for the example of Figure II-4 follows:

```
      DIMENSION CPR(25),WAR(25),CMR(25)
      DIMENSION MM(10)
      DIMENSION PRI(4,5)
      DIMENSION PRX(5)
      DO 10 I=1,25
      CPR(I)=0.0
   10 CONTINUE
      CPR(1)=4.0
      CPR(9)=-1.0
      CPR(15)=2.0
      CPR(25)=-5.0
      CMIN=999.0
      DO 1000 I4=1,8
      DO 1000 I3=1,9
      DO 1000 I2=1,12
      DO 1000 I1=1,7
      DO 20 I=1,25
      WAR(I)=CPR(I)
   20 CONTINUE
      IA1=14+I1
      IB1=IA1+4
      IA2=10+I2
      IB2=IA2+3
      IA3=I3
      IB3=IA3+6
      IA4=I4
      IB4=IA4+3
      WAR(IA1)=WAR(IA1)+3.0
      WAR(IB1)=WAR(IB1)-3.0
      WAR(IA2)=WAR(IA2)+1.0
      WAR(IB2)=WAR(IB2)-1.0
      WAR(IA3)=WAR(IA3)+7.0
      WAR(IB3)=WAR(IB3)-7.0
      WAR(IA4)=WAR(IA4)+4.0
      WAR(IB4)=WAR(IB4)-4.0
      TTL=0
      DO 30 I=1,25
      TTL=TTL+WAR(I)*WAR(I)
   30 CONTINUE
      IF(TTL.LT.CMIN) GO TO 2000
 1000 CONTINUE
      GO TO 3000
 2000 CMIN=TTL
      DO 40 I=1,25
      CMR(I)=WAR(I)
   40 CONTINUE
      MI1=I1
      MI2=I2
```

```
      MI3=I3
      MI4=I4
      GO TO 1000
 3000 PRI(1,1)=3.0
      PRI(1,2)=4.0
      PRI(1,3)=1.0
      PRI(1,4)=8.0
      PRI(2,1)=6.0
      PRI(2,2)=7.0
      PRI(2,3)=1.0
      PRI(2,4)=9.0
      PRI(3,1)=3.0
      PRI(3,2)=1.0
      PRI(3,3)=11.0
      PRI(3,4)=12.0
      PRI(4,1)=4.0
      PRI(4,2)=3.0
      PRI(4,3)=15.0
      PRI(4,4)=7.0
      PRINT 50
      PRINT 50
      PRINT 51
      PRINT 50
      PRINT 50
      PRINT 52
      PRINT 50
      PRI(1,5)=MI4+PRI(1,3)-1.0
      PRI(2,5)=MI3+PRI(2,3)-1.0
      PRI(3,5)=MI2+PRI(3,3)-1.0
      PRI(4,5)=MI1+PRI(4,3)-1.0
      DO 60 J=1,5
      PRX(J)=PRI(1,J)
   60 CONTINUE
      PRINT 61,(PRX(J),J=1,5)
      DO 70 J=1,5
      PRX(J)=PRI(2,J)
   70 CONTINUE
      PRINT 62,(PRX(J),J=1,5)
      DO 80 J=1,5
      PRX(J)=PRI(3,J)
   80 CONTINUE
      PRINT 63,(PRX(J),J=1,5)
      DO 90 J=1,5
      PRX(J)=PRI(4,J)
   90 CONTINUE
      PRINT 64,(PRX(J),J=1,5)
      PRINT 100
      PRINT 51
```

```
      PRINT 50
      PRINT 101
      PRINT 103
      DO 200 J=1,24
      DO 110 J1=1,10
      MM(J1)=0
  110 CONTINUE
      DATA=0.0
      DO 111 J2=1,J
      DATA=DATA+CMR(J2)
  111 CONTINUE
      L=1
  112 IF(DATA.LE.0.0) GO TO 140
      MM(L)=11111
      L=L+1
      DATA=DATA-1.0
      GO TO 112
  140 PRINT 104,(J,(MM(L),L=1,10))
      PRINT 105,(MM(L),L=1,10)
  200 CONTINUE
      PRINT 103
      PRINT 101
   50 FORMAT(1H0)
   51 FORMAT(1H )
   52 FORMAT(1H ,75H                  A(J)  DURATION  RESOURCE
     *    FLOAT SCHEDULED-DATE)
   61 FORMAT(1H ,16X,4HA(1),5F10.1)
   62 FORMAT(1H ,16X,4HA(2),5F10.1)
   63 FORMAT(1H ,16X,4HA(3),5F10.1)
   64 FORMAT(1H ,16X,4HA(4),5F10.1)
  100 FORMAT(1H1)
  101 FORMAT(1H ,60H         0    1    2    3    4    5    6    7    8
     *  9    10)
  102 FORMAT(1H0)
  103 FORMAT(1H )
  104 FORMAT(1H ,4X,I2,3X,1H1,10I5)
  105 FORMAT(1H ,8X,2H-1,10I5)
      END
```

APPENDIX-B

BACS

(Build-up Activity Chain Schedule)

Computer programme for optimal resource scheduling

The information on BACS is given in the Following order:

(1) Source Listing

(2) Input Format

(3) Input Deck

Source Listing

```
C
C
C
C
C
C      INITIALIZATION
C
C
C
C
C
      INTEGER*2 IDT(30),MWT(100,7),MVAR(200,3),MRES(100)
      INTEGER*2 IAF(100,61),ICA(30,30),ISC(30,30)
C
C
      NC=0
      NCX=0
      DO 100 J1=1,100
      DO 100 K1=1,61
      IAF(J1,K1)=0
  100 CONTINUE
      DO 160 J1=1,30
      DO 150 K1=1,30
      ICA(J1,K1)=0
      ISC(J1,K1)=0
  150 CONTINUE
      IDT(K1)=0
  160 CONTINUE
C
C     READ INPUT DATA
C
      DO 450 J1=1,101
      READ (3,200)(IDT(J2),J2=1,8)
  200 FORMAT(8(1X,I4))
      IF(IDT(1).EQ.99) GO TO 500
      DO 250 K1=1,8
      IAF(J1,K1)=IDT(K1)
  250 CONTINUE
      IF(IAF(J1,5).NE.IAF(J1,6)) GO TO 350
      IF(IAF(J1,7).NE.IAF(J1,8)) GO TO 350
      L=IAF(J1,8)-IAF(J1,3)-IAF(J1,6)
      IF(L.NE.0) GO TO 350
      IF(IAF(J1,4).NE.0) GO TO 300
      IAF(J1,11)=21
      GO TO 450
  300 IAF(J1,11)=20
      GO TO 450
  350 IF(IAF(J1,4).NE.0) GO TO 400
```

```
      IAF(J1,11)=11
      GO TO 450
  400 IAF(J1,11)=10
  450 CONTINUE
  500 NOI=J1-1
C
C
      PRINT 510
  510 FORMAT(1H ,15X,40HN E T W O R K      I N F O R M A T I O N)
  515 FORMAT(1H ,15X,40H----------------------------------------)
      PRINT 515
      PRINT 525
  525 FORMAT(1H ,10X,50H   NO    I    J    D    R  ES1  LF1  ES2  LF2 CO
     *DE)
      DO 650 J1=1,NOI
      DO 550 K1=1,20
      IDT(K1)=IAF(J1,K1)
  550 CONTINUE
      PRINT 600,J1,(IDT(K1),K1=1,8),IDT(11)
  600 FORMAT(1H ,10X,10I5)
  650 CONTINUE
      PRINT 9980
      MSTG=1
C
C
  780 MSG=2
      DO 1000 J1=1,NOI
      IF(IAF(J1,11).GE.20) GO TO 1000
      L14=0
      L15=0
      L16=0
      L17=0
      DO 950 K1=1,NOI
      IF(IAF(K1,11).GE.20) GO TO 950
      IF(IAF(J1,1).NE.IAF(K1,2)) GO TO 800
      L14=L14+1
  800 IF(IAF(J1,1).NE.IAF(K1,1)) GO TO 850
      L15=L15+1
  850 IF(IAF(J1,2).NE.IAF(K1,2)) GO TO 900
      L16=L16+1
  900 IF(IAF(J1,2).NE.IAF(K1,1)) GO TO 950
      L17=L17+1
  950 CONTINUE
      IAF(J1,14)=L14
      IAF(J1,15)=L15
      IAF(J1,16)=L16
      IAF(J1,17)=L17
 1000 CONTINUE
```

```
C
C
C
C
C
C           SCHEDULER PHASE - I
C
C
C
C
C
      J1=1
      I1=1
 1700 IF(IAF(J1,14).EQ.1) GO TO 1950
      IF(IAF(J1,16).NE.1) GO TO 1950
      IF(IAF(J1,17).NE.1) GO TO 1950
      M1=1
      ICA(I1,M1)=J1
      IAF(J1,13)=1
      N2=J1
 1750 N1=1
 1800 IF(IAF(N1,11).GE.20) GO TO 1850
      IF(IAF(N1,1).NE.IAF(N2,2)) GO TO 1850
      M1=M1+1
      IF(M1.LE.29) GO TO 1820
      MERR=1
      GO TO 9990
 1820 ICA(I1,M1)=N1
      IF(IAF(N1,16).NE.1) GO TO 1900
      IF(IAF(N1,17).NE.1) GO TO 1900
      IAF(N1,13)=2
      N2=N1
      GO TO 1750
 1850 N1=N1+1
      IF(N1.LE.NOI) GO TO 1800
      MERR=2
      GO TO 9990
 1900 IAF(N1,13)=3
      ICA(I1,30)=N1
      I1=I1+1
      IF(I1.LE.29) GO TO 1950
      MERR=3
      GO TO 9990
 1950 J1=J1+1
      IF(J1.LE.NOI) GO TO 1700
C
C
C
```

```
C
C
C
C         MAKE ACTIVITY CHAINS
C
C
 2230 MSG=3
      J1=1
 2250 IF(IAF(J1,11).GE.20) GO TO 2450
      L1=IAF(J1,5)+IAF(J1,3)-IAF(J1,7)
      L2=IAF(J1,6)+IAF(J1,3)-IAF(J1,8)
      IF(L1.EQ.0) GO TO 2300
      IF(L2.EQ.0) GO TO 2400
      GO TO 2450
 2300 IF(L2.EQ.0) GO TO 2350
      IAF(J1,19)=1
      GO TO 2450
 2350 IAF(J1,19)=2
      GO TO 2450
 2400 IAF(J1,19)=3
 2450 J1=J1+1
      IF(J1.LE.NOI) GO TO 2250
C
C         STORE ACTIVITY CHAINS
C
C
      I1=1
      J1=1
 2500 IF(IAF(J1,11).GE.20) GO TO 2950
      IF(IAF(J1,19).NE.1) GO TO 2950
      J3=J1
      M1=1
      ISC(I1,1)=J1
      IAF(J1,18)=1
 2550 M1=M1+1
      IF(M1.LE.29) GO TO 2570
      MERR=4
      GO TO 9990
 2570 J2=1
 2600 IF(IAF(J2,11).GE.20) GO TO 2900
      IF(IAF(J2,1).NE.IAF(J3,2)) GO TO 2900
      IF(IAF(J2,19).EQ.0) GO TO 2900
C
C
C
      JJ=1
 2650 KK=1
      IF(ISC(JJ,1).EQ.0) GO TO 2800
```

```
 2700 IF(ISC(JJ,KK).EQ.J2) GO TO 2900
      IF(ISC(JJ,KK).EQ.0) GO TO 2750
      KK=KK+1
      IF(KK.LE.29) GO TO 2700
      MERR=5
      GO TO 9990
 2750 JJ=JJ+1
      IF(JJ.LE.30) GO TO 2650
      MERR=6
      GO TO 9990
 2800 IF(IAF(J2,19).EQ.3) GO TO 2850
      ISC(I1,M1)=J2
      IAF(J2,18)=2
      J3=J2
      GO TO 2550
 2850 ISC(I1,M1)=J2
      ISC(I1,30)=J2
      I1=I1+1
      IF(I1.LE.30) GO TO 2870
      MERR=7
      GO TO 9990
 2870 IAF(J2,18)=3
      GO TO 2950
 2900 J2=J2+1
      IF(J2.LE.NOI) GO TO 2600
      ISC(I1,30)=J3
      I1=I1+1
      IF(I1.LE.30) GO TO 2920
      MERR=8
      GO TO 2950
 2920 IAF(J3,18)=3
 2950 J1=J1+1
      IF(J1.LE.NOI) GO TO 2500
      J1=1
 3000 IF(IAF(J1,11).GE.20) GO TO 3950
      IF(IAF(J1,19).NE.2) GO TO 3950
      JJ=1
 3050 KK=1
      IF(ISC(JJ,1).EQ.0) GO TO 3200
 3100 IF(ISC(JJ,KK).EQ.J1) GO TO 3950
      IF(ISC(JJ,KK).EQ.0) GO TO 3150
      KK=KK+1
      IF(KK.LE.30) GO TO 3100
      MERR=9
      GO TO 9990
 3150 JJ=JJ+1
      IF(JJ.LE.30) GO TO 3050
      MERR=10
```

```
      GO TO 9990
 3200 J2=1
 3250 IF(IAF(J2,11).GE.20) GO TO 3450
      IF(IAF(J2,2).NE.IAF(J1,1)) GO TO 3450
      IF(IAF(J2,19).EQ.0) GO TO 3450
      JJ=1
 3300 KK=1
      IF(ISC(JJ,1).EQ.0) GO TO 3950
 3350 IF(ISC(JJ,KK).EQ.J2) GO TO 3450
      IF(ISC(JJ,KK).EQ.0) GO TO 3400
      KK=KK+1
      IF(KK.LE.30) GO TO 3350
      MERR=11
      GO TO 9990
 3400 JJ=JJ+1
      IF(JJ.LE.30) GO TO 3300
      MERR=12
      GO TO 9990
 3450 J2=J2+1
      IF(J2.LE.NOI) GO TO 3250
C
C
C
C
 3500 J3=J1
      M1=1
      ISC(I1,1)=J1
      IAF(J1,18)=1
 3550 M1=M1+1
      IF(M1.LE.29) GO TO 3570
      MERR=13
      GO TO 9990
 3570 J2=1
 3600 IF(IAF(J2,1).GE.20) GO TO 3900
      IF(IAF(J2,1).NE.IAF(J3,2)) GO TO 3900
      IF(IAF(J2,19).EQ.0) GO TO 3900
      JJ=1
 3650 KK=1
      IF(ISC(JJ,1).EQ.0) GO TO 3800
 3700 IF(ISC(JJ,KK).EQ.J2) GO TO 3900
      IF(ISC(JJ,KK).EQ.0) GO TO 3750
      KK=KK+1
      IF(KK.LE.29) GO TO 3700
      MERR=14
      GO TO 9990
 3750 JJ=JJ+1
      IF(JJ.LE.29) GO TO 3650
      MERR=15
```

```
      GO TO 9990
 3800 IF(IAF(J2,19).EQ.3) GO TO 3850
      ISC(I1,M1)=J2
      IAF(J2,18)=2
      J3=J2
      GO TO 3550
 3850 ISC(I1,M1)=J2
      ISC(I1,30)=J2
      I1=I1+1
      IAF(J2,18)=3
      GO TO 3950
 3900 J2=J2+1
      IF(J2.LE.NOI) GO TO 3600
      ISC(I1,30)=J3
      I1=I1+1
      IAF(J3,18)=3
      IF(I1.LE.30) GO TO 3950
      MERR=16
      GO TO 9990
 3950 J1=J1+1
      IF(J1.LE.NOI) GO TO 3000
C
C
C
C
C
C
 4000 DO 4010 J1=1,30
      IF(ISC(J1,1).EQ.0) GO TO 4015
 4010 CONTINUE
 4015 DO 4020 K1=1,NOI
      IF(IAF(K1,11).GE.20) GO TO 4020
      IF(IAF(K1,18).NE.0) GO TO 4020
      IAF(K1,18)=5
      ISC(J1,1)=K1
      J1=J1+1
 4020 CONTINUE
      DO 4025 I1=1,30
      IF(ISC(I1,1).EQ.0) GO TO 4028
 4025 CONTINUE
 4028 I1=I1-1
      PRINT 9980
      PRINT 4029
 4029 FORMAT(1H0,20X,30HA C T I V I T Y     C H A I N S)
      PRINT 4030
 4030 FORMAT(1H0,10X,29H   NO              ACTIVITY NUMBER)
      DO 4100 J1=1,I1
      DO 4050 K1=1,30
```

```
      IDT(K1)=ISC(J1,K1)
 4050 CONTINUE
      PRINT 4040,J1,(IDT(K1),K1=1,19)
 4040 FORMAT(1H0,10X,I5,5X,19I5)
 4100 CONTINUE
      PRINT 9980
C
C
 5000 DO 5100 J1=1,NOI
      I1=20
      I2=40
      DO 5100 J2=1,NOI
      IF(IAF(J1,1).NE.IAF(J2,2)) GO TO 5050
      IAF(J1,I1)=J2
      I1=I1+1
      IF(I1.LE.39) GO TO 5050
      MERR=17
      GO TO 9990
 5050 IF(IAF(J1,2).NE.IAF(J2,1)) GO TO 5100
      IAF(J1,I2)=J2
      I2=I2+1
      IF(I2.LE.59) GO TO 5100
      MERR=18
      GO TO 9990
 5100 CONTINUE
C
C
C
C
 5150 DO 5200 I1=1,100
      DO 5200 M1=1,7
      MWT(I1,M1)=0
 5200 CONTINUE
      MP=13
      I1=1
      IF(MSG.EQ.1) GO TO 5500
      IF(MSG.EQ.3) GO TO 5350
      DO 5250 II1=1,30
      IF(ICA(II1,1).EQ.0) GO TO 5300
      MWT(I1,1)=ICA(II1,1)
      JK=ICA(II1,1)
      MWT(I1,2)=IAF(JK,8)-IAF(JK,3)-IAF(JK,5)+1
      I1=I1+1
      IF(I1.LE.100) GO TO 5250
      MERR=19
      GO TO 9990
 5250 CONTINUE
 5300 MP=13
```

```
      GO TO 5500
 5350 DO 5400 II1=1,30
      IF(ISC(II1,1).EQ.0) GO TO 5450
      MWT(I1,1)=ISC(II1,1)
      JK=ICA(II1,1)
      MWT(I1,2)=IAF(JK,8)-IAF(JK,3)-IAF(JK,5)+1
      I1=I1+1
      IF(I1.LE.100) GO TO 5400
      MERR=20
      GO TO 9990
 5400 CONTINUE
 5450 MP=18
 5500 DO 5600 J1=1,NOI
      IF(IAF(J1,MP).NE.0) GO TO 5600
      IF(IAF(J1,11).GE.20) GO TO 5600
      MWT(I1,1)=J1
      MWT(I1,2)=IAF(J1,8)-IAF(J1,3)-IAF(J1,5)+1
      I1=I1+1
      IF(I1.LE.100) GO TO 5600
      MERR=21
      GO TO 9990
 5600 CONTINUE
      I1=I1-1
      PRINT 5650,(MWT(MM,1),MM=1,50)
      PRINT 9980
 5650 FORMAT(1H ,50I2,13H----MWT(I1,1))
      IF(MSG.NE.3) GO TO 5690
      DO 5680 II1=1,30
      DO 5680 II2=1,30
      ICA(II1,II2)=ISC(II1,II2)
 5680 CONTINUE
 5690 IF(I1.LE.30) GO TO 5700
      MERR=22
      GO TO 9990
C
C
C
C
C
C
C
C
C
C
C
C
C
C
```

```
      C
      C
      C
      C         SCHEDULER PHASE - II
      C
      C
      C
      C
      C
      C
      C
 5700 DO 5750 I1=1,100
      DO 5750 I2=3,6
      MWT(I1,I2)=0
 5750 CONTINUE
      DO 5800 I1=1,200
      DO 5800 I2=1,2
      MVAR(I1,I2)=0
 5800 CONTINUE
      DO 5850 I1=1,NOI
      IAF(I1,50)=0
      IF(IAF(I1,11).LE.19) GO TO 5850
      M1=IAF(I1,5)
      M2=IAF(I1,7)
      MVAR(M1,2)=MVAR(M1,2)+IAF(I1,4)
      MVAR(M2,2)=MVAR(M2,2)-IAF(I1,4)
      IAF(I1,50)=IAF(I1,5)
 5850 CONTINUE
      C
      C
      C
      C    PRIORITY RULE IS APPLIED
      C
      C
      C
      MM=0
      DO 5870 I1=1,100
      IF(MWT(I1,1).EQ.0) GO TO 5880
      MM=MM+1
 5870 CONTINUE
 5880 DO 5950 I1=1,MM
      MWT(I1,4)=IAF(MWT(I1,1),5)
      IF(ICA(I1,1).NE.0) GO TO 5885
      MTD=IAF(MWT(I1,1),3)
      GO TO 5900
 5885 MTD=0
      DO 5890 MY=1,30
      IF(ICA(I1,MY).EQ.0) GO TO 5900
```

```
      MTD=MTD+IAF(ICA(I1,MY),3)
 5890 CONTINUE
 5900 MWT(I1,5)=999-MTD
      MYS=MWT(I1,1)
      MWT(I1,6)=IAF(MYS,8)-IAF(MYS,3)-IAF(MYS,5)+1
 5950 CONTINUE
C
C
C
C
C
C
 6000 MIN=999999999
      PRINT 9980
      MAX=0
      DO 6001 MST=1,NOI
      IF(IAF(MST,8).LT.MAX) GO TO 6001
      MAX=IAF(MST,8)
 6001 CONTINUE
      MTY=0
      MMTY=0
      DO 6005 I1=1,MAX
      MTY=MTY+MVAR(I1,2)
      MMTY=MMTY+MTY
      IF(MTY.GE.100) GO TO  6005
      DO 6002 KK2=1,100
      MRES(KK2)=0
 6002 CONTINUE
      DO 6003 KK1=1,MTY
      MRES(KK1)=1
 6003 CONTINUE
      PRINT 6004,(I1,MVAR(I1,2),MTY,MMTY,(MRES(KK1),KK1=1,MTY))
 6004 FORMAT(4I5,5X,100I01)
 6005 CONTINUE
 6006 FORMAT(1H0,30X,28HPRIORITY VALUE OF EACH CHAIN)
 6007 FORMAT(1H0,20X,40H CHAIN NO.                         VALUE )
      DO 6050 I1=1,MM
      ND=MWT(I1,4)*1000000+MWT(I1,6)*1000+MWT(I1,5)
      PRINT 6010,I1,ND
 6010 FORMAT(1H0,23X,I3,5X,13H.............,5X,I10)
      IF(MIN.LE.ND) GO TO 6050
      MIN=ND
      MINN=I1
 6050 CONTINUE
      PRINT 9980
      MFLOA=MWT(MINN,6)
      MWT(MINN,4)=999
      MWT(MINN,5)=999
      MWT(MINN,6)=999
```

```
      PRINT 6080,MINN
 6080 FORMAT(1H ,I5,7H,..,MINN)
C
C
C
C
C
C
      MINV=999999
      DO 6350 I1=1,MFLOA
      DO 6100 I5=1,200
      MVAR(I5,1)=MVAR(I5,2)
 6100 CONTINUE
C
C
C     FOLLOWING IS A CHECK ROUTINE WHETHER FEASIBLE OR NOT
C
C
C
C
      DO 6110 N1=1,30
      IF(ICA(MINN,N1).EQ.0) GO TO 6120
      KK1=IAF(ICA(MINN,N1),5)+I1    -1+IAF(ICA(MINN,N1),3)
      KK3=IAF(ICA(MINN,N1),5)+I1    -1
      KK2=ICA(MINN,N1)
      DO 6105 L1=1,NOI
      IF(IAF(L1,2).NE.IAF(KK2,1)) GO TO 6105
      IF(IAF(L1,50).EQ.0) GO TO 6105
      IFD=IAF(L1,50)+IAF(L1,3)
 6102 FORMAT(1H ,4I10,20HKK2,..L1,..KK3,..IFD)
      IF(IFD.GT.KK3) GO TO 6350
 6105 CONTINUE
      DO 6110 L1=1,NOI
      IF(IAF(L1,1).NE.IAF(KK2,2)) GO TO 6110
      IF(IAF(L1,50).EQ.0) GO TO 6110
 6108 FORMAT(1H ,40X,4I10,21HKK2,K1,KK1,IAF(L1,50))
      IF(KK1.GT.IAF(L1,50)) GO TO 6350
 6110 CONTINUE
 6120 DO 6200 M1=1,30
      IF(ICA(MINN,M1).EQ.0) GO TO 6250
C
C
C
C
C
C
      MA=IAF(ICA(MINN,M1),5)+I1-1
      MB=MA+IAF(ICA(MINN,M1),3)
```

```
      MVAR(MA,1)=MVAR(MA,1)+IAF(ICA(MINN,M1),4)
      MVAR(MB,1)=MVAR(MB,1)-IAF(ICA(MINN,M1),4)
 6200 CONTINUE
 6250 MVK=0
      DO 6300 I3=1,200
      MVK=MVK+MVAR(I3,1)**2
 6300 CONTINUE
      PRINT 6320,MVK,MINV,MF
 6320 FORMAT(1H ,3I10,16H...MVK AND MINV)
      NOOI=MAX/2+5
      MSTT=IAF(ICA(MINN,1),5)
      DO 6330 MNA=1,30
      IF(ICA(MINN,MNA).EQ.0) GO TO 6340
 6330 CONTINUE
 6340 MNA=MNA-1
      MEND=IAF(ICA(MINN,MNA),8)
      MCHE=(MSTT+MEND)/2
      PRINT 6341,NOOI,MSTT,MNA,MEND,MCHE
 6341 FORMAT(1H ,50X,5I10)
      IF(MCHE.LE.NOOI) GO TO 6342
      IF(MINV.LE.MVK) GO TO 6350
      GO TO 6345
 6342 IF(MINV.LT.MVK) GO TO 6350
 6345 MF=I1
      MINV=MVK
C
C
C
C
C
C
      DO 6350 MM1=1,200
      MVAR(MM1,3)=MVAR(MM1,1)
 6350 CONTINUE
      MWT(MINN,7)=IAF(MWT(MINN,1),5)+MF-1
      DO 6370 I3=1,30
      IF(ICA(MINN,I3).EQ.0) GO TO 6372
      IAF(ICA(MINN,I3),50)=IAF(ICA(MINN,I3),5)+MF-1
 6370 CONTINUE
 6372 DO 6375 I1=1,200
      MVAR(I1,2)=MVAR(I1,3)
 6375 CONTINUE
 6380 DO 6400 I1=1,MM
      IF(MWT(I1,4).LT.999) GO TO 6000
 6400 CONTINUE
      PRINT 6450,(MWT(I1,7),I1=1,MM)
 6450 FORMAT(1H ,30I3)
      PRINT 6500,(IAF(I1,50),I1=1,NOI)
```

```
 6500 FORMAT(1H ,31HSTART DATE OF EACH ACTIVITY.....,20I3)
      MC=1
 6530 MM=0
      PRINT 9980
      PRINT 6535
 6535 FORMAT(1H0,18X,40HOPTIMAL SCHEDULED DATE FOR EACH ACTIVITY)
      PRINT 6540
 6540 FORMAT(1H0,10X,67H   NO    I    J    D    R  ES1  LF1  ES2  LF2 CO
     *DE   SCHEDULED DATE)
      DO 6545 J1=1,NOI
      DO 6542 K1=1,20
      IDT(K1)=IAF(J1,K1)
 6542 CONTINUE
      PRINT 6547,J1,(IDT(K1),K1=1,8),IDT(11),IAF(J1,50)
 6545 CONTINUE
 6547 FORMAT(1H ,10X,10I5,I10)
 6548 FORMAT(1H0,18X,I5,20X,I6)
      PRINT 9980
      MTY=0
      MSS=0
      DO 7000 I1=1,MAX
      MTY=MTY+MVAR(I1,2)
      MSS=MSS+MTY
      DO 6550 KK2=1,100
      MRES(KK2)=0
 6550 CONTINUE
      DO 6700 KK1=1,MTY
      MRES(KK1)=1
 6700 CONTINUE
      PRINT 7004,(I1,MVAR(I1,2),MTY,MSS,(MRES(KK1),KK1=1,MTY))
 7000 CONTINUE
 7004 FORMAT(4I5,5X,100I01)
      IF(MC.EQ.1) GO TO 7050
      GO TO 9999
C
C
C
C
C
C
C
C
C
C
C
C
C
```

```
C
C
C
C
C
C
C         RE-SCHEDULER PHASE
C
C
C
C
C
 7050 MDATE=1
      MC=2
      NOX=MAX/2
 7100 MDATE=MDATE+1
      IF(MDATE.GE.MAX) GO TO 6530
      IF(MVAR(MDATE,2).EQ.0) GO TO 7100
      IF(MVAR(MDATE,2).LT.0) GO TO 7150
      IF(MDATE.LT.(NOX+5)) GO TO 7100
      GO TO 8000
 7150 IF(MDATE.GT.(NOX-5)) GO TO 7100
C
C
C
C          PEAK IS SEEN BEFORE HALF OF THE PROJECT
C
C
C
C
      KK=0
      MCT=1
 7200 KK=KK+1
      MS=MDATE-KK
      IF(MS.LE.0) GO TO 7250
      IF(MVAR(MS,2).EQ.0) GO TO 7200
      IF(MVAR(MS,2).LT.0) GO TO 7100
 7250 DO 7350 I1=1,NOI
      IF(IAF(I1,50).NE.MS) GO TO 7350
      IF(MVAR(MS,2).LT.(-MVAR(MDATE,2))) GO TO 7300
      IF(IAF(I1,4).GT.MVAR(MS,2)) GO TO 7350
      IF(IAF(I1,4).LT.(-MVAR(MDATE,2))) GO TO 7350
      GO TO 7380
C
C
C
C
C
```

```
C
 7300 IF(IAF(I1,4).GT.(-MVAR(MDATE,2))) GO TO 7350
      IF(IAF(I4,4).LT.MVAR(MS,2)) GO TO 7350
      GO TO 7380
 7350 CONTINUE
      GO TO 7100
 7380 LL=IAF(I1,8)-IAF(I1,3)-IAF(I1,50)
C
C
C
C
C
C
C
C
      IF(LL.LE.KK) GO TO 7400
      MM=KK
      GO TO 7450
 7400 MM=LL
 7450 DO 7500 I=1,NOI
      IAF(I,51)=0
 7500 CONTINUE
      IAF(I1,51)=IAF(I1,50)+MM
 7550 DO 7600 I2=1,NOI
      IF(IAF(I2,51).EQ.0) GO TO 7600
      DO 7600 I3=1,NOI
      IF(IAF(I3,1).NE.IAF(I2,2)) GO TO 7600
      MM=IAF(I2,51)+IAF(I2,3)
      PRINT 7580,IAF(I3,51),MM,I3
 7580 FORMAT(1H ,3I10)
      IF(IAF(I3,50).GE.MM) GO TO 7600
      IF(IAF(I3,51).GE.MM) GO TO 7600
      IAF(I3,51)=MM
 7600 CONTINUE
      DO 7650 I2=1,NOI
      IF(IAF(I2,51).NE.0) GO TO 7650
      IAF(I2,51)=IAF(I2,50)
 7650 CONTINUE
 7660 DO 7700 I2=1,200
      MVAR(I2,3)=0
 7700 CONTINUE
      DO 7750 I2=1,NOI
      MM=IAF(I2,51)+IAF(I2,3)
      MVAR(IAF(I2,51),3)=MVAR(IAF(I2,51),3)+IAF(I2,4)
      MVAR(MM,3)=MVAR(MM,3)-IAF(I2,4)
 7750 CONTINUE
      KVAR=0
      DO 7800 I2=1,200
```

```
      KVAR=KVAR+MVAR(I2,3)**2
 7800 CONTINUE
      PRINT 7850,KVAR,MINV
      PRINT 7900,(IAF(I1,50),I1=1,NOI)
      PRINT 7900,(IAF(I1,51),I1=1,NOI)
 7850 FORMAT(1H ,2I10)
 7900 FORMAT(1H ,40I3)
      IF(KVAR.LT.MINV) GO TO 7920
      IF(MCT.EQ.1) GO TO 7350
      GO TO 8200
 7920 DO 7950 I=1,200
      MVAR(I,2)=MVAR(I,3)
 7950 CONTINUE
      MINV=KVAR
      DO 7980 I=1,NOI
      IAF(I,50)=IAF(I,51)
 7980 CONTINUE
      GO TO  7050
C
C
C
C           PEAK IS SEEN AFTER THE HALF OF THE PROJECT
C
C
C
 8000 KK=0
      MCT=2
 8050 KK=KK+1
      MS=MDATE+KK
      IF(MS.GE.MAX) GO TO 8100
      IF(MVAR(MS,2).EQ.0) GO TO 8050
      IF(MVAR(MS,2).GT.0) GO TO 7100
 8100 DO 8200 I1=1,NOI
      MED=IAF(I1,50)+IAF(I1,3)
      IF(MED.NE.MS) GO TO 8200
      IF((-MVAR(MS,2)).LT.MVAR(MDATE,2)) GO TO 8150
      IF(IAF(I1,4).GT.(-MVAR(MS,2))) GO TO 8200
      IF(IAF(I1,4).LT.MVAR(MDATE,2)) GO TO 8200
      GO TO 8250
 8150 IF(IAF(I1,4).GT.MVAR(MDATE,2)) GO TO 8200
      IF(IAF(I1,4).LT.(-MVAR(MS,2))) GO TO 8200
      GO TO 8250
 8200 CONTINUE
      GO TO 7100
 8250 MFU=IAF(I1,50)-IAF(I1,5)
      IF(MFU.GT.KK) GO TO 8300
      MM=MFU
      GO TO 8350
```

```
8300 MM=KK
8350 DO 8400 I=1,NOI
     IAF(I,51)=0
8400 CONTINUE
     IAF(I1,51)=IAF(I1,50)-MM
8450 DO 8500 I2=1,NOI
     IF(IAF(I2,51).EQ.0) GO TO 8500
     DO 8500 I3=1,NOI
     IF(IAF(I3,2).NE.IAF(I2,1)) GO TO 8500
     MM=IAF(I3,50)+IAF(I3,3)
     IF(IAF(I2,51).GE.MM) GO TO 8500
     IAF(I3,51)=IAF(I2,51)-IAF(I3,3)
     PRINT 8555,IAF(17,51)
8555 FORMAT(1H ,10X,I5)
8500 CONTINUE
     DO 8550 I2=1,NOI
     IF(IAF(I2,51).NE.0) GO TO 8550
     IAF(I2,51)=IAF(I2,50)
8550 CONTINUE
     PRINT 8556,IAF(17,51)
8556 FORMAT(1H ,20X,I5)
     GO TO 7660
9980 FORMAT(1H1)
9990 PRINT 9993,MERR
9993 FORMAT(1H ,18HERR MESSAGE NO---,I5)
9999 NOP=1
     END
```

Input Format

		Column
1.	Preceding Node	1 - 5
2.	Succeeding Node	6 - 10
3.	Duration	11 - 15
4.	Resource Requirement	16 - 20
5.	Early Start Date and Late Finish Date for Preceding Node	21 - 25
6.	Early Start Date and Late Finish Date for Succeeding Node	26 - 30

INPUT DECK

```
// ....... ( job card ) ........
//s1 EXEC FORTGCLG,REGION=120K
//FORT.SYSIN DD *
        .
        .
        .
  (FORTRAN IV SOURCE PROGRAMME)
        .
        .
        .
//GO.SYSIN DD DUMMY
//GO.FT03F001   DD *
        .
        .
        .
  (INPUT DATA)
        .
        .
        .
99
/*
//
```