RESOURCE LEVELING IN CONSTRUCTION BY OPTIMIZATION

By Said M. Easa,[1] Member, ASCE

ABSTRACT: Resource leveling is needed in construction to avoid the difficulties associated with the large variations in resource usage. This paper presents an integer-linear optimization model of resource leveling (single resource, continuous activities) which guarantees the optimal leveling. The objective function of the model minimizes the absolute deviations between the resource requirements and a uniform resource level, between consecutive resource requirements, or between the resource requirements and desirable nonuniform resource levels. The model requires as input the critical path method (CPM) scheduling results, from which the constraints and objective function of the model are established automatically by an interface program. Extensions of the model to multiple resources and trade-off of cost scheduling are suggested. The model is applicable to activity-on-arrow, activity-on-node, and precedence networks and is intended for small- to medium-sized construction projects.

INTRODUCTION

There are two basic categories of scheduling resources: fixed-limits resource scheduling and resource leveling (Antill and Woodhead 1982; Moder et al. 1983). The fixed-limits resource scheduling arises when there are definite limitations on the amount of resources available. The scheduling objective is to extend the project duration as little as possible beyond the original critical path length such that the resource constraints are met. In this process, both critical and noncritical activities are shifted. Many heuristic and optimization methods have been developed for fixed-limits resource scheduling and can be found in the preceding references.

The resource leveling problem arises when there are sufficient resources available and it is necessary to reduce the fluctuations in the pattern of resource usage. These fluctuations are very undesirable because they often present labor, utilization, and financial difficulties for the contractor. The scheduling objective is to make the resource requirements as uniform as possible, or in some cases to make them match desirable nonuniform resource levels (Adrian 1973; Harris 1978). In resource leveling, there are no resource limits and the process is accomplished by shifting only the noncritical activities within their available floats; the project duration of the original critical path remains fixed.

Numerous analytical methods have been developed for resource leveling, almost all of them are heuristic in nature. Examples of these heuristic methods are the work by Burges and Killebrew (1962), Galbreath (1965), Shaffer et al. (1965), Harris (1978), and Antill and Woodhead (1982). The basic idea of these methods is to shift the noncritical activities in a systematic order according to some rules-of-thumb. The heuristic methods can handle

[1]Prof., Dept. of Civ. Engrg., Lakehead Univ., Thunder Bay, Ontario, Canada P7B 5E1.

Note. Discussion open until November 1, 1989. To extend the closing date one month, a written request must be filed with the ASCE Manager of Journals. The manuscript for this paper was submitted for review and possible publication on June 15, 1987. This paper is part of the ***Journal of Construction Engineering and Management***, Vol. 115, No. 2, June, 1989. ©ASCE, ISSN 0733-9364/89/0002-0302/ $1.00 + $.15 per page. Paper No. 23582.

very large projects, but the solution they provide is not necessarily the optimum.

Very little work has been done to solve the resource leveling problem by optimization (Phillips and Garcia-Diaz 1981; Antill and Woodhead 1982; Moder et al. 1983). A model that uses the complete enumeration of all possible solutions has been presented by Ahuja (1976). The model minimizes the period-by-period resource variations by enumerating all possible combinations of activities within their available floats. However, as the number of noncritical activities becomes large, the required number of combinations would be impractical. For example, if a project has 10 noncritical activities, each having a float of 6 time units, there will be 6^{10} (over 60 million) possible combinations.

With the development of efficient algorithms that can handle relatively large integer optimization problems, an optimization model of resource leveling can be formulated and solved efficiently. Recently, Karaa and Nasr (1986) developed an optimization model that minimizes the cost of leasing additional resources in construction. The purpose of this paper is to present an integer-linear optimization model of resource leveling which is intended for small to medium-sized projects.

Resource Leveling Model

Model Formulation

Suppose that the CPM scheduling of the critical and noncritical activities, has been established for an activity-on-arrow network (extensions to other network types are addressed later). Each noncritical activity k has duration D_k, free float FF_k, total float TF_k, interfering float IF_k, earliest start time ES_k, earliest finish time EF_k, and required resource rate (e.g., units/day) R_k; $k = 1, 2, \ldots, n$ where n is the number of noncritical activities. The duration, earliest start time, and resource rate of each critical activity are also known, and let these be denoted by T_l, E_l, and S_l, respectively; $l = 1, 2, \ldots, m$ where m is the number of critical activities. The total duration of the project, T, is given by

$$T = \sum_{l=1}^{m'} T_l \quad \ldots\ldots (1)$$

where m' = number of critical activities along a critical path. If the project has multiple critical paths, then $m' < m$ and if it has a single critical path then $m' = m$.

The previous variables are all known and represent the input to the model. A schematic representation of these variables for a general noncritical activity k and another noncritical activity q which immediately follows activity k are shown in Fig. 1 (there may be more than one following activity q). The critical activities performed during the available time of activity k are also shown in the figure. The model will now be formulated by considering the general noncritical activity k.

Activity Shifting

Let the amount of shifting (the decision variable) of activity k be denoted by x_k. The variable x_k is integer and can be represented by

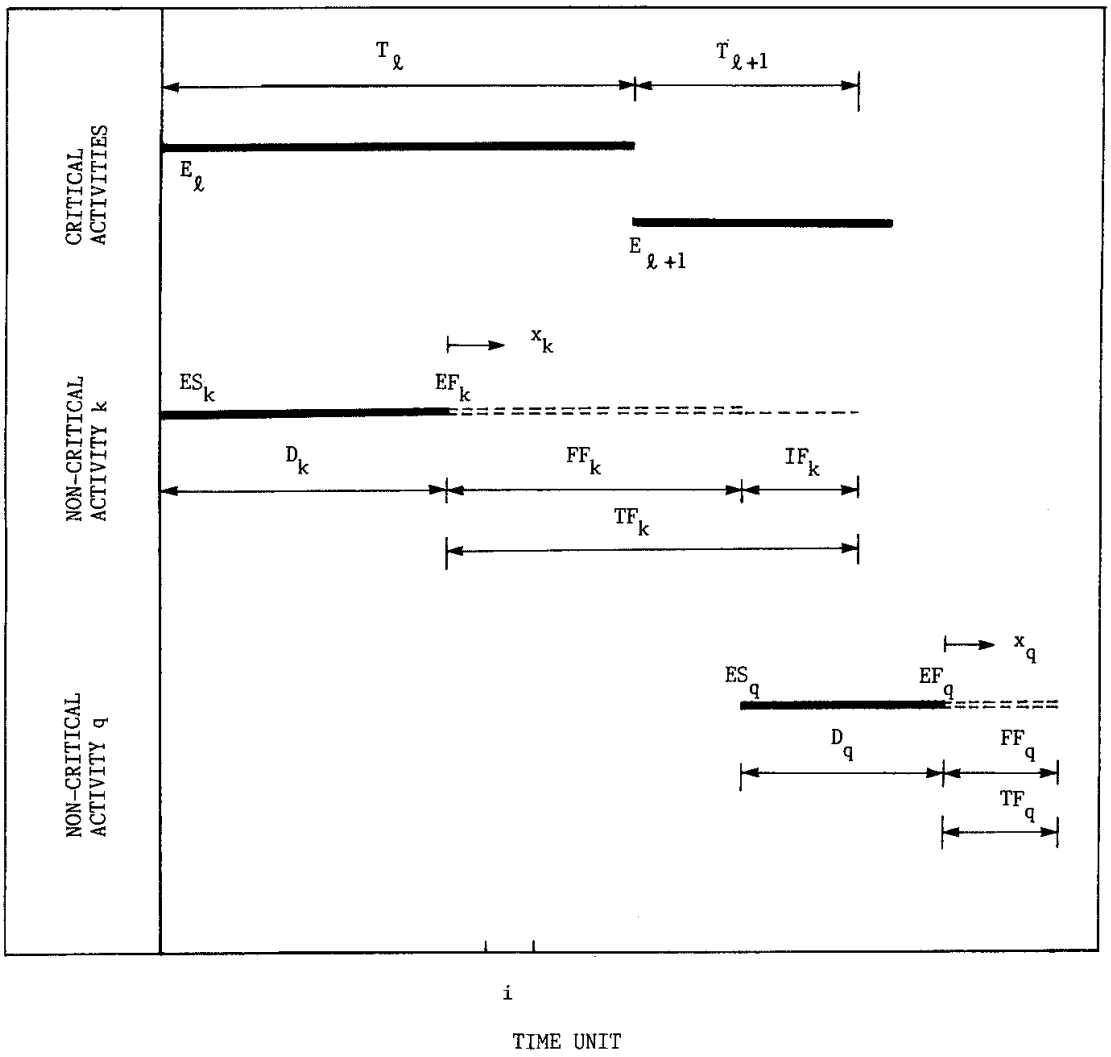

FIG. 1. Schematic Representation of General Noncritical Activity k and Following Noncritical Activity q

$$x_k = \sum_{j=1}^{TF_k} j\lambda_{kj}; \qquad k = 1, 2, \ldots, n \ldots\ldots\ldots\

This is because the amount of shifting of activity k beyond its free float (i.e., during its interfering float) must be equal to or less than the amount of shifting of each of the activities q beyond its earliest start time. This can be represented mathematically by

For AON use Lag_{kq} instead of FF_k

$$(x_k - FF_k) \leq x_q; \qquad k = 1, 2, \ldots, n \quad \text{for all } q \in Q_k \ldots\ldots\ldots\ldots\ldots\ldots (4)$$

When $x_k \leq FF_k$, Eq. 4 is automatically satisfied since $x_q \geq 0$. When $x_k > FF_k$, Eq. 4 ensures that the shifting of activity k beyond the free float $(x_k - FF_k)$ is equal to or less than the shifting of activity q. Note that this constraint is not needed when the interfering float of activity k equals zero.

Project Resource Rate

Let the resource rate of the activities performed in time unit i be denoted by y_i; this will be referred to as project resource rate. The variable y_i consists of the resource rate of the critical activity (or activities, in case of multiple critical paths), y_{ci}, and those of the noncritical activities. Thus

T = project duration

$$y_i = y_{ci} + \sum_{p=1}^{n_i} y_{pi}; \qquad i = 1, 2, \ldots, T \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (5)$$

in which n_i is the number of noncritical activities that have their durations, free floats, or interfering floats overlapping with time unit i. The variable y_{ci} equals the resource rate of the critical activity (or activities) which is being performed in time unit i. For example, for the situation of Fig. 1 one has

Notation: $E_l := ESD_l$

Correction. For Start activity, $E_l = ESD_l = 0$, so i must be 1 (not 0)

$$y_{ci} = S_l; \qquad E_l \leq i < E_{l+1} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (6)$$

The second term in Eq. 5 is the sum of the resource rates of all the noncritical activities that could be performed in time unit i. For activity k, the resource rate in time unit i can be calculated as follows

Correction.

$$y_{ki} = \left(1 - \sum_{j=1}^{TF_k} \lambda_{kj}\right) R_k; \qquad ES_k \leq i < EF_k \ldots\ldots\ldots\ldots\ldots\ldots\ldots (7a)$$

Correction.

$$y_{ki} = \lambda_{kj} R_k; \qquad ES_k + j \leq i < EF_k + j, \qquad j = 1, 2, \ldots, TF_k \ldots\ldots\ldots (7b)$$

$$y_{ki} = 0, \text{ otherwise} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (7c)$$

As noted from Eq. 7*a*, when $\Sigma\lambda_{kj} = 0$ (i.e., $x_k = 0$ from Eq. 2) the resource rate R_k will be considered for each time unit within ES_k and EF_k. As also noted from Eq. 7*b*, when any $\lambda_{kj} = 1$ (i.e. $x_k = j$ from Eqs. 2 and 3) the resource rate R_k will be considered for each time unit in which activity k is performed in its shifted position.

Objective Function

An objective function for performing resource leveling is to minimize the deviation between the resource requirements y_i of Eq. 5 and the desirable resource rates c_i. The values of c_i would be constant if the objective is to establish a uniform resource level (in this case, c_i equals the total resource rates required divided by T, rounded to nearest integer). The values of c_i would be different if the objective is to match desirable non-uniform resource levels. To simplify the solution of the model, the absolute deviation between

y_i and c_i is used as the criterion for minimization. Thus, the objective function takes the following form

$$\text{Minimize } z = \sum_{i=1}^{T} (e_i^- + e_i^+) \quad \ldots\ldots (8)$$

$$\text{Subject to } y_i - c_i + e_i^- - e_i^+ = 0; \qquad i = 1, 2, \ldots, T \quad \ldots\ldots (9)$$

$$e_i^-, e_i^+ \geq 0; \qquad i = 1, 2, \ldots, T \quad \ldots\ldots (10)$$

in which T = duration of the project (given by Eq. 1), e_i^-, e_i^+ = additional variables. For any y_i, either e_i^+ (if $y_i > c_i$) or e_i^- (if $y_i < c_i$) takes a positive value and the other one will be zero.

A summary of the complete resource leveling model is given here

$$\text{Minimize } z = \sum_{i=1}^{T} (e_i^- + e_i^+) \quad \ldots\ldots (11)$$

$$\text{Subject to } y_i - c_i + e_i^- - e_i^+ = 0; \qquad i = 1, 2, \ldots, T \quad \ldots\ldots (12)$$

$$y_i = y_{ci} + \sum_{p=1}^{n_i} y_{pi}; \qquad i = 1, 2, \ldots, T \quad \ldots\ldots (13)$$

$$x_k = \sum_{j=1}^{TF_k} j\lambda_{kj}; \qquad k = 1, 2, \ldots, n \quad \ldots\ldots (14)$$

$$\sum_{j=1}^{TF_k} \lambda_{kj} \leq 1; \qquad k = 1, 2, \ldots, n \quad \ldots\ldots (15)$$

For AON use Lag_{kq} instead of FF_k

$$(x_k - FF_k) \leq x_q; \qquad k = 1, 2, \ldots, n \text{ for all } q \in Q_k \quad \ldots\ldots (16)$$

$$\lambda_{kj} \in \{0, 1\}; \qquad k = 1, 2, \ldots, n, \quad j = 1, 2, \ldots, TF_k \quad \ldots\ldots (17)$$

$$e_i^-, e_i^+ \geq 0; \qquad i = 1, 2, \ldots, T \quad \ldots\ldots (18)$$

in which y_{ci} and y_{pi} are calculated by Eqs. 6 and 7, respectively (note that Eq. 7 uses the subscript k to illustrate activity k, but it should be used for all activities p which are performed in time unit i).

The formulation of Eqs. 11–18 represents an integer-linear optimization model that can be solved directly by a number of the existing computer software programs (e.g., Schrage 1981). After substituting for the variables y_i and x_k of the equality constraints of Eqs. 13 and 14 into Eqs. 12 and 16, respectively, the model will have $\Sigma_{k=1}^{n} TF_k$ decision (binary) variables, at most $(T + n + \Sigma_{k=1}^{n} n_k)$ constraints, and $2T$ nonnegativity restrictions; where n_k is the number of activities immediately following activity k. Clearly, the number of constraints could be less than the above quantity because the constraint of Eq. 16 is not needed for the noncritical activities that have no interfering floats.

Improved Criterion for Uniform Leveling

For uniform leveling, the criterion of minimizing the absolute deviation from a uniform resource level may result in project resource rates close to that level, but which vary frequently during the project duration. To reduce

these frequent variations, another criterion for minimizing the absolute deviations between the project resource rates of consecutive time units can be used. In this case, the model of Eqs. 11–18 will remain the same except that the objective function of Eq. 11 and the constraints of Eqs. 12 and 18 are modified to

$$\text{Minimize } z = \sum_{i=1}^{T-1} (u_i^- + u_i^+) \quad \ldots\ldots (19)$$

$$\text{Subject to } y_{i+1} - y_i + u_i^- - u_i^+ = 0; \qquad i = 1, 2, \ldots, (T-1) \quad \ldots\ldots (20)$$

$$u_i^-, u_i^+ \geq 0; \qquad i = 1, 2, \ldots, (T-1) \quad \ldots\ldots (21)$$

in which u_i^- and u_i^+ are additional variables defined with respect to y_{i+1} and y_i in a similar manner as e_i^- and e_i^+ of Eq. 9.

Extensions

The presented model is applicable to activity-on-node and precedence networks, after modifying the network logic constraint of Eq. 16. For activity-on-node networks, this constraint is modified to

I.e., for AON use Lag_{kq} instead of FF_k

$$[x_k - (ES_q - EF_k)] \leq x_q; \qquad k = 1, 2, \ldots, n \text{ for all } q \in Q_k \quad \ldots\ldots (22)$$

For precedence networks (which involve start-to-start, start-to-finish, finish-to-start, and finish-to-finish relationships) the network logic can be preserved by considering the lag times between activities. These lag times are output of the CPM scheduling computations. Let L_{kq} represent the lag time between activities k and q. Then, the modified network logic constraint of Eq. 16 becomes

I.e., for overlapping networks use Lag_{kq} instead of FF_k

$$(x_k - L_{kq}) \leq x_q; \qquad k = 1, 2, \ldots, n \text{ for all } q \in Q_k \quad \ldots\ldots (23)$$

If the relationship between k and q is compound (start-to-start and finish-to-finish), there will be two values of the lag time and the smaller value must be used in Eq. 23.

The model allows only a single resource to leveled. However, this is not a restriction because normally when one resource is leveled other resources tend to be leveled as well (Harris 1978). In addition, the model can be applied directly to multiple resources if the prime concern in the construction project is to level a common base (e.g., cash flow) in which each resource can be expressed. For other cases in which it is required to level distinct resources directly, the model can be extended to accommodate such cases. The objective function of the model is set to minimize the sum of the total deviations of each resource, with a weight associated with each resource.

Implementation Aspects

The model requires as input the CPM scheduling results and the resource rates of activities. This output information is used by an interface program which establishes automatically a file containing the objective function and constraints of the model. This file is then input to an integer-linear optimization program (LINDO) which solves the model (Schrage 1981, 1984). The interface and optimization programs can be chained to existing CPM scheduling packages. In this way, resource leveling can be performed with little extra effort placed on the user.

As is the case with integer optimization models, the presented model has a limitation. For large projects, the number of binary variables and constraints may be too large to be handled by existing computer packages. Of course, this number depends on project duration, the number of non-critical activities, and their float and lag times. However, recent optimization packages can handle very large problems; for example LINDO can solve problems with up to 4,500 variables and 800 constraints (Schrage 1984). Some ideas to overcome the dimensionality problem of optimization models are presented by Karaa and Nasr (1986).

APPLICATION

Example A: Activity-on-Arrow Network

Consider the simple example of Fig. 2 which is an activity-on-arrow network for a project consisting of five activities. Activity A must precede activities B and D, while activities C and D must precede activity E. The duration (in days) and the resource requirements (in units/day) for each activity are shown in the figure. Following the conventional forward and backward-pass computations of CPM, the early start scheduling results are obtained as shown in Fig. 3. These CPM scheduling results represent the input data to the model. The histogram of resource requirements corresponding to the early start schedule is shown in Fig. 4. The model will now be used to level that histogram to uniform and desirable non-uniform histograms.

Case 1: Minimizing Deviations for a Uniform Histogram

Since $T = 5$, the objective function of Eq. 11 is

$$\text{Minimize } z = e_1^- + e_1^+ + e_2^- + e_2^+ + e_3^- + e_3^+ + e_4^- + e_4^+ + e_5^- + e_5^+ \quad \ldots\ldots (24)$$

For a uniform histogram, c_i is a constant equal to the total resource rates divided by $5 = 46/5 = 9.2$, say 9. The constraints of Eqs. 12–18 become

$$y_1 - 9 + e_1^- - e_1^+ = 0 \quad \ldots\ldots (25)$$

$$y_2 - 9 + e_2^- - e_2^+ = 0 \quad \ldots\ldots (26)$$

$$y_3 - 9 + e_3^- - e_3^+ = 0 \quad \ldots\ldots (27)$$

$$y_4 - 9 + e_4^- - e_4^+ = 0 \quad \ldots\ldots (28)$$

$$y_5 - 9 + e_5^- - e_5^+ = 0 \quad \ldots\ldots (29)$$

$$y_1 = 14 - 4\lambda_{C1} - 4\lambda_{C2} - 4\lambda_{C3} \quad \ldots\ldots (30)$$

$$y_2 = 10 + 4\lambda_{C1} \quad \ldots\ldots (31)$$

$$y_3 = 8 + 4\lambda_{C2} - 2\lambda_{D1} \quad \ldots\ldots (32)$$

$$y_4 = 8 + 4\lambda_{C3} + 2\lambda_{D1} - 2\lambda_{E1} \quad \ldots\ldots (33)$$

$$y_5 = 6 + 2\lambda_{E1} \quad \ldots\ldots (34)$$

$$x_C = \lambda_{C1} + 2\lambda_{C2} + 3\lambda_{C3} \quad \ldots\ldots (35)$$

$$x_D = \lambda_{D1} \quad \ldots\ldots (36)$$

$$x_E = \lambda_{E1} \quad \ldots\ldots (37)$$

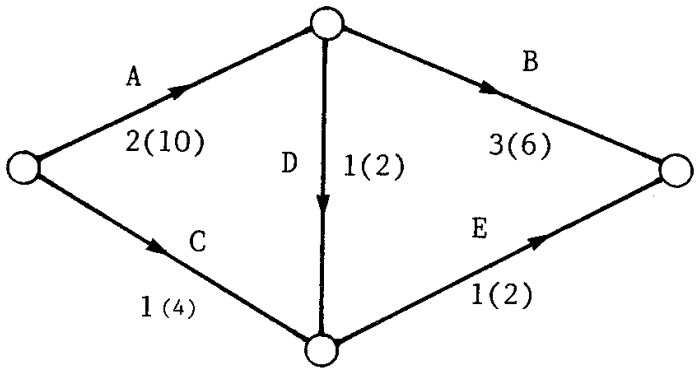

FIG. 2. Arrow Diagram for Model Application (Example A)

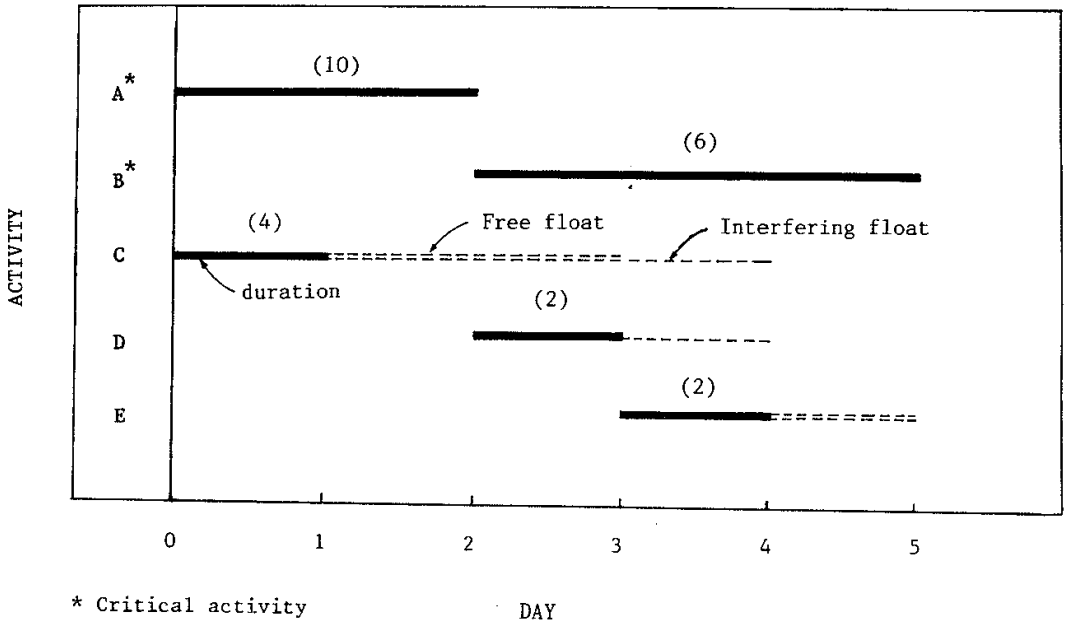

FIG. 3. CPM Scheduling Results (Early Start Schedule)

$$\lambda_{C1} + \lambda_{C2} + \lambda_{C3} \leq 1 \quad \ldots\ldots (38)$$

$$\lambda_{D1} \leq 1 \quad \ldots\ldots (39)$$

$$\lambda_{E1} \leq 1 \quad \ldots\ldots (40)$$

$$x_C - 2 \leq x_E \quad \ldots\ldots (41)$$

$$x_D \leq x_E \quad \ldots\ldots (42)$$

$$\lambda_{C1}, \lambda_{C2}, \lambda_{C3}, \lambda_{D1}, \lambda_{E1} \in \{0, 1\} \quad \ldots\ldots (43)$$

$$e_i^-, e_i^+ \geq 0; \qquad i = 1, 2, \ldots, 5 \quad \ldots\ldots (44)$$

Substituting for the variables of the equality constraints of Eqs. 30–34

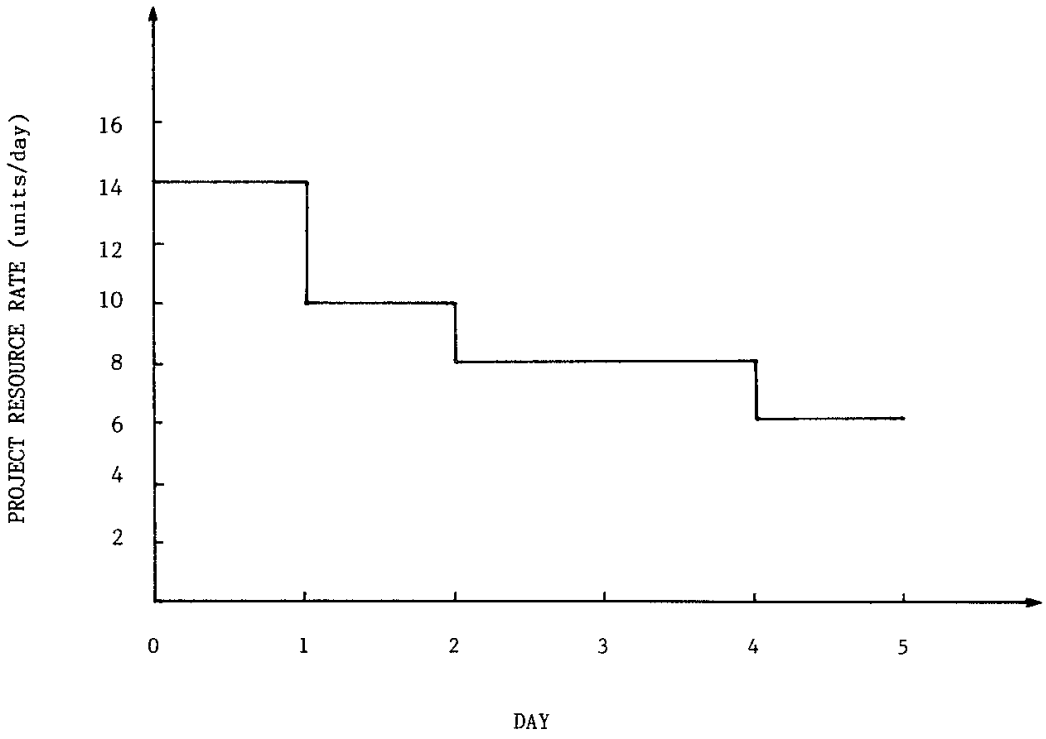

FIG. 4. Histogram of Required Resources for CPM Early Start Schedule

and Eqs. 35–37 into Eqs. 25–29 and Eqs. 41–42, respectively, the reduced model becomes

$$\text{Minimize } z = e_1^- + e_1^+ + e_2^- + e_2^+ + e_3^- + e_3^+ + e_4^- + e_4^+ + e_5^- + e_5^+ \quad \ldots\ (45)$$

$$\text{Subject to } 5 - 4\lambda_{C1} - 4\lambda_{C2} - 4\lambda_{C3} + e_1^- - e_1^+ = 0 \quad \ldots\ (46)$$

$$1 + 4\lambda_{C1} + e_2^- - e_2^+ = 0 \quad \ldots\ (47)$$

$$-1 + 4\lambda_{C2} - 2\lambda_{D1} + e_3^- - e_3^+ = 0 \quad \ldots\ (48)$$

$$-1 + 4\lambda_{C3} + 2\lambda_{D1} - 2\lambda_{E1} + e_4^- - e_4^+ = 0 \quad \ldots\ (49)$$

$$-3 + 2\lambda_{E1} + e_5^- - e_5^+ = 0 \quad \ldots\ (50)$$

$$\lambda_{C1} + \lambda_{C2} + \lambda_{C3} \leq 1 \quad \ldots\ (51)$$

$$\lambda_{C1} + 2\lambda_{C2} + 3\lambda_{C3} - 2 \leq \lambda_{E1} \quad \ldots\ (52)$$

$$\lambda_{D1} \leq \lambda_{E1} \quad \ldots\ (53)$$

$$\lambda_{C1}, \lambda_{C2}, \lambda_{C3}, \lambda_{D1}, \lambda_{E1} \in \{0, 1\} \quad \ldots\ (54)$$

$$e_i^-, e_i^+ \geq 0; \qquad i = 1, 2, \ldots, 5 \quad \ldots\ (55)$$

The previous problem has two optimum solutions with $z = 5$. The first solution is $\lambda_{C1} = 0$, $\lambda_{C2} = 0$, $\lambda_{C3} = 1$, $\lambda_{D1} = 0$, and $\lambda_{E1} = 1$. Thus from Eq. 2, $x_C = 3$, $x_D = 0$, and $x_E = 1$. The second solution is $\lambda_{C1} = 0$, $\lambda_{C2} = 1$, $\lambda_{C3} = 0$, $\lambda_{D1} = 1$, and $\lambda_{E1} = 1$. Thus, $x_C = 2$, $x_D = 1$, and $x_E = 1$.

The resource histograms for the two solutions are shown in Fig. 5. Al-

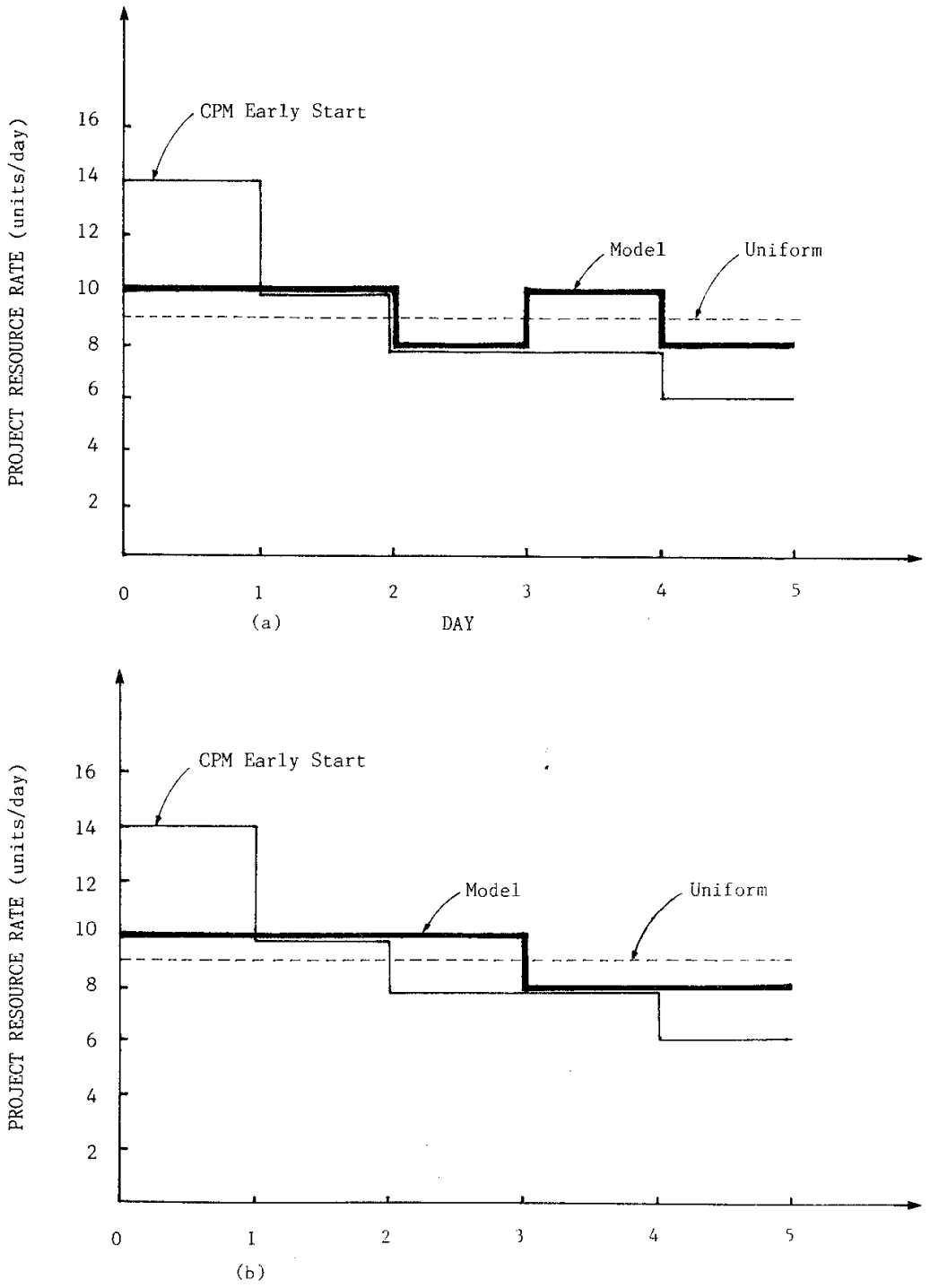

FIG. 5. Resource Leveling for: (*a*) First Optimum Solution for Uniform Histogram (Case 1); and (*b*) Second Optimum Solution for Uniform Histogram (Case 1) and Optimum Solution for Minimizing Daily Resource Variations (Case 2)

though the two histograms have the same value of z (i.e., the same total absolute deviations from the uniform value of 9) the second optimum solution is better than the first, because the former has less variations between the project resource rates of consecutive days. This observation has led to the consideration of the other criterion of minimizing the deviations between daily resource rates.

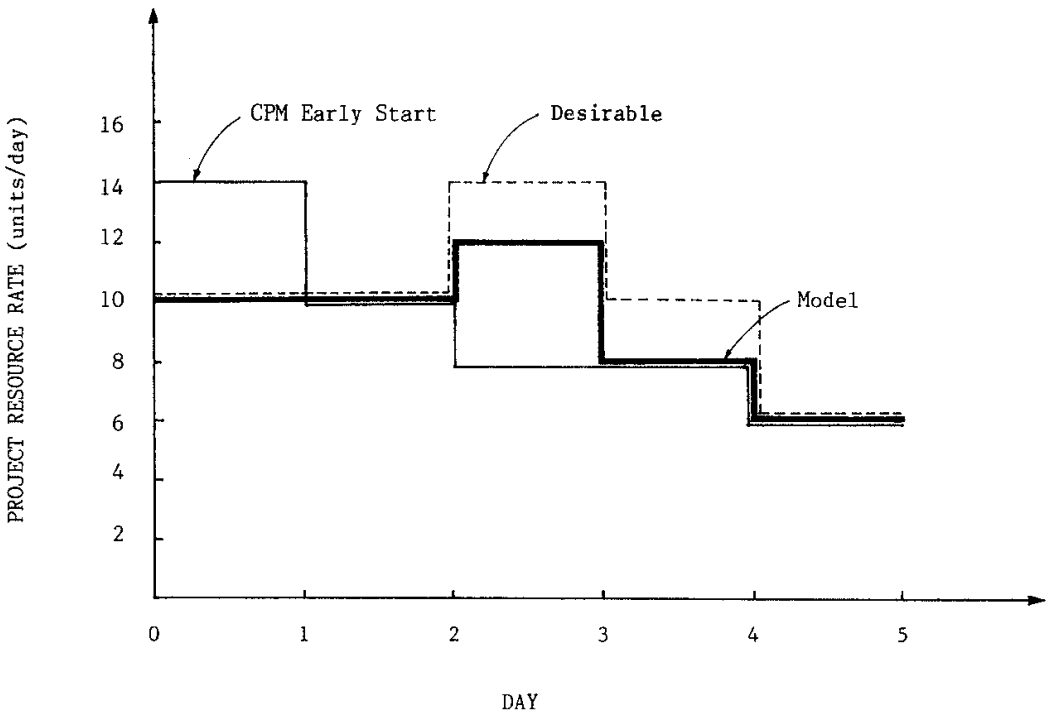

FIG. 6. Resource Leveling for Desirable Nonuniform Histogram (Case 3)

Case 2: Minimizing Daily Resource Variations

The uniform leveling is also accomplished using the criterion of minimizing the absolute deviations between the project resource rates of consecutive days. Substituting for the variables in Eqs. 19–21 and the additional constraints of Eqs. 13–17, one obtains

$$\text{Minimize } z = u_1^- + u_1^+ + u_2^- + u_2^+ + u_3^- + u_3^+ + u_4^- + u_4^+ \quad (56)$$

$$\text{Subject to } -4 + 8\lambda_{C1} + 4\lambda_{C2} + 4\lambda_{C3} + u_1^- - u_1^+ = 0 \quad (57)$$

$$-2 - 2\lambda_{C1} + 4\lambda_{C2} - 2\lambda_{D1} + u_2^- - u_2^+ = 0 \quad (58)$$

$$-4\lambda_{C2} + 4\lambda_{C3} + 4\lambda_{D1} - 2\lambda_{E1} + u_3^- - u_3^+ = 0 \quad (59)$$

$$-2 - 4\lambda_{C3} - 2\lambda_{D1} + 4\lambda_{E1} + u_4^- - u_4^+ = 0 \quad (60)$$

$$\lambda_{C1} + \lambda_{C2} + \lambda_{C3} \leq 1 \quad (61)$$

$$\lambda_{C1} + 2\lambda_{C2} + 3\lambda_{C3} - 2 \leq \lambda_{E1} \quad (62)$$

$$\lambda_{D1} \leq \lambda_{E1} \quad (63)$$

$$\lambda_{C1}, \lambda_{C2}, \lambda_{C3}, \lambda_{D1}, \lambda_{E1} \in \{0, 1\} \quad (64)$$

$$u_i^-, u_i^+ \geq 0; i = 1, 2, 3, 4 \quad (65)$$

The solution of this model happens to be the second optimum solution of Case 1 shown in Fig. 5(*b*), with an objective function value $z = 2$.

Case 3: Minimizing Deviations for a Nonuniform Histogram

Suppose it is required to accomplish resource leveling so that the project resource rates are as close as possible to the histogram shown by dotted lines

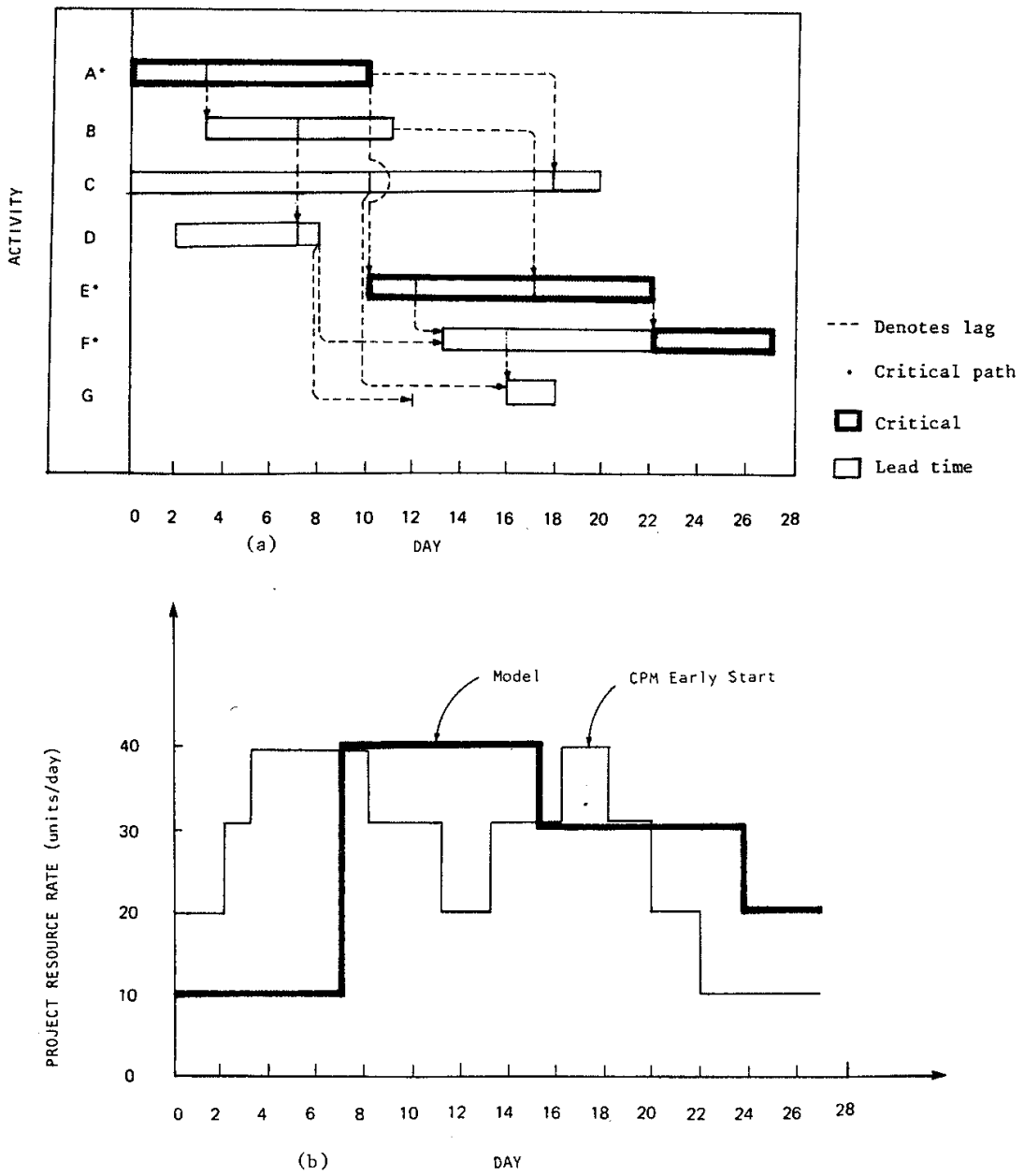

FIG. 7. Example Application to Precedence Network: (*a*) CPM Early Start Schedule; (*b*) Optimum and CPM Early Start Resource Histograms

in Fig. 6. The desirable levels of consecutive days are 10, 10, 14, 10, 6, respectively. In this case, the value 9 in Eqs. 25–29 is replaced by the previous values. The model to be solved in this case is the same as Eqs. 45–55 except that the two terms -1 and -3 in Eqs. 48 and 50 are replaced by -6 and 0, respectively. The solution of the model is $\lambda_{C1} = 0$, $\lambda_{C2} = 1$, $\lambda_{C3} = 0$, $\lambda_{D1} = 0$, $\lambda_{E1} = 0$. Thus from Eq. 2, $x_C = 2$, $x_D = 0$, and $x_E = 0$ and the objective function value $z = 4$. The respective histogram of resource rates is shown by solid-thick lines in Fig. 6.

Example B: Precedence Network

Consider the precedence network with the CPM early-start scheduling results shown in Fig. 7(*a*) (Harris 1978). The network has three critical and

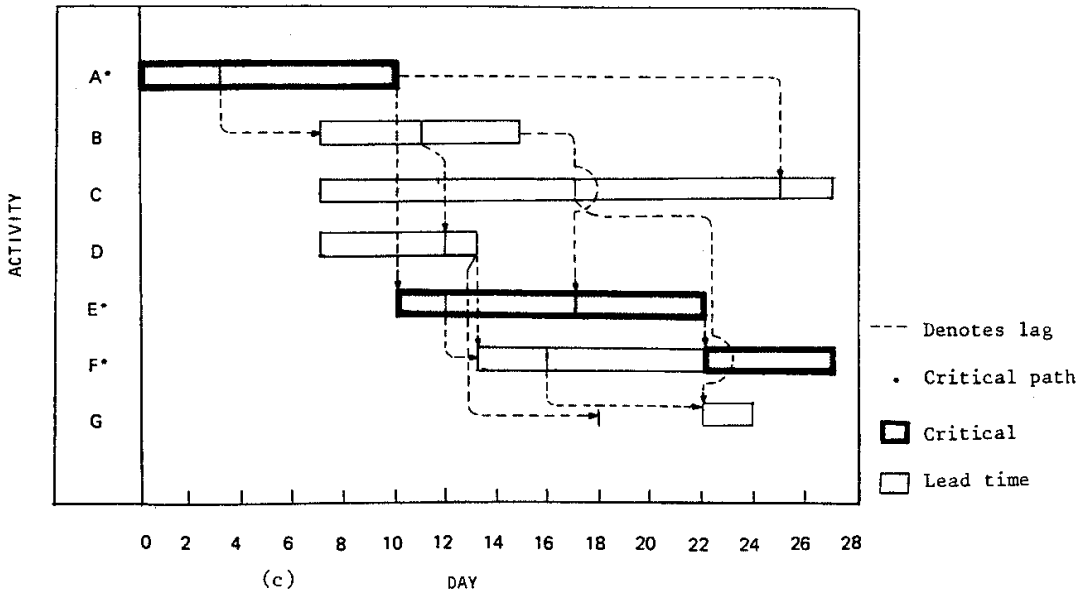

FIG. 7. Example Application to Precedence Network: (*c*) Optimum Schedule

TABLE 1. Total Float and Lag Times for Precedence Network (Example B)

Activity (1)	Duration (days) (2)	Total float (days) (3)	Lag time (days) (4)	
A[a]	10	0	$A - B$ 0 (4)	$A - C$ 8[b] (15)
B	8	5	$B - D$ 0 (1)	$B - E$ 6 (2)
C	20	7	$C - G$ 6 (5)	
D	6	5	$D - F$ 5 (0)	$D - G$ 4 (5)
E[a]	12	0	$E - F$ 0 (0)	
F[a]	14	0	$F - G$ 0 (6)	
G	2	9	—	

[a]Critical activity.

[b]The lag time between activities A and C for CPM early start schedule is 8; lag time between activities A and C for optimum schedule is (15).

four noncritical activities. The total float of activities and the lag times between them are shown in Table 1. The relationship between activities E and F is compound with lag times 1 and 0, respectively, and the smaller value is used. Suppose the resource rates of all activities are equal (10 units/day). The CPM early-start resource histogram is shown in Fig. 7(*b*). It is required to make the resource use uniform by minimizing the daily variations.

The model with the objective function of Eqs. 19–21 and the modified constraint of Eq. 23 is applied to this precedence network. The optimum resource histogram is shown in Fig. 7(*c*). As noted, the lag times for this schedule (shown also in parentheses in Table 1) are positive or zero, implying that the precedence relationships are maintained.

Summary and Conclusions

This paper presents an optimization model of resource leveling that can be used to establish the resource usage that matches as close as possible a uniform or nonuniform histogram. The model objective is to minimize the total absolute deviations between actual and desirable resource rates. The model has the following features: (a) it guarantees the optimum leveling; (b) it is applicable to activity-on-arrow, activity-on-node, and precedence networks with single or multiple critical paths; (c) it can be used directly to level multiple resources if they can be expressed in a common base; and (d) the model formulation is prepared automatically by an interface program, based on the CPM scheduling results.

The formulation presented can be used in studying the subject of cost scheduling with multiple objectives. For projects with large durations, the late start schedule provides the least interest cost, but such a schedule exhibits a serious problem since the project will be delayed if any activity of the project is delayed. On the other hand, the early start schedule alleviates this delay problem but it is associated with the greatest interest cost. A tradeoff model which enables project managers to select the optimum scheduling that satisfy the above contradicting objectives is needed.

Appendix I. References

Adrian, J. (1973). *Quantitative methods in construction management*. American Elsevier Publishing Co., New York, N.Y.

Ahuja, H. (1976). *Construction performance control by networks*. John Wiley and Sons, New York, N.Y.

Antill, J., and Woodhead, R. (1982). *Critical path methods in construction practice*. 3d Ed., Wiley-Interscience, New York, N.Y.

Burges, A., and Killebrew, J. (1962). Variation in activity level in a cyclical arrow diagram. *J. Indust. Engrg.*, 13(2).

Galbreath, R. (1965). Computer program for leveling resource usage. *J. Constr. Div.*, ASCE, 91(1), 107–124.

Harris, R. (1978). *Precedence and arrow networking techniques for construction*. John Wiley and Sons, New York, N.Y.

Karaa, F., and Nasr, A. (1986). Resource management in construction. *J. Constr. Engrg. and Mgmt.*, ASCE, 112(3), 346–357.

Moder, J., Phillips, C., and Davis, E. (1983). *Project management with CPM, PERT, and precedence diagramming*. 3d Ed., Van Nostrand Reinhold Company, New York, N.Y.

Phillips, D., and Garcia-Diaz, A. (1981). *Fundamentals of network analysis*. Prentice-Hall, Inc., Englewood Cliffs, N.J.

Schrage, L. (1981). *Linear programming models with LINDO*. The Scientific Press, Palo Alto, Calif.
Schrage, L. (1984). *Linear, integer, and quadratic programming with LINDO: User's manual*. The Scientific Press, Palo Alto, Calif.
Shaffer, L., Ritter, J., and Mayer, W. (1965). *The critical path method*. McGraw-Hill Book Co., New York, N.Y.

APPENDIX II. NOTATION

The following symbols are used in this paper:

c_i = desirable value of project resource rate for time unit i;
D_k, D_q = durations of noncritical activities k and q;
e_i^+, e_i^- = variables for time unit i that take positive value and zero, respectively, if $y_i > c_i$, and vice versa if $y_i < c_i$;
E_l = earliest start time of critical activity l;
ES_k, ES_q = earliest start times of noncritical activities k and q;
EF_k, EF_q = earliest finish times of noncritical activities k and q;
FF_k, FF_q = free floats of noncritical activities k and q;
IF_k = interfering float of noncritical activity k;
L_{kq} = lag time between activities k and q;
m = number of critical activities;
m' = number of critical activities along critical path;
n = number of noncritical activities;
n_i = number of noncritical activities that can be performed in time unit i;
n_k = number of activities immediately following noncritical activity k;
q = noncritical activity which immediately follows activity k;
R_k = resource rate of noncritical activity k;
S_l = resource rate of critical activity l;
T = duration of project;
T_l = duration of critical activity l;
TF_k, TF_q = total floats of noncritical activities k and q;
u_i^+, u_i^- = variables for time unit i that take positive value and zero, respectively, if $y_{i+1} > y_i$ and vice versa if $y_{i+1} < y_i$;
x_k, x_q = shiftings of noncritical activities k and q;
y_i = project resource rate for time unit i;
y_{ci} = resource rate of critical activity (or activities) for time unit i;
y_{pi} = total resource rate of noncritical activities p that can be performed for time unit i;
z = value of objective function; and
λ_{kj} = binary variable $\{0, 1\}$ of noncritical activity k for shifting equal to j.

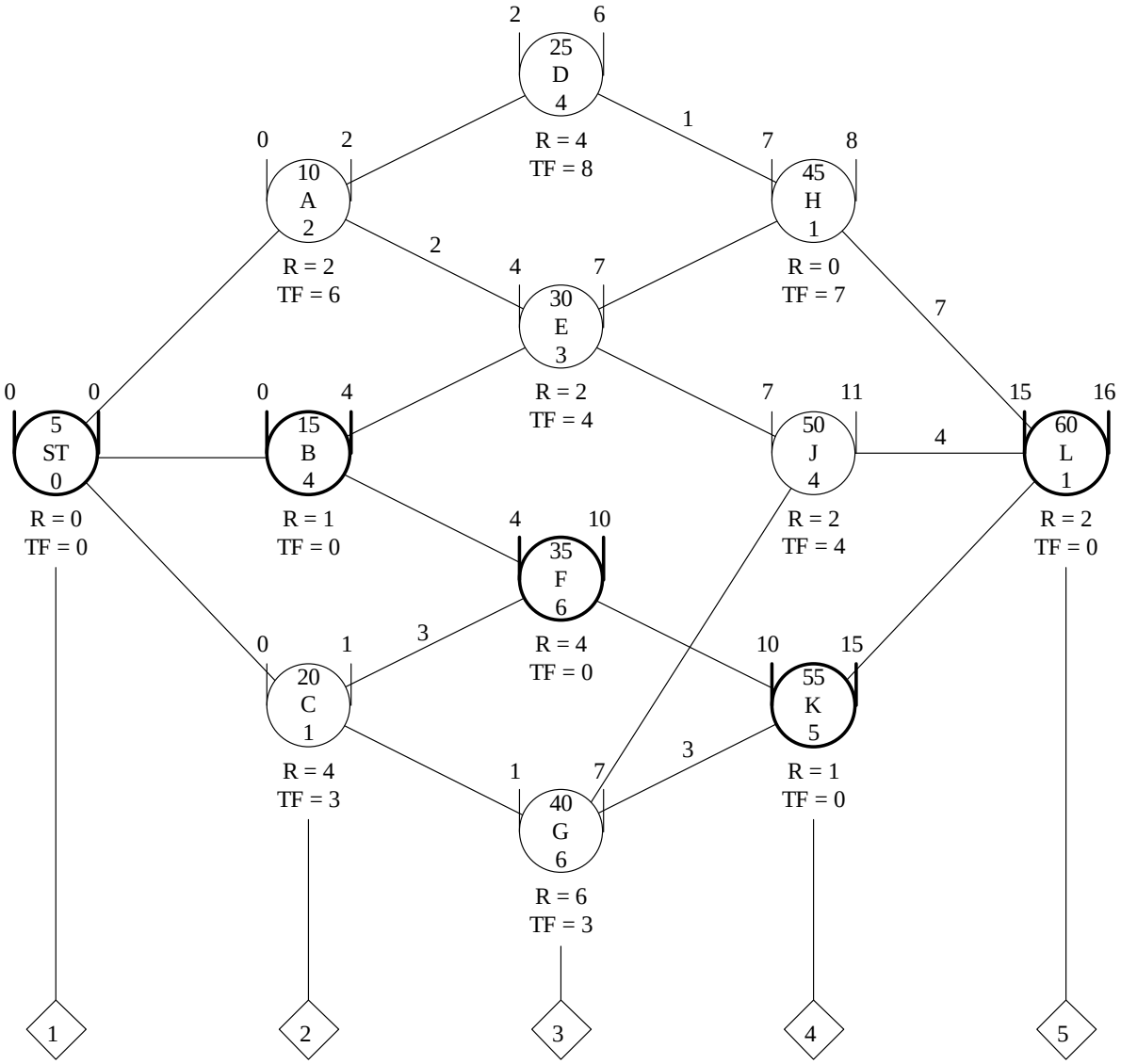

Figure 5 - Example Network

Resource Leveling Using Integer Programming (IP) in LINDO. P.G. Ioannou, June 14, 1999

Reference (IP): Easa, Said M., (1989), "Resource Leveling in Construction by Optimization," ASCE JCEM (115)2 302-316

Reference (Example): Harris, Robert B., (1989), PACK – Packing Method for Resource Leveling

PACK Example, Early Start Solution

Act.	TF	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
B	0	1	1	1	1												
F	0					4	4	4	4	4	4						
K	0											1	1	1	1	1	
L	0																2
A	6	2	2	---	---	---	---	---	---								
C	3	4	---	---	---												
D	8			4	4	4	4	---	---	---	---	---	---	---	---		
E	4					2	2	2	---	---	---	---					
G	3		6	6	6	6	6	6	---	---	---						
H	7								0	---	---	---	---	---	---	---	
J	4								2	2	2	2	---	---	---	---	
RS(J)		7	9	11	11	16	16	12	6	6	6	3	1	1	1	1	2
RIC		**1.552731252**														Sum =	109

2L_A0	2L_A0					Daily resource requirements of non-crtical activities based on their shifts LW_k										
	2L_A1	2L_A1														
		2L_A2	2L_A2													
4L_C0			2L_A3	2L_A3												
	4L_C1			2L_A4	2L_A4											
		4L_C2			2L_A5	2L_A5										
			4L_C3			2L_A6	2L_A6									
		4L_D0	4L_D0	4L_D0	4L_D0											
			4L_D1	4L_D1	4L_D1	4L_D1										
				4L_D2	4L_D2	4L_D2	4L_D2									
					4L_D3	4L_D3	4L_D3	4L_D3								
						4L_D4	4L_D4	4L_D4	4L_D4							
							4L_D5	4L_D5	4L_D5	4L_D5						
								4L_D6	4L_D6	4L_D6	4L_D6	4L_D6				
				2L_E0	2L_E0	2L_E0			4L_D7	4L_D7	4L_D7	4L_D7	4L_D7			
					2L_E1	2L_E1	2L_E1			4L_D8	4L_D8	4L_D8	4L_D8			
						2L_E2	2L_E2	2L_E2								
							2L_E3	2L_E3	2L_E3							
								2L_E4	2L_E4	2L_E4						
	6L_G0	6L_G0	6L_G0	6L_G0	6L_G0	6L_G0	6L_G0									
		6L_G1	6L_G1	6L_G1	6L_G1	6L_G1	6L_G1	6L_G1								
			6L_G2	6L_G2	6L_G2	6L_G2	6L_G2	6L_G2	6L_G2							
				6L_G3	6L_G3	6L_G3	6L_G3	6L_G3	6L_G3							
							2L_J0	2L_J0	2L_J0	2L_J0						
								2L_J1	2L_J1	2L_J1	2L_J1					
									2L_J2	2L_J2	2L_J2	2L_J2				
										2L_J3	2L_J3	2L_J3	2L_J3			
											2L_J4	2L_J4	2L_J4	2L_J4		

Resource Leveling Using Integer Programming (IP) in LINDO. P.G. Ioannou, June 14, 1999

Reference (IP): Easa, Said M., (1989), "Resource Leveling in Construction by Optimization," ASCE JCEM (115)2 302-316

Reference (Example): Harris, Robert B., (1989), PACK – Packing Method for Resource Leveling

PACK Example, Early Start Solution

Act.	TF	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
B	0	1	1	1	1												
F	0					4	4	4	4	4	4						
K	0											1	1	1	1	1	
L	0																2
A	6	2	2	---	---	---	---	---	---								
C	3	4	---	---	---												
D	8			4	4	4	4	---	---	---	---	---	---	---	---		
E	4					2	2	2	---	---	---	---					
G	3		6	6	6	6	6	6	---	---	---						
H	7								0	---	---	---	---	---	---	---	
J	4								2	2	2	2	---	---	---	---	
RS(J)		7	9	11	11	16	16	12	6	6	6	3	1	1	1	1	2
RIC		**1.552731252**														Sum =	109

The following constraints define the daily resource requirement for IP in LINDO.
yj (the total resource requirement in day j) depends on which activities work on that day.
L_Wk is a 0/1 variable that equals 1 only when activity W will be shifted by exactly k days from its early start (otherwise it is 0).
Each L_Xk below is multiplied by the daily resource requirement for the corresponding activity W.
The RHS values are the resource histogram based on an early start solution.

y1 + 2L_A1 + 2L_A2 + 2L_A3 + 2L_A4 + 2L_A5 + 2L_A6 + 4L_C1 + 4L_C2 + 4L_C3 = 7

y2 + 2L_A2 + 2L_A3 + 2L_A4 + 2L_A5 + 2L_A6 - 4L_C1 + 6L_G1 + 6L_G2 + 6L_G3 = 9

y3 - 2L_A1 - 2L_A2 - 4L_C2 + 4L_D1 + 4L_D2 + 4L_D3 + 4L_D4 + 4L_D5 + 4L_D6 + 4L_D7 + 4L_D8 + 6L_G2 + 6L_G3 = 11

y4 - 2L_A2 - 2L_A3 - 4L_C3 + 4L_D2 + 4L_D3 + 4L_D4 + 4L_D5 + 4L_D6 + 4L_D7 + 4L_D8 + 6L_G3 = 11

y5 - 2L_A3 - 2L_A4 + 4L_D3 + 4L_D4 + 4L_D5 + 4L_D6 + 4L_D7 + 4L_D8 + 2L_E1 + 2L_E2 + 2L_E3 + 2L_E4 = 16

y6 - 2L_A4 - 2L_A5 + 4L_D4 + 4L_D5 + 4L_D6 + 4L_D7 + 4L_D8 + 2L_E2 + 2L_E3 + 2L_E4 = 16

y7 - 2L_A5 - 2L_A6 - 4L_D1 - 4L_D2 - 4L_D3 - 4L_D4 + 2L_E3 + 2L_E4 = 12

y8 - 2L_A6 - 4L_D2 - 4L_D3 - 4L_D4 - 4L_D5 - 2L_E1 - 2L_E2 - 2L_E3 - 6L_G1 - 6L_G2 - 6L_G3 + 2L_J1 + 2L_J2 + 2L_J3 + 2L_J4 = 6

y9 - 4L_D3 - 4L_D4 - 4L_D5 - 4L_D6 - 2L_E2 - 2L_E3 - 2L_E4 - 6L_G2 - 6L_G3 + 2L_J2 + 2L_J3 + 2L_J4 = 6

y10 - 4L_D4 - 4L_D5 - 4L_D6 - 4L_D7 - 2L_E3 - 2L_E4 - 6L_G3 + 2L_J3 + 2L_J4 = 6

y11 - 4L_D5 - 4L_D6 - 4L_D7 - 4L_D8 - 2L_E4 + 2L_J4 = 3

y12 - 4L_D6 - 4L_D7 - 4L_D8 - 2L_J1 - 2L_J2 - 2L_J3 - 2L_J4 = 1

y13 - 4L_D7 - 4L_D8 - 2L_J2 - 2L_J3 - 2L_J4 = 1

y14 - 4L_D8 - 2L_J3 - 2L_J4 = 1

y15 - 2L_J4 = 1

y16 = 2

```
! Resource Leveling Using Integer Programming (IP) in LINDO.
! P.G. Ioannou, June 14, 1999
! Reference (IP): Easa, Said M., (1989), "Resource Leveling in Construction by Optimization," ASCE JCEM (115)2 302-316
! Reference (Example): Harris, Robert B., (1989), PACK - Packing Method for Resource Leveling

! The following formulation minimizes the sum of the absolute values of the resource requirement deviations from 7.

MIN Em1 + Ep1 + Em2 + Ep2 + Em3 + Ep3 + Em4 + Ep4 + Em5 + Ep5 + Em6 + Ep6 + Em7 + Ep7 + Em8 + Ep8 + Em9 + Ep9
  + Em10 + Ep10 + Em11 + Ep11 + Em12 + Ep12 + Em13 + Ep13 + Em14 + Ep14 + Em15 + Ep15 + Em16 + Ep16

SUBJECT TO
! yj is the total resource requirement in day j.
! yj - 7 = Epj - Emj is the resource deviation from 7 resources per day (approx. equal to (109 res_days)/(16 days)
y1 + Em1 - Ep1 = 7
y2 + Em2 - Ep2 = 7
y3 + Em3 - Ep3 = 7
y4 + Em4 - Ep4 = 7
y5 + Em5 - Ep5 = 7
y6 + Em6 - Ep6 = 7
y7 + Em7 - Ep7 = 7
y8 + Em8 - Ep8 = 7
y9 + Em9 - Ep9 = 7
y10 + Em10 - Ep10 = 7
y11 + Em11 - Ep11 = 7
y12 + Em12 - Ep12 = 7
y13 + Em13 - Ep13 = 7
y14 + Em14 - Ep14 = 7
y15 + Em15 - Ep15 = 7
y16 + Em16 - Ep16 = 7

! yj (the total resource requirement in day j) depends on which activities work on that day.
! L_Wk is a 0/1 variable that equals 1 only when activity W will be shifted by exactly k days from its early start (otherwise it is 0).
! Each L_Xk below is multiplied by the daily resource requirement for the corresponding activity W.
! The RHS values are the resource histogram based on an early start solution.
y1  + 2L_A1 + 2L_A2 + 2L_A3 + 2L_A4 + 2L_A5 + 2L_A6 + 4L_C1 + 4L_C2 + 4L_C3 = 7
y2  + 2L_A2 + 2L_A3 + 2L_A4 + 2L_A5 + 2L_A6 - 4L_C1 + 6L_G1 + 6L_G2 + 6L_G3 = 9
y3  - 2L_A1 - 2L_A2 - 4L_C2 + 4L_D1 + 4L_D2 + 4L_D3 + 4L_D4 + 4L_D5 + 4L_D6 + 4L_D7 + 4L_D8  + 6L_G2 + 6L_G3 = 11
y4  - 2L_A2 - 2L_A3 - 4L_C3 + 4L_D2 + 4L_D3 + 4L_D4 + 4L_D5 + 4L_D6 + 4L_D7 + 4L_D8 + 6L_G3 = 11
y5  - 2L_A3 - 2L_A4 + 4L_D3 + 4L_D4 + 4L_D5 + 4L_D6 + 4L_D7 + 4L_D8 + 2L_E1 + 2L_E2 + 2L_E3 + 2L_E4 = 16
y6  - 2L_A4 - 2L_A5 + 4L_D4 + 4L_D5 + 4L_D6 + 4L_D7 + 4L_D8 + 2L_E2 + 2L_E3 + 2L_E4 = 16
y7  - 2L_A5 - 2L_A6 - 4L_D1 - 4L_D2 - 4L_D3 - 4L_D4 + 2L_E3 + 2L_E4 = 12
y8  - 2L_A6 - 4L_D2 - 4L_D3 - 4L_D4 - 4L_D5 - 2L_E1 - 2L_E2 - 2L_E3 - 6L_G1 - 6L_G2 - 6L_G3 + 2L_J1 + 2L_J2 + 2L_J3 + 2L_J4 = 6
y9  - 4L_D3 - 4L_D4 - 4L_D5 - 4L_D6 - 2L_E2 - 2L_E3 - 2L_E4 - 6L_G2 - 6L_G3 + 2L_J2 + 2L_J3 + 2L_J4 = 6
y10 - 4L_D4 - 4L_D5 - 4L_D6 - 4L_D7 - 2L_E3 - 2L_E4 - 6L_G3  + 2L_J3 + 2L_J4 = 6
y11 - 4L_D5 - 4L_D6 - 4L_D7 - 4L_D8 - 2L_E4 + 2L_J4 = 3
y12 - 4L_D6 - 4L_D7 - 4L_D8 - 2L_J1 - 2L_J2 - 2L_J3 - 2L_J4 = 1
y13 - 4L_D7 - 4L_D8 - 2L_J2 - 2L_J3 - 2L_J4 = 1
y14 - 4L_D8 - 2L_J3 - 2L_J4 = 1
y15 - 2L_J4 = 1
y16 = 2

! X_W is the (forward) shift of activity W from its Early Start position (X_W <= TF of activity W)
X_A - L_A1 - 2 L_A2 - 3 L_A3 - 4 L_A4 - 5 L_A5 - 6 L_A6 = 0
X_C - L_C1 - 2 L_C2 - 3 L_C3 = 0
X_D - L_D1 - 2 L_D2 - 3 L_D3 - 4 L_D4 - 5 L_D5 - 6 L_D6 - 7 L_D7 - 8 L_D8 = 0
X_E - L_E1 - 2 L_E2 - 3 L_E3 - 4 L_E4 = 0
X_G - L_G1 - 2 L_G2 - 3 L_G3 = 0
X_H - L_H1 - 2 L_H2 - 3 L_H3 - 4 L_H4 - 5 L_H5 - 6 L_H6 - 7 L_H7 = 0
X_J - L_J1 - 2 L_J2 - 3 L_J3 - 4 L_J4 = 0
```

```
! At most only one of L_Wk can equal 1 for a given activity W. When all are 0 activity W is not shifted from its Early Start position.
L_A1 + L_A2 + L_A3 + L_A4 + L_A5 + L_A6 <= 1
L_C1 + L_C2 + L_C3 <=1
L_D1 + L_D2 + L_D3 + L_D4 + L_D5 + L_D6 + L_D7 + L_D8 <= 1
L_E1 + L_E2 + L_E3 + L_E4 <= 1
L_G1 + L_G2 + L_G3 <= 1
L_H1 + L_H2 + L_H3 + L_H4 + L_H5 + L_H6 + L_H7 <= 1
L_J1 + L_J2 + L_J3 + L_J4 <= 1

! For every link IJ, the shift of the predecessor X_I cannot exceed the shift of the successor X_J incremented by the lag_IJ
X_A - X_D <= 0
X_A - X_E <= 2
X_C - X_G <= 0
X_D - X_H <= 1
X_E - X_H <= 0
X_E - X_J <= 0
X_G - X_J <= 0

END

INT L_A1
INT L_A2
INT L_A3
INT L_A4
INT L_A5
INT L_A6

INT L_C1
INT L_C2
INT L_C3

INT L_D1
INT L_D2
INT L_D3
INT L_D4
INT L_D5
INT L_D6
INT L_D7
INT L_D8

INT L_E1
INT L_E2
INT L_E3
INT L_E4

INT L_G1
INT L_G2
INT L_G3

INT L_J1
INT L_J2
INT L_J3
INT L_J4

!INT L_H1
!INT L_H2
!INT L_H3
!INT L_H4
!INT L_H5
!INT L_H6
!INT L_H7
```

```
! Resource Leveling Using Integer Programming (IP) in LINDO.
! P.G. Ioannou, June 14, 1999
! Reference (IP): Easa, Said M., (1989), "Resource Leveling in Construction by Optimization," ASCE JCEM (115)2 302-316
! Reference (Example): Harris, Robert B., (1989), PACK - Packing Method for Resource Leveling

! The following formulation minimizes the sum of the absolute values of the changes in resource requirements from day-to-day.

MIN Um1 + Up1 + Um2 + Up2 + Um3 + Up3 + Um4 + Up4 + Um5 + Up5 + Um6 + Up6 + Um7 + Up7 + Um8 + Up8 + Um9 + Up9
  + Um10 + Up10 + Um11 + Up11 + Um12 + Up12 + Um13 + Up13 + Um14 + Up14 + Um15 + Up15

SUBJECT TO
! yj+1 = yj - Umj + (Upj) is the amount by which y(j+1) is
! yj+1 is the total resource requirement in day j+1 and yj is the total resource requirement in day j.
y2 - y1 + Um1 - Up1 = 0
y3 - y2 + Um2 - Up2 = 0
y4 - y3 + Um3 - Up3 = 0
y5 - y4 + Um4 - Up4 = 0
y6 - y5 + Um5 - Up5 = 0
y7 - y6 + Um6 - Up6 = 0
y8 - y7 + Um7 - Up7 = 0
y9 - y8 + Um8 - Up8 = 0
y10 - y9 + Um9 - Up9 = 0
y11 - y10 + Um10 - Up10 = 0
y12 - y11 + Um11 - Up11 = 0
y13 - y12 + Um12 - Up12 = 0
y14 - y13 + Um13 - Up13 = 0
y15 - y14 + Um14 - Up14 = 0
y16 - y15 + Um15 - Up15 = 0

! yj (the total resource requirement in day j) depends on which activities work on that day.
! L_Wk is a 0/1 variable that equals 1 only when activity W will be shifted by exactly k days from its early start (otherwise it is 0).
! Each L_Xk below is multiplied by the daily resource requirement for the corresponding activity W.
! The RHS values are the resource histogram based on an early start solution.
y1  + 2L_A1 + 2L_A2 + 2L_A3 + 2L_A4 + 2L_A5 + 2L_A6 + 4L_C1 + 4L_C2 + 4L_C3 = 7
y2  + 2L_A2 + 2L_A3 + 2L_A4 + 2L_A5 + 2L_A6 - 4L_C1 + 6L_G1 + 6L_G2 + 6L_G3 = 9
y3  - 2L_A1 - 2L_A2 - 4L_C2 + 4L_D1 + 4L_D2 + 4L_D3 + 4L_D4 + 4L_D5 + 4L_D6 + 4L_D7 + 4L_D8  + 6L_G2 + 6L_G3 = 11
y4  - 2L_A2 - 2L_A3 - 4L_C3 + 4L_D2 + 4L_D3 + 4L_D4 + 4L_D5 + 4L_D6 + 4L_D7 + 4L_D8 + 6L_G3 = 11
y5  - 2L_A3 - 2L_A4 + 4L_D3 + 4L_D4 + 4L_D5 + 4L_D6 + 4L_D7 + 4L_D8 + 2L_E1 + 2L_E2 + 2L_E3 + 2L_E4 = 16
y6  - 2L_A4 - 2L_A5 + 4L_D4 + 4L_D5 + 4L_D6 + 4L_D7 + 4L_D8 + 2L_E2 + 2L_E3 + 2L_E4 = 16
y7  - 2L_A5 - 2L_A6 - 4L_D1 - 4L_D2 - 4L_D3 - 4L_D4 + 2L_E3 + 2L_E4 = 12
y8  - 2L_A6 - 4L_D2 - 4L_D3 - 4L_D4 - 4L_D5 - 2L_E1 - 2L_E2 - 2L_E3 - 6L_G1 - 6L_G2 - 6L_G3 + 2L_J1 + 2L_J2 + 2L_J3 + 2L_J4 = 6
y9  - 4L_D3 - 4L_D4 - 4L_D5 - 4L_D6 - 2L_E2 - 2L_E3 - 2L_E4 - 6L_G2 - 6L_G3 + 2L_J2 + 2L_J3 + 2L_J4 = 6
y10 - 4L_D4 - 4L_D5 - 4L_D6 - 4L_D7 - 2L_E3 - 2L_E4 - 6L_G3  + 2L_J3 + 2L_J4 = 6
y11 - 4L_D5 - 4L_D6 - 4L_D7 - 4L_D8 - 2L_E4 + 2L_J4 = 3
y12 - 4L_D6 - 4L_D7 - 4L_D8 - 2L_J1 - 2L_J2 - 2L_J3 - 2L_J4 = 1
y13 - 4L_D7 - 4L_D8 - 2L_J2 - 2L_J3 - 2L_J4 = 1
y14 - 4L_D8 - 2L_J3 - 2L_J4 = 1
y15 - 2L_J4 = 1
y16 = 2

! X_W is the (forward) shift of activity W from its Early Start position (X_W <= TF of activity W)
X_A - L_A1 - 2 L_A2 - 3 L_A3 - 4 L_A4 - 5 L_A5 - 6 L_A6 = 0
X_C - L_C1 - 2 L_C2 - 3 L_C3 = 0
X_D - L_D1 - 2 L_D2 - 3 L_D3 - 4 L_D4 - 5 L_D5 - 6 L_D6 - 7 L_D7 - 8 L_D8 = 0
X_E - L_E1 - 2 L_E2 - 3 L_E3 - 4 L_E4 = 0
X_G - L_G1 - 2 L_G2 - 3 L_G3 = 0
X_H - L_H1 - 2 L_H2 - 3 L_H3 - 4 L_H4 - 5 L_H5 - 6 L_H6 - 7 L_H7 = 0
X_J - L_J1 - 2 L_J2 - 3 L_J3 - 4 L_J4 = 0
```

```
! At most only one of L_Wk can equal 1 for a given activity W. When all are 0 activity W is not shifted from its Early Start position.
L_A1 + L_A2 + L_A3 + L_A4 + L_A5 + L_A6 <= 1
L_C1 + L_C2 + L_C3 <=1
L_D1 + L_D2 + L_D3 + L_D4 + L_D5 + L_D6 + L_D7 + L_D8 <= 1
L_E1 + L_E2 + L_E3 + L_E4 <= 1
L_G1 + L_G2 + L_G3 <= 1
L_H1 + L_H2 + L_H3 + L_H4 + L_H5 + L_H6 + L_H7 <= 1
L_J1 + L_J2 + L_J3 + L_J4 <= 1

! For every link IJ, the shift of the predecessor X_I cannot exceed the shift of the successor X_J incremented by the lag_IJ
X_A - X_D <= 0
X_A - X_E <= 2
X_C - X_G <= 0
X_D - X_H <= 1
X_E - X_H <= 0
X_E - X_J <= 0
X_G - X_J <= 0

END

INT L_A1
INT L_A2
INT L_A3
INT L_A4
INT L_A5
INT L_A6

INT L_C1
INT L_C2
INT L_C3

INT L_D1
INT L_D2
INT L_D3
INT L_D4
INT L_D5
INT L_D6
INT L_D7
INT L_D8

INT L_E1
INT L_E2
INT L_E3
INT L_E4

INT L_G1
INT L_G2
INT L_G3

INT L_J1
INT L_J2
INT L_J3
INT L_J4

!INT L_H1
!INT L_H2
!INT L_H3
!INT L_H4
!INT L_H5
!INT L_H6
!INT L_H7
```

Resource Leveling Using Integer Programming (IP) in LINDO. P.G. Ioannou, June 14, 1999

PACK Example, Early Start Solution

Act.	TF	Day 1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	
B	0	1	1	1	1													
F	0					4	4	4	4	4	4							
K	0											1	1	1	1	1		
L	0																2	
A	6	2	2	---	---	---	---	---	---									
C	3	4	---	---	---													
D	8			4	4	4	4	---	---	---	---	---	---	---	---			
E	4					2	2	2	---	---	---	---						
G	3		6	6	6	6	6	6	---	---	---							
H	7								0	---	---	---	---	---	---	---		
J	4								2	2	2	2	---	---	---	---		Sum
RS(J)		7	9	11	11	16	16	12	6	6	6	3	1	1	1	1	2	109
Abs(7-RS(J))		0	2	4	4	9	9	5	1	1	1	4	6	6	6	6	5	69
Abs(dRS(J))		2	2	0	5	0	4	6	0	0	3	2	0	0	0	1		25

RIC **1.552731252**

PACK & IP Solution: Minimize the absolute values of daily deviations from 7 resources (109/16 =6.81)

Act.	Shift	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	
B		1	1	1	1													
F						4	4	4	4	4	4							
K												1	1	1	1	1		
L																	2	
A	0	2	2															
C	0	4																
D	8											4	4	4	4			
E	3								2	2	2							
G	0		6	6	6	6	6	6										
H	7															0		
J	4												2	2	2	2		Sum
Resource		7	9	7	7	10	10	10	6	6	6	5	7	7	7	3	2	109
Abs(7-RS(J))		0	2	0	0	3	3	3	1	1	1	2	0	0	0	4	5	25
Abs(dRS(J))		2	2	0	3	0	0	4	0	0	1	2	0	0	4	1		19

RIC **1.105630839**

IP Solution: Minimize the absolute values of resource demand changes from day-to-day

Act.	Shift	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	
B		1	1	1	1													
F						4	4	4	4	4	4							
K												1	1	1	1	1		
L																	2	
A	2			2	2													
C	0	4																
D	5								4	4	4	4						
E	3								2	2	2							
G	0		6	6	6	6	6	6										
H	4												0					
J	4												2	2	2	2		Sum
RS(J)		5	7	9	9	10	10	10	10	10	10	5	3	3	3	3	2	109
Abs(7-RS(J))		2	0	2	2	3	3	3	3	3	3	2	4	4	4	4	5	47
Abs(dRS(J))		2	2	0	1	0	0	0	0	0	5	2	0	0	0	1		13

RIC **1.213365878**

Resource Leveling Using Integer Programming (IP) in LINDO. P.G. Ioannou, June 14, 1999

IP Solution: Minimize the absolute values of daily deviations from 6 resources

Act.	Shift	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	
B		1	1	1	1													
F						4	4	4	4	4	4							
K												1	1	1	1	1		
L																	2	
A	6							2	2									
C	0	4																
D	8											4	4	4	4			
E	4									2	2	2						
G	0		6	6	6	6	6	6										
H	7															0		
J	4												2	2	2	2		Sum
Resource		5	7	7	7	10	10	12	6	6	6	7	7	7	7	3	2	109
Abs(6-RS(J))		1	1	1	1	4	4	6	0	0	0	1	1	1	1	3	4	29
Abs(dRS(J))		2	0	0	3	0	2	6	0	0	1	0	0	0	4	1		19

RIC **1.121791095**

PACK & IP Solution: Minimize the absolute values of daily deviations from 7 or 8 resources

Act.	Shift	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	
B		1	1	1	1													
F						4	4	4	4	4	4							
K												1	1	1	1	1		
L																	2	
A	0	2	2															
C	0	4																
D	8											4	4	4	4			
E	3								2	2	2							
G	0		6	6	6	6	6	6										
H	7															0		
J	4												2	2	2	2		Sum
Resource		7	9	7	7	10	10	10	6	6	6	5	7	7	7	3	2	109
Abs(7-RS(J))		0	2	0	0	3	3	3	1	1	1	2	0	0	0	4	5	25
Abs(dRS(J))		2	2	0	3	0	0	4	0	0	1	2	0	0	4	1		19

RIC **1.105630839**

Alternate (bad) IP Solution: Minimize the absolute values of daily deviations from 7 or 8 resources

Act.	Shift	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	
B		1	1	1	1													
F						4	4	4	4	4	4							
K												1	1	1	1	1		
L																	2	
A	0	2	2															
C	0	4																
D	8											4	4	4	4			
E	3								2	2	2							
G	0		6	6	6	6	6	6										
H	7															0		
J	3											2	2	2	2			Sum
Resource		7	9	7	7	10	10	10	6	6	6	7	7	7	7	1	2	109
Abs(8-RS(J))		1	1	1	1	2	2	2	2	2	2	1	1	1	1	7	6	33
Abs(dRS(J))		2	2	0	3	0	0	4	0	0	1	0	0	0	6	1		19

RIC **1.127177847**