

A nonlinear allocation problem

by E. V. Denardo
A. J. Hoffman
T. Mackenzie
W. R. Pulleyblank

We consider the problem of deploying work force to tasks in a project network for which the time required to perform each task depends on the assignment of work force to the task, for the purpose of minimizing the time to complete the project. The rules governing the deployment of work force and the resulting changes in task times of our problem are discussed in the contexts of a) related work on project networks and b) more general allocation problems on polytopes. We prove that, for these problems, the obvious lower bound for project completion time is attainable.

1. Introduction

A PERT network is an approach to organizing a project that consists of individual tasks satisfying precedence relations. The project is modeled as an acyclic directed graph with initial node s , terminal node t , and tasks $\{T_e\}$ corresponding to edges $\{e = (u, v)\}$ of the graph. For each T_e , we are given $d_e > 0$, the time to perform the task. $T_{(u,v)}$ cannot be started until all tasks $T_{(w,u)}$ are completed. Because of this, the earliest possible completion time y_t of the project is the length of a longest st -path with respect to the edge lengths d_e . This value represents the most time-consuming sequence of activities that must be performed in completing the project.

There is interest in studying how y_t can be altered if the set $\{d_e\}$ can be changed by actions of the project planner. The allocation of extra resources to critical tasks in a

project so as to reduce their duration is commonly called "crashing." We cite two examples from the literature.

Example 1.1 [1] For each T_e , d_e is changed to $d_e - m_e z_e$, where each m_e is a prescribed positive number, each z_e is a nonnegative variable bounded from above, and $\sum z_e$ equals a prescribed value.

Example 1.2 [2] For each T_e , d_e is changed to d_e/z_e , where each z_e is a positive variable and $\sum z_e$ equals a prescribed value.

In both these cases, the general objective is to allocate the resources in such a way that the length of a longest path is minimized.

In this paper we consider a problem similar to Example 1.2, but one in which the resources may be reused in the manner described below. Again, d_e is replaced by d_e/z_e , but we impose conditions different from those of Example 1.2 on the positive work force variables z_e , which we do not require to be integral. As in Example 1.2, d_e represents the time to perform T_e with one unit of work force, and d_e/z_e the time with z_e units. When a task is completed, in the variation we consider, the work force assigned to that task may then be assigned to different tasks. As usual, a new task may start when all of its predecessor tasks have been completed.

A total of W units of work force are available. These units can be shifted from task to task, but at no time can more than W units be active. How shall the work force be assigned to minimize the completion time of the total project? We put some restrictions on this allocation below, but first we make some preliminary remarks.

©Copyright 1994 by International Business Machines Corporation. Copying in printed form for private use is permitted without payment of royalty provided that (1) each reproduction is done without alteration and (2) the *Journal* reference and IBM copyright notice are included on the first page. The title and abstract, but no other portions, of this paper may be copied or distributed royalty free without further permission by computer-based and other information-service systems. Permission to republish any other portion of this paper must be obtained from the Editor.

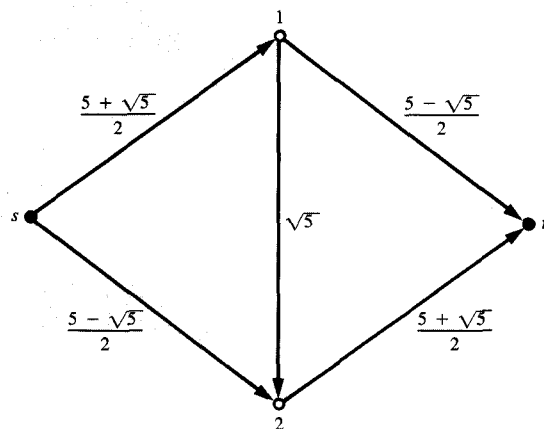


Figure 1

Sample PERT network. Labels on the edges are values of z_e for the optimal assignment of work force satisfying Condition 1.3 when all $d_e = 1$ and $W = 5$.

If work force can be reassigned at any time to any task that is available, given the precedence restrictions of the PERT network, it is easy to prove that *every* allocation that keeps the work force busy will complete the project in time $\Sigma d_e/W$.

If the work force assigned to a task cannot be changed during the performance of the task, then $\Sigma d_e/W$ is a lower bound on the completion time, and can be attained in many ways (for instance, by using the entire work force on each task as it becomes eligible).

The restriction we consider on the allocations was suggested to us by some managers of software projects. They pointed out that it is desirable to assign work force to tasks that grow from and relate to tasks that they have just performed. A rough attempt to model such a desideratum would be the requirement that the work force satisfy the following "flow" condition.

Condition 1.3 For each node $v \neq s, t$, the total work force assigned to tasks with terminal node v equals the total work force assigned to tasks with initial node v . The total work force assigned to tasks with initial node s equals W , which equals the total work force assigned to tasks with terminal node t .

Figure 1 gives an example of an allocation satisfying Condition 1.3, with all $d_e = 1$ and $W = 5$. The project is completed in time $\Sigma d_e/W = 1$. In view of the foregoing, this is optimum. Moreover, it is the *unique* optimum satisfying Condition 1.3. It is also the only allocation for which all three st -paths have the identical length, 1.

We show below that, for every problem, the minimum completion time is $\Sigma d_e/W$, and is attained by a unique allocation. In Section 2, we describe a general mathematical programming model that encompasses the above problem. This involves the concepts of *positive polytope* and *flat positive polytope*, for which we offer two *Pangloss* theorems*. The first is proved in Section 3 for positive polytopes. A strengthening for flat positive polytopes is established in Section 4. In Section 5, we explore the relation between these two theorems, and in Section 6 we present concluding remarks.

For convenience, in the rest of this paper, we take $W = 1$. Because we impose no integrality conditions on z_e , we lose no generality.

2. Allocation problems on positive polytopes

A polytope is a bounded polyhedron. A point x is called *positive* (written $x > 0$) if $x_i > 0$ for all i . We call a polytope *positive* if it is entirely contained in the nonnegative orthant and contains at least one positive point. We call a positive polytope *flat* if it is the set of nonnegative solutions to a nontrivial system of linear equations.

For a PERT network, let P be the set of all allocations satisfying Condition 1.3. (Recall that we have set $W = 1$.) Then P is a polyhedron, and, since PERT networks are acyclic, P is bounded. Each extreme point of P corresponds to an assignment of one unit to the edges in an st -path. The polytope P is positive because each edge is part of some st -path.

Let $d = (d_1, \dots, d_n) > 0$ be any positive vector indexed by the edges of our network. Consider the problem

$$\max_{x \in P} \sum d_j x_j. \quad (2.1)$$

Since each extreme point of P assigns one unit to the edges belonging to a particular st -path, (2.1) computes the length of a longest st -path, with respect to edge lengths d_e .

We consider the PERT problem in which the time needed for task T_j equals d_j/z_j , where z_j is the amount of work force assigned to task T_j . Condition 1.3 states that $z \in P$; that is, the assignment of work force must be an st flow of one unit. Hence, our PERT problem is this variation of (2.1):

$$\min_{z \in P} \max_{x \in P} \sum (d_j/z_j) x_j. \quad (2.2)$$

Thus, (2.2) allocates the work force so as to minimize the length of a longest st -path, where the length of edge j is d_j/z_j .

*In Voltaire's *Candide*, Dr. Pangloss says, "All is for the best ... in this best of all possible worlds."

We actually study (2.1) for every positive polytope P and every positive vector $\mathbf{d}$, not just for those arising from PERT problems.

Note that for any positive $\mathbf{z} \in P$, if we define $c_j = d_j/z_j$ for all $j = 1, \dots, n$, then

$$\max_{\mathbf{x} \in P} \sum c_j x_j \geq \sum c_j z_j = \sum d_j. \quad (2.3)$$

One principal result is the following.

• **Theorem 2.1 (Pangloss theorem for positive polytopes)**
Let $P \subset \mathbb{R}^n$ be a positive polytope and let $\mathbf{d} = (d_1, d_2, \dots, d_n) > \mathbf{0}$. Then there exists a unique $\mathbf{c} = (c_1, \dots, c_n) > \mathbf{0}$ such that

there exists positive $\mathbf{z} \in P$ such that $c_j z_j = d_j$ for all j

and

$$\max_{\mathbf{x} \in P} \sum c_j x_j = \sum d_j = \sum c_j z_j. \quad (2.4)$$

This can be restated as follows: Interpret each $\mathbf{x} \in P$ as a vector of feasible activities. Suppose we are given a vector $\mathbf{d}$ of target revenues for our activities. Then there exists a unique unit profit for each activity with the following property: If we allocate our resources so as to maximize the total revenue with respect to these unit profits, each activity in the optimum solution generates exactly its prescribed target revenue. More simply, every vector of proposed target revenues is the optimum set of revenues for a suitably chosen set of unit profits. This is why we refer to this as a "Pangloss" theorem.

Theorem 2.1 is the same as

$$\min_{\mathbf{z} \in P} \max_{\mathbf{x} \in P} \sum (d_j/z_j) x_j = \sum d_j, \quad (2.5)$$

which we prove in the next section.

We have the following consequence.

• **Corollary 2.2**

Let $\mathbf{z}$ be the optimal solution to (2.5). Let S be the set of extreme points of P . Let $T \subseteq S$ be such that $\mathbf{z}$ can be expressed as a convex combination of $\{\mathbf{x} : \mathbf{x} \in T\}$ with positive weights. Then

$$\sum d_j = \sum (d_j/z_j) x_j \text{ for each } \mathbf{x} \in T. \quad (2.6)$$

Let us interpret this corollary for the case of PERT networks. With P defined as in the second paragraph of this section, it is well known that each positive vector $\mathbf{x}$ in P is a positive convex combination of all extreme points of P . Hence, for a PERT network, Corollary 2.2 shows that every st -path has length $\sum d_j$, the length of edge j being d_j/z_j . For a more general positive polytope P , however, there can exist extreme points $\mathbf{x}$, which we cannot use to express $\mathbf{z}$, for which $\sum (d_j/z_j) x_j < \sum d_j$.

We now describe an example of a polyhedron P that is positive but not flat. Consider a graph with node set $\{s, t, 1, 2, 3, 4, 5\}$ containing the following three st -paths:

$$p_1 = \{(s, 1), (1, 3), (3, 4), (4, t)\},$$

$$p_2 = \{(s, 1), (1, 3), (3, 5), (5, t)\},$$

$$p_3 = \{(s, 2), (2, 3), (3, 5), (5, t)\}.$$

Let $\mathbf{v}(p_i)$ be the path-edge incidence vector of path p_i , that is, the n -dimensional vector whose j th component equals 1 if edge j is in path p_i , and equals 0 otherwise. Define P to be the convex hull of $\{\mathbf{v}(p_1), \mathbf{v}(p_2), \mathbf{v}(p_3)\}$. Any hyperplane containing P must contain the incidence vector $\mathbf{v}(p_4)$ of $p_4 \equiv \{(s, 2), (2, 3), (3, 4), (4, t)\}$, since $\mathbf{v}(p_1) + \mathbf{v}(p_3) - \mathbf{v}(p_2) = \mathbf{v}(p_4)$. The positive polytope P is not flat because it does not contain $\mathbf{v}(p_4)$.

We return to flat positive polytopes in Section 4.

3. Pangloss theorem for positive polytopes

It is possible that this is a folk theorem in utility theory, but we do not know of any reference. We offer two proofs for Theorem 2.1. For the first, consider the optimization problem:

$$\min_{\mathbf{z} \in P} F(\mathbf{z}) \equiv \sum -d_j \ln z_j. \quad (3.1)$$

• **Theorem 3.1**

Let P be a positive polytope and let $\mathbf{d} > \mathbf{0}$. Then (3.1) has a unique optimal solution $\mathbf{z}$, which also solves (2.5).

Proof Since P contains a positive vector, a standard compactness argument shows that (3.1) has an optimal solution $\mathbf{z}$, all of whose components are strictly positive. Strict convexity of $-\ln z_j$ shows that $\mathbf{z}$ is unique.

To show that this $\mathbf{z}$ solves (2.5), we select any $\mathbf{x} \in P$ different from $\mathbf{z}$ and perturb $\mathbf{z}$ in the feasible direction $\mathbf{x} - \mathbf{z}$. Specifically, we consider the function $g(t) = F[\mathbf{z} + t(\mathbf{x} - \mathbf{z})]$ for $0 \leq t \leq 1$, with $F(\mathbf{z})$ defined as in (3.1). The function $g(t)$ is convex, differentiable, and nondecreasing in t —the last because $\mathbf{z}$ is optimal. Thus,

$$0 \leq g'(0) = \sum (-d_j/z_j)(x_j - z_j), \text{ for all } \mathbf{x} \in P. \quad (3.2)$$

Expression (3.2) simplifies to $\sum d_j \geq \sum (d_j/z_j) x_j$ for all $\mathbf{x} \in P$, which, when combined with (2.3), yields (2.5).

A more direct proof proceeds as follows:

First we establish uniqueness. Let $\mathbf{z}$ and $\bar{\mathbf{z}}$ be two minimizers in (2.5). Since $\mathbf{z} \in P$, and $\bar{\mathbf{z}}$ is a minimizer in (2.5),

$$\sum \frac{d_j}{\bar{z}_j} \bar{z}_j \leq \sum d_j. \quad (3.3)$$

Similarly,

$$\sum \frac{d_j}{z_j} \bar{z}_j \leq \sum d_j. \quad (3.4)$$

Add the inequalities (3.3) and (3.4). Since each $d_j > 0$ and $z_j/\bar{z}_j + \bar{z}_j/z_j \geq 2$ and we have strict inequality unless $z_j = \bar{z}_j$, the sum of the left sides is at least $2\sum d_j$. The sum is $2\sum d_j$ if and only if $z_j = \bar{z}_j$ for all j . The sum of the right sides is $2\sum d_j$, so $\mathbf{z} = \bar{\mathbf{z}}$.

As noted in (2.3), for any positive $\mathbf{z} \in P$, $\max_{x \in P} \sum (d_j/z_j)x_j \geq \sum d_j$. We give two different proofs of the reverse inequality.

Let $\mathbf{M}$ be a matrix whose rows are the vertices of P , so that P is the convex hull $K(\mathbf{M})$ of the rows of $\mathbf{M}$. We must prove that

$$\text{there exists } \mathbf{z} \in K(\mathbf{M}) \text{ such that } \sum \frac{m_{ij}d_j}{z_j} \leq \sum d_j \text{ for all } i. \quad (3.5)$$

Note that for every $\mathbf{z} \in K(\mathbf{M})$, we have $\sum_j m_{ij}d_j/z_j \geq \sum d_j$ for at least one i . It is convenient to have $\mathbf{M}$ positive. To that end, let $\mathbf{J}$ be the matrix of 1's, $\epsilon > 0$. We prove that there exists $\mathbf{z}(\epsilon) \in K(\mathbf{M} + \epsilon\mathbf{J})$ such that

$$\sum \frac{m_{ij} + \epsilon}{z_j(\epsilon)} \leq \sum d_j \text{ for all } i. \quad (3.6)$$

Now $\mathbf{z}(\epsilon)$ is in a compact region, so there exists a sequence of ϵ 's converging to 0 such that the corresponding $\mathbf{z}(\epsilon)$'s converge to some $\mathbf{z} \in K(\mathbf{M})$. This $\mathbf{z}$ cannot have any coordinate 0; otherwise, since each column of $\mathbf{M}$ contains at least one positive entry (say in row i), (3.6) would be violated for row i and some ϵ . So, returning to (3.5), we assume that all entries in $\mathbf{M}$ are positive.

Here is an elementary proof of (3.5). Let $\mathbf{z}$ be a minimizer in (2.5), and assume that (3.5) is false, so that

$$\max_i \sum_j (d_j/z_j)m_{ij} = D^* > D \equiv \sum d_j.$$

It can be seen that the maximum cannot be attained for all i , so if $I^* \equiv \{i : \sum_j (d_j/z_j)m_{ij} = D^*\}$, then $|I^*| < m$. Now apply induction on the number of rows of $\mathbf{M}$, since (3.5) clearly holds if $\mathbf{M}$ has one row. Let $\mathbf{M}^*$ be the submatrix of $\mathbf{M}$ formed by rows in I^* . Then, by induction, there exists $\bar{\mathbf{z}} \in K(\mathbf{M}^*)$ with $\sum_j m_{ij}d_j/\bar{z}_j \leq D$. If $\epsilon > 0$ is small, then setting $\hat{\mathbf{z}} = \epsilon\bar{\mathbf{z}} + (1 - \epsilon)\mathbf{z}$ yields a value of $\max_i \sum_j m_{ij}d_j/\hat{z}_j$ strictly smaller than D^* , contradicting the definition of $\mathbf{z}$. Here we use the fact that because $\mathbf{M}$ is strictly positive, $\sum_j m_{ij}d_j/w_j$ is a strictly convex function on positive $\mathbf{w}$, smaller when $\mathbf{w} = \bar{\mathbf{z}}$ than when $\mathbf{w} = \mathbf{z}$.

Another proof of (3.5) uses Brouwer's fixed-point theorem. Assume that $\mathbf{M}$ has m rows, and let Λ be the

simplex $\{\lambda : \lambda \geq 0, \sum \lambda_i = 1\}$. With $\mathbf{z} = \lambda\mathbf{M}$ and $D = \sum d_j$, consider the continuous mapping of Λ into itself:

$$\lambda'_i = \frac{\lambda_i + \left(\sum_j \frac{m_{ij}d_j}{z_j} - D \right)}{1 + \sum_k \left(\sum_j \frac{m_{kj}d_j}{z_j} - D \right)},$$

where $a_+ \equiv \max(a, 0)$. Let λ be a fixed point of this map, $\mathbf{z} = \lambda\mathbf{M}$. Then

$$\lambda_i \sum_k \left(\sum_j \frac{m_{kj}d_j}{z_j} - D \right)_+ = \left(\sum_j m_{ij} \frac{d_j}{z_j} - D \right)_+. \quad (3.7)$$

We must show that for every i , the right side of (3.7) is 0. This is surely so if $\lambda_i = 0$. Further, if it is false, the left side of (3.7) would be positive for $\lambda_i > 0$, so that the right side would be positive.

Let $\Lambda^* = \{i : \lambda_i > 0\}$. For each $i \in \Lambda^*$, we would have $\sum_j m_{ij}d_j/z_j > D$, so

$$\sum_{i \in \Lambda^*} \lambda_i \sum_j \frac{m_{ij}d_j}{z_j} > D \sum_{i \in \Lambda^*} \lambda_i = D. \quad (3.8)$$

But (3.8) can be rewritten

$$D < \sum_j \frac{d_j}{z_j} \sum_{i \in \Lambda^*} \lambda_i m_{ij} = \sum_j \frac{d_j}{z_j} \sum_i \lambda_i m_{ij} = \sum d_j = D,$$

which is a contradiction.

4. Pangloss theorem for flat positive polytopes

• *Theorem 4.1 (Pangloss theorem for flat positive polytopes)*

Let P be a flat positive polytope, and let $\mathbf{d} = (d_1, \dots, d_n) > 0$. Then there exists a unique $\mathbf{c} = (c_1, \dots, c_n) > 0$ such that

there exists positive $\mathbf{z} \in P$ such that $c_j z_j = d_j$ for all j

and

$$\text{for all } \mathbf{x} \in P, \sum c_j x_j = \sum d_j = D. \quad (4.1)$$

This theorem asserts that any positive vector $\mathbf{d}$ is both the "best" and "worst" vector of revenues for some linear objective function $\mathbf{c}$, maximized over P .

Let us deduce Theorem 4.1 from Corollary 2.2. By hypothesis, P is flat and $\mathbf{z}$ is positive. It is easy to show that $\mathbf{z}$ can be written as a positive convex combination of all extreme points of P . Hence (2.6) holds for each extreme point of P , and (4.1) follows.

A more insightful proof of (4.1) comes from the concept of entropy (see [3] for other uses of entropy in

combinatorial optimization). First, we present some preliminaries. Let $P \equiv \{x : Ax = b, x \geq 0\}$ be the given flat positive polytope. Let A_j denote the j th column of A . The following optimization problem is motivated by the PERT problem:

$$\begin{aligned} & \text{Minimize } \theta = yb \quad \text{subject to} \\ p : & \quad Az = b \\ & \quad z \geq 0 \\ w_j : & \quad yA_j - d_j/z_j \geq 0 \quad \text{for all } j. \end{aligned} \quad (4.2)$$

Clearly, (4.2) is a convex program. Multipliers p and w are assigned to particular constraints. No multipliers are assigned to the constraints $z \geq 0$, because an optimal solution to (4.2) would have $z > 0$; hence, corresponding multipliers would all be zero. Because we assume $z > 0$, the Karush-Kuhn-Tucker (KKT) optimality conditions for (4.2) are

$$\begin{aligned} & w \geq 0, \\ y : & \quad Aw = b \\ z_j : & \quad pA_j - w_j d_j / (z_j)^2 = 0, \quad \text{for all } j \\ & \quad w_j (d_j / z_j - yA_j) = 0, \quad \text{for all } j. \end{aligned} \quad (4.3)$$

We shall see that an optimal solution to (4.2) and its KKT multipliers can be obtained from the familiar program

$$\min \sum -d_j \ln z_j : Az = b, z \geq 0. \quad (4.4)$$

• Theorem 4.2

Let z be an optimum for (4.4), and let y be its KKT multipliers for the constraints $Az = b$. Then $(y; z)$ is optimal for (4.2); that program's KKT multipliers are $p = y$ and $w = z$; and the optimal value θ^* is $\sum d_j$.

Proof The KKT conditions for (4.4) are

$$d_j/z_j = yA_j \quad \text{for all } j. \quad (4.5)$$

The pair $(y; z)$ satisfies the constraints in (4.2). To satisfy the optimality conditions in (4.3), we take $p = y$ and $w = z$. Finally, we multiply (4.5) by z_j and then sum, to obtain $\sum d_j = yAz = yb = \theta^*$.

Theorem 4.2 establishes a "self-dual" property of (4.2). Its optimal solution and its KKT multipliers equal each other.

5. Relation between the two Pangloss theorems

The contrast between Theorems 2.1 and 4.1 suggests that (4.1) characterizes flat positive polytopes.

• Theorem 5.1

Let P be a positive polytope. Then P is flat if and only if, for each $d > 0$, there exists positive $z \in P$ such that

$$\sum d_j = \sum (d_j/z_j)x_j \text{ for all } x \in P. \quad (5.1)$$

Proof The necessity is just Theorem 4.1. We prove the sufficiency. Let $d > 0$. By hypothesis, there exists $z \in P$ for which (5.1) holds. Let $c_j = d_j/z_j$ for each j . Equation (5.1) becomes $\sum c_j x_j = \sum d_j$ for each $x \in P$. Add sufficiently large multiples of this equation to the equations and inequalities of a minimal representation of P , to cause the representation to have the form $\{x : A_1 x = b^1, A_2 x \leq b^2, x \geq 0\}$, where all coefficients in A_1 and A_2 are positive, and every inequality is essential. Thus, at least one of these inequalities, which we write as $\sum a_j x_j \equiv (a, x) \leq b$, has the following properties:

$$\text{each } a_j > 0; \quad (5.2)$$

$$\text{at least one positive } z \in P \text{ satisfies } (a, z) = b; \quad (5.3)$$

$$\text{at least one } x \in P \text{ satisfies } (a, x) < b. \quad (5.4)$$

Let $d = (a_1 z_1, \dots, a_n z_n)$ and $c = (a_1, \dots, a_n)$. Then, by (5.2) and (5.3), c and z satisfy the Pangloss theorem for positive polytopes, and they are unique. By (5.1), we must have $\sum a_j x_j = b$ for all $x \in P$, which contradicts (5.4). Hence P is flat.

6. Remarks

In closing, we mention three generalizations. First, PERT networks may require "dummy" edges that are not associated with any real task of the project, but are used to impose precedence constraints on the other tasks. A dummy edge e normally has $d_e = 0$. There is no difficulty in extending our previous results to this case, but uniqueness of the optimal solution value z_e for those e with $d_e = 0$ is lost. An alternative to the introduction of dummy edges is to let the nodes of an acyclic graph correspond to the tasks of a project, and to use the edges simply to indicate precedence. Results analogous to those presented in this paper hold in this framework.

Second, we can accommodate "nonconcurrency conditions," that is, requirements that certain pairs of tasks cannot be performed simultaneously, even if neither is a predecessor (direct or indirect) of the other in the acyclic graph. To do so, we consider each such pair in turn, adding an edge from the terminal node of one of the task edges to the initial node of the other if no edges previously added have made either one a predecessor of the other.

Third, our theorems about positive polytopes hold for any compact convex subset P of the nonnegative orthant that contains a positive vector—not just for polytopes.

Nimrod Megiddo has pointed out to us that problem (3.1) is an instance of finding the weighted analytic center of a polytope. (The word "center" is used as it is in barrier methods for linear programming.) Thus, efficient algorithms for finding the center (see [4]) are adaptable.

Acknowledgments

We thank Nimrod Megiddo, Don Coppersmith, Greg Glockner, Rolf Möhring, Michael Powell, David Jensen, Uri Rothblum, and Pete Veinott for helpful discussions during the course of this project.

References

1. J. E. Kelley, Jr., "Critical Path Planning and Scheduling: Mathematical Basis," *Oper. Res.* **9**, 296–320 (1961).
2. C. L. Monma, A. Schrijver, M. J. Todd, and V. K. Wei, "Convex Resource Allocation Problems on Directed Acyclic Graphs: Duality, Complexity, Special Cases, and Extensions," *Math. Oper. Res.* **15**, 736–748 (1990).
3. I. Csizsar, J. Körner, L. Lovász, K. Marton, and G. Simonyi, "Entropy Splitting for Antiblocking Pairs and Perfect Graphs," *Combinatorics* **10**, 27–40 (1990).
4. P. M. Vaidya, "A Locally Well-Behaved Potential Function and a Simple Newton-Type Method for Finding the Center of a Polytope," *Progress in Mathematical Programming*, N. Megiddo, Ed., Springer-Verlag, New York, 1989, pp. 79–90.

Received August 3, 1993; accepted for publication April 11, 1994

Eric V. Denardo *Center for Systems Science, Yale University, P.O. Box 208267, New Haven, Connecticut 06520.* Dr. Denardo has been at Yale University since 1968, in the Department of Administrative Sciences, in the School of Management and in the Department of Operations Research, prior to his present affiliation. He graduated from Princeton University in 1958, with a B.S. degree in engineering, and worked for Western Electric's Engineering Research Center until 1962, primarily on industrial uses of digital computers. From 1962 to 1965, he was a Ph.D. student at Northwestern University and a consultant to the RAND Corporation. At RAND (1965–1968), he worked on dynamic programming and management information systems. Dr. Denardo is perhaps best known for his thesis, papers, and monograph on dynamic programming. His more recent work is on uncertainty in manufacturing and in telecommunications. He has served on the editorial boards of *Management Science* and *Mathematics of Operations Research*.

Alan J. Hoffman *IBM Research Division, Thomas J. Watson Research Center, P.O. Box 218, Yorktown Heights, New York 10598 (HOFFA at YKTVMV, hoffa@watson.ibm.com).* Dr. Hoffman joined IBM in 1961 as a Research Staff Member in the Department of Mathematical Sciences at the IBM Thomas J. Watson Research Center; he was appointed an IBM Fellow in 1977. He received A.B. (1947) and Ph.D. (1950) degrees from Columbia University and worked at the Institute for Advanced Study (Princeton), National Bureau of Standards (Washington), Office of Naval Research (London) and General Electric Company (New York) prior to joining IBM. Dr. Hoffman has been adjunct or visiting professor at various universities and has supervised fifteen doctoral theses in mathematics and operations research. He is currently serving or has served on the editorial boards of *Linear Algebra and Its Applications* (founding editor) and ten other journals in applied mathematics, combinatorics, and operations research. Dr. Hoffman holds an honorary doctorate from the Israel Institute of Technology (Technion); he was a co-winner in 1992 (with Philip Wolfe of the Mathematical Sciences Department) of the von Neumann Prize of the Operations Research Society and the Institute of Management Science.

Todd Mackenzie *Department of Statistics, McGill University, Montreal, Quebec, H3A 2K6 Canada.* Mr. Mackenzie received his B.Sc. degree from Dalhousie University in 1990 and his M.Sc. degree from McGill University in 1993. He has worked as a research assistant in the Division of Clinical Epidemiology, Montreal General Hospital, since 1989 and is currently a Ph.D. student in the Department of Statistics at McGill University.

William R. Pulleyblank *IBM Research Division, Thomas J. Watson Research Center, P.O. Box 218, Yorktown Heights, New York 10598 (WRP at YKTVMV, wrp@watson.ibm.com).* Dr. Pulleyblank was a systems engineer with IBM Canada, Ltd. from 1969 through 1974. During this period, he also completed his doctoral degree at the University of Waterloo. From 1974 to 1990, he was a faculty member, first at the University of Calgary, later at the University of Waterloo. He spent four of these years working at research centers in Belgium, France, and Germany. His main research activities have been in the areas of algorithmic graph theory, combinatorial optimization, and polyhedral combinatorics. He has also worked on various applied problems. In addition to writing a large number of research papers and book chapters, Dr. Pulleyblank is a coauthor of TRAVEL, an interactive, graphics-based system for solving traveling salesman problems. He is currently involved in writing a student textbook on combinatorial optimization. He is Editor-in-Chief of *Mathematical Programming Series B* and also serves on several other editorial boards. Since August of 1990, he has been a member of the Mathematical Sciences Department at the IBM Thomas J. Watson Research Center in Yorktown Heights, NY. Dr. Pulleyblank is currently manager of the Optimization and Statistics Center.