Signature Methods for the Assignment Problem

M. L. BALINSKI

C.N.R.S., Laboratoire d'Econométrie de l'Ecole Polytechnique, Paris

(Received November 1982; accepted March 1984)

The "signature" of a dual feasible basis of the assignment problem is an n-vector whose ith component is the number of nonbasic activities of type (i, j). This paper uses signatures to describe a method for finding optimal assignments that terminates in at most $(n - 1)(n - 2)/2$ pivot steps and takes at most $O(n^3)$ work.

"It is a kind of universal signature by which nature makes known to us the several species of her production."

J. HARRIS (1775).

BELIEVE it or not, yet another approach to the assignment problem! It is a dual simplex method in the following sense: each step goes from one dual feasible basis to a neighboring one. One variant is "purely" dual: it pays no attention to the primal problem, or to a "partial" assignment, so there is no to-do over "breakthrough" versus "nonbreakthrough," or primal change versus dual change. Degeneracy is moot. The bookkeeping is thus simple, and the method easy to describe. At most $(n - 1)(n - 2)/2$ pivots or basis changes are required to solve the problem. It is of geometric although not of practical interest that this is better than the best previously known upper bound of $n(n + 1)/2$ steps, where each "step" represents a change either in the primal or the dual variables (see, e.g., Balinski and Gomory [1964], Munkres [1957]). The approach compares favorably both with Hung's [1983] recent primal simplex method for the assignment problem that generates at most $n^3 \ln \Delta$ bases, where Δ is a constant that depends upon the costs, and with Roohy-Laleh's [1981] primal simplex method that generates at most $O(n^3)$ bases (independent of the costs).

There is, however, a cost to ignoring the primal problem: prior to its termination, the method may encounter a dual feasible basis that already admits an optimal assignment. A bit of additional accounting overcomes this difficulty. Doing it enables one to see that the method is a dual

Subject classification: 432 bipartite matching, 481 algorithm for assignment problem, 637 assignment problem.

Operations Research
Vol. 33, No. 3, May–June 1985

0030-364X/85/3303-0527 $01.25

simplex procedure: the objective of the dual problem is monotone nondecreasing. A stickler for detail might argue with this appellation, in that the variable that is chosen to enter the basis is not necessarily one that has a positive price (note that the dual objective is to be maximized); it may have a zero price. However, many variants of signature methods are possible, and I would expect that this detail can be overcome. A particularly attractive variant uses the fundamental idea inductively, producing an optimal solution to the n-person problem from an optimal solution to the $(n - 1)$-person problem (Goldfarb [1983a]).

The current standard for evaluating an algorithm is a worst case analysis of the number of arithmetic operations necessary to obtain a solution. For the assignment problem $O(n^3)$ comparisons and additions is the best known bound to date (e.g., Edmonds and Karp [1972]). Cunningham [1983] and Goldfarb [1983b] have independently pointed out to me a trick in counting that shows the signature method is $O(n^3)$. It is of interest to note that this is the first instance in which a simplex method has been found to be competitive with nonsimplex methods in terms of worst case analysis. In addition, the count for this *dual* simplex method compares favorably with the $O(n^5)$ bound of Roohy-Laleh's *primal* simplex method.

It is also of interest that, for the assignment problem, the "strongly feasible bases" of Cunningham [1976, 1979] and the "alternating bases" of Barr et al. [1977] are, in my terminology, nothing but bases whose signature is always $(1, 2, 2, \cdots, 2)$, where the 1 corresponds to the "root." Such bases are always primal feasible. Their methods seek a basis that, among all the bases having that signature, is dual feasible. Signature methods seek, among dual feasible bases, one that has that signature. The dual approach is better theoretically, in terms of number of pivots and worst case work (in the present state of knowledge), but as a matter of *practical* computational competitivity nothing definitive is yet known. The simplicity of the geometry of the dual polyhedron in comparison with that of the primal polyhedron (Balinski [1984], Balinski and Russakoff [1984]) leads one to hope that the dual approach may prove to be good in practice as well. Signatures are the key to establishing these properties (Balinski [1983]).

Signature methods—despite this technical introduction—can be explained and shown to work without recourse to duality, extreme points, and all that, so I will argue from first principles, and give a simple, self-contained account.

1. SETTING THE STAGE

An *assignment problem* is specified by an n by n matrix $\mathbf{c} = (c_{ij})$: find a permutation of the column indices σ that minimizes $\sum_i c_{i\sigma(i)}$. Call the set of row indices R and the set of column indices C.

LEMMA 1. *σ solves $\mathbf{c} = (c_{ij})$ if and only if it solves $\mathbf{c}' = (c'_{ij})$, $c'_{ij} = c_{ij} - u_i - v_j$ for any choice of $\mathbf{u} = (u_i)$, $i \in R$, and $\mathbf{v} = (v_j)$, $j \in C$.*

Proof. Exactly one entry from each row and each column of the matrix must be chosen. Therefore, this transformation does not change the relative contributions of entries: it changes only the quantity to be minimized by the constant $\sum_i u_i + \sum_j v_j$.

Consider the following model. Let the set of n nodes R represent the rows of the matrix, the set of n nodes C represent the columns, and let T be any spanning tree of edges (i, j), $i \in R$, $j \in C$ (Figure 1). T must contain exactly $2n - 1$ edges. Given any T, unique values of u_i and v_j that solve the equations $u_i + v_j = c_{ij}$ for $(i, j) \in T$ may be computed as follows:

(i) Set $u_1 = 0$;
(ii) If $(i, j) \in T$ and i has value u_i, define $v_j = c_{ij} - u_i$;
if $(i, j) \in T$ and j has value v_j, define $u_i = c_{ij} - v_j$.

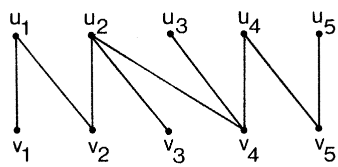

Figure 1. Spanning tree T.

If, in addition, $u_i + v_j \leq c_{ij}$ for $(i, j) \notin T$, then $\mathbf{u}$, $\mathbf{v}$ and its $T = T(\mathbf{u}, \mathbf{v})$ is said to be *feasible.* From now on, every mention of "tree" means "feasible spanning tree."

THEOREM 1. *If $T(\mathbf{u}, \mathbf{v})$ is a tree with some one row node i^* of degree 1 and the remaining rows of degree 2, then the permutation σ defined as follows solves the assignment problem:*

$$\sigma(i^*) = j \quad \textit{for} \quad (i^*, j) \in T(\mathbf{u}, \mathbf{v})$$

$$\sigma(i) = j, \quad i \neq i^*, \quad \textit{for} \quad (i, j) \in T(\mathbf{u}, \mathbf{v}) \textit{ the unique edge incident to } i \textit{ not on the path joining } i \textit{ to } i^*.$$

Proof. $\sigma(i) \neq \sigma(h)$, for otherwise $T(\mathbf{u}, \mathbf{v})$ would contain a cycle; so σ is a permutation. σ solves the problem $\mathbf{c}' = (c'_{ij})$, $c'_{ij} = c_{ij} - u_i - v_j \geq 0$ because *relative to* $\mathbf{c}'$ its cost is 0, whereas any permutation has a nonnegative total cost relative to $\mathbf{c}'$. But by the lemma, this means σ solves $\mathbf{c}$ as well.

The *signature* of a tree T is the vector of its row node degrees $\mathbf{a} = (a_1, \cdots, a_n)$, $\sum_i a_i = 2n - 1$, $a_i \geq 1$. The method seeks a tree whose signature contains exactly one 1 and otherwise 2's. It iterates from one tree T to another "neighboring" tree T', obtained by *pivoting* on an edge $(k, l) \in T$ (Figure 2): Given $(k, l) \in T(u, v)$, both k and l having degree at least 2, drop (k, l). This cuts T into two distinct components: T^k, which contains $k \in R$, and T^l, which contains $l \in C$. Let

$$\delta = \min\{c_{ij} - u_i - v_j;\ i \in T^l, j \in T^k\} \geq 0,$$

and (g, h) be some pair at which the minimum is achieved. Define

$$T' = T^k \cup T^l \cup (g, h) \quad ((g, h) \text{ is the "incoming" edge}) \quad \text{and,}$$

$$u_i' = u_i + \delta, \quad i \in T^l, \quad u_i' = u_i \quad \text{otherwise,}$$

$$v_j' = v_j - \delta, \quad j \in T^l, \quad v_j' = v_j \quad \text{otherwise.}$$

$\delta \geq 0$ because T is feasible; and the choice of δ guarantees that T' is feasible as well. The signature $\mathbf{a}'$ of T' is the same as that of T except that $a_k' = a_k - 1$ and $a_g' = a_g + 1$.

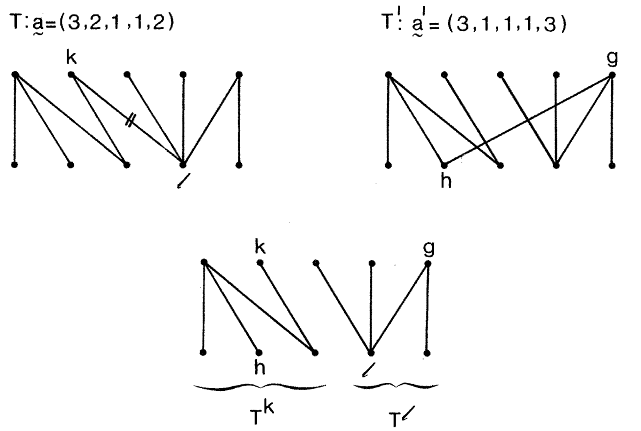

Figure 2. Pivoting from T to T'.

2. THE PURELY DUAL METHOD

Assume that every assignment $i \in R$ to $j \in C$ is admissable. If this is not the case (i.e., if there is "sparsity"), then for the purposes of this exposition make any inadmissible assignment admissable, but at a prohibitive cost M. (The algorithm can be modified to treat sparse networks more efficiently.)

The method is entirely guided by the signatures. It begins at "level $n - 1$" and ends when "level 1" is reached. A tree is in *level k* if its signature has exactly k 1's.

The initial tree $T(\mathbf{u}, \mathbf{v})$ has signature $(n, 1, \cdots, 1)$. It is

$$u_1 = 0; \quad v_j = c_{1j}, \quad j \in C \quad \text{and} \quad (1, j) \in T \quad \text{for every} \quad j \in C;$$

and

$$u_i = \min_j(c_{ij} - c_{1j}), \quad 1 \neq i \in R \quad \text{and} \quad (i, j) \in T \quad \text{for one } j \text{ that gives the minimum.}$$

The *first level* k tree T encountered has (by construction) the form $(k + 1, 2, \cdots, 2, 1, \cdots, 1)$: the source row node s has degree $k + 1$, some k row nodes have degree 1, and the remaining $n - k - 1$ row nodes have degree 2 (Figure 3). Single out some row node of degree 1 and designate it as the *target* t for level k. *Pivot* on the edge (s, l) of the path joining the source s to the target t. Call Q the set of row nodes of the component of $T \sim (s, l)$ that contains t, and T' the new tree: $s \notin Q$ and the degree of some node $s' \in Q$ is increased by 1. If (i) s' was of degree 1 in T, level

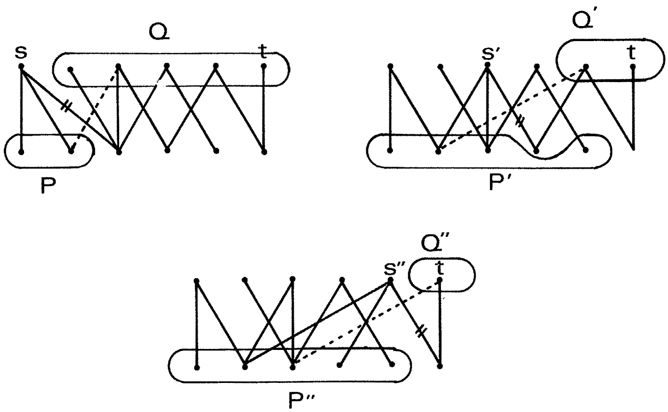

Figure 3. First three level 2 trees. (Source nodes s, target t, broken edges = incoming edges.) One more pivot terminates the procedure.

$k - 1$ has been reached. If (ii) s' was of degree 2 in T, take s' as the source in T' and repeat, pivoting on (s', l') of the path joining s' to t. Call Q' the set of row nodes of the component $T' \sim (s', l')$ that contains t, and T'' the new tree. $Q' \subset Q$ and $s' \notin Q'$. Therefore, case (ii) can occur at most $n - k - 1$ times before a case (i) occurs, so the method encounters at most $n - k$ level k trees. Another way of giving the argument is by studying the changes in P, the set of column nodes of the component of the tree that does not contain the target t. P must grow at each step until the incoming edge encounters a row node of degree 1.

THEOREM 2. *The method terminates in at most* $(n - 1)(n - 2)/2$ *steps.*

Proof. The method encounters at most 1 level $n - 1$ tree; at most 2 level $n - 2$ trees; $\cdots$; at most $n - 2$ level 2 trees.

The obvious way of obtaining a worst case count is to notice that each pivot step involves at most $O(n^2)$ operations, since at most n^2 comparisons are needed to find the value of δ. This count yields $O(n^4)$ for the method. The much less obvious way pointed out to me by Cunningham [1983] and Goldfarb [1983b] is to notice that at most n^2 comparisons are needed to compute the δ's for all of the pivots of any one level. To see this, suppose a pivot with change $\delta \geq 0$ is made from one tree to another within a level, with $(\mathbf{u}, \mathbf{v})$ transformed to $(\mathbf{u}', \mathbf{v}')$. Let Q be the set of row nodes of the component (with pivot edge removed) containing the target t in the first tree and Q' the corresponding set at the next tree, P the set of column nodes of the component not containing t in the first tree, and P' the corresponding set at the next tree. By construction, $Q' \subset Q$ and $P \subset P'$. Since $u_i' = u_i + \delta$ for *every* $i \in Q'$, an edge at which $\min\{c_{ij} - u_i'; j \in P\}$ is achieved does not change, and the value of the minimum is decreased by δ for every $i \in Q'$. The name of the column $j \in P$ at which the minimum is achieved and its value can therefore be stored "at" the node $i \in Q'$ and the only extra comparisons that are necessary to find the new minima for $i \in Q'$ over all of P' is to compute $\min\{c_{ij} - u_i'; i \in Q')$ for each $j \in P' \sim P$. This can of course result in replacing a name and value at some $i \in Q'$. But it means that column node j needs to be "scanned" only once within a level, and, therefore, at most n^2 comparisons are necessary per level. Since there are at most $n - 1$ levels, this count yields $O(n^3)$.

THEOREM 3. *The method requires at most $O(n^3)$ comparisons and additions.*

Variants of this simple method can be given. For instance, some different initial tree might be at hand (due, for example, to sparsity). Suppose the signature is level k. If $k = 1$, it is a solution. If $k > 1$, choose any row node i with degree greater than 2 as the source, any node of degree 1 as target, and apply the method without changing the target node until a tree of level $k - 1$ obtains. This transition to a new level must happen in at most $n - k$ steps, and so, in at most $\sum_k^{n-2} (n - j)$ steps, a solution is found.

Example

Set $u_1 = 0$, $\mathbf{v} = (14, 18, 15, 10, 10)$ and compute $u_i = \min_j(c_{ij} - v_j)$ giving: $\mathbf{u} = (0, -2, -2, -8, -3)$.

$$\mathbf{c} = \begin{array}{|ccccc|} 14 & 18 & 15 & 10 & 10 \\ 18 & 17 & 15 & 8 & 8 \\ 16 & 16 & 24 & 25 & 12 \\ 19 & 10 & 8 & 14 & 11 \\ 22 & 15 & 28 & 24 & 12 \end{array}$$

In Figure 4, the values of $\mathbf{u}$, $\mathbf{v}$ are attached to each node. $(i, j) \in T$ implies $u_i + v_j = c_{ij}$. The source node is indicated by a circle, the target node by a triangle. The pivot edge (k, l) is distinguished by being crossed by two lines. The $\pm\delta$'s single out the subtree T^l whose u's and v's are changed in pivoting. The dashed line is the new incoming edge. In the final optimal tree, the heavy lines single out an optimal assignment.

This problem required 4 pivots; the upper bound is 6. One could, of course, use column signatures instead of row signatures. In this case, the initial column signature is (1, 4, 1, 2, 1); consequently, at most 5 steps are necessary to terminate, using the column signatures as the guide.

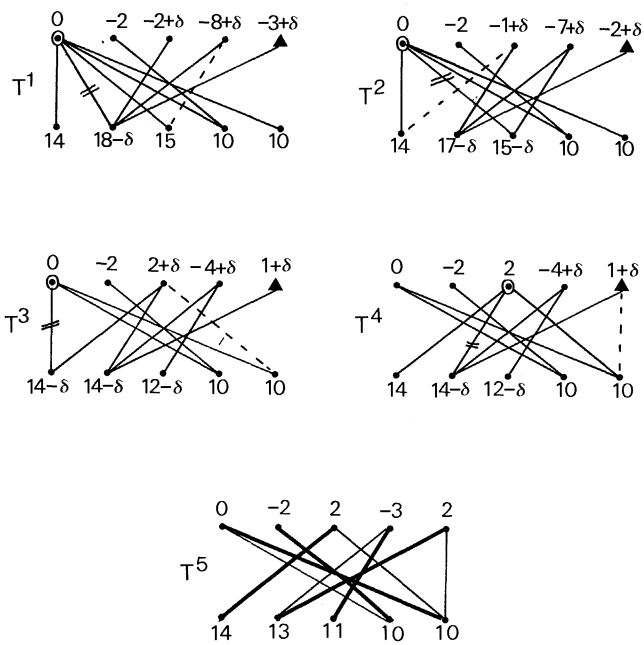

Figure 4. Successive trees of the algorithm for the example.

3. ADJOINING PRIMAL ACCOUNTING

To adjoin primal information to the dual solutions determined by the method, describe the possible permutations in the form:

$$P = \{\mathbf{x};\ \textstyle\sum_i x_{ij} = 1, \sum_j x_{ij} = 1, x_{ij} \geq 0 \text{ integer } i \in R, j \in C\}.$$

Any $\mathbf{x} \in P$ defines a permutation σ by the following rule: $x_{ij} = 1$ implies $\sigma(i) = j$.

Let $\bar{P}$ denote P without the constraints $x_{ij} \geq 0$. A tree T determines a unique solution to $\{\mathbf{x} \in \bar{P};\ x_{ij} = 0 \text{ for } (i, j) \notin T\}$ that can be found recursively. To each T of the purely dual method, adjoin its associated $\mathbf{x}$. If $\mathbf{x} \geq 0$, it is an optimal assignment.

Suppose $\mathbf{x}$ is associated with T and a *pivot* is made on (k, l) to obtain T' with incoming edge (g, h). $T \cup (g, h)$ determines a unique cycle that includes (k, l) and (g, h). If (k, l) is called odd, the next edge of the cycle even, and so forth, then (g, h) is odd. $\mathbf{x}'$ associated with T' is

$$x'_{ij} = x_{ij} - x_{kl} \quad \text{for} \quad (i, j) \quad \text{in the cycle and odd;}$$

$$x'_{ij} = x_{ij} + x_{kl} \quad \text{for} \quad (i, j) \quad \text{in the cycle and even; and}$$

$$x'_{ij} = x_{ij} \quad \text{otherwise.}$$

Every pivot edge (k, l) of the algorithm is chosen so that the row nodes of the component T^l of $T \sim (k, l)$ that contains l all have degree 1 or 2. Any algorithm that adheres to this choice (which depends entirely on the signatures) pivots on an edge (k, l) with associated $x_{kl} \leq 0$. For, suppose T^l has $n_1 \geq 1$ nodes of degree 1 and n_2 of degree 2. Then it has $n_1 + n_2$

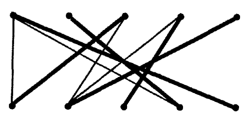

Figure 5. Optimal tree for the example.

row nodes, $n_1 + 2n_2$ edges and so $n_2 + 1$ column nodes. Now, if $x_{kl} \geq 1$—contrary to what is asserted—then,

$$n_1 + n_2 = \{\textstyle\sum x_{ij};\ (i, j) \in T^l\} \leq n_2.$$

The equation comes from summing the x_{ij} over the row nodes, the inequality from summing them over the column nodes. But this outcome contradicts $n_1 \geq 1$, and shows that every pivot is a dual simplex method choice (admittedly of a special kind): the "activity" (k, l) that leaves the dual basis has a nonpositive value $x_{kl} \leq 0$.

The example in Figure 4 is solved in only two steps when the method incorporates primal accounting. Tree T^3 contains an optimal solution (see Figure 5, where heavy edges denote $x_{ij} = 1$; otherwise $x_{ij} = 0$) even though neither the row nor the column signature is of the sought-for form.

An alternative to doing the $\mathbf{x}$-bookkeeping at each step is to check only the first level k tree for each k. Suppose T has signature $(k + 1, 2, \cdots, 2, 1, \cdots, 1)$, with node s of degree $k + 1$ and associated $\mathbf{x}$. If $x_{sj} \geq 0$ for all j, then $\mathbf{x}$ is optimal. For, let (s, h) be any edge with $x_{sh} = 0$ and consider the component T^h of $T \sim (s, h)$ that contains h. Each of the row nodes of T^h has degree 1 or 2. By analysis similar to that in the preceding paragraph, one can conclude that T^h has exactly one node of degree 1. Lemma 1 then shows all x's of T^h must be 0's and 1's.

THEOREM 4. *$(n-1)(n-2)/2$ is the best possible bound.*

Proof. There is at least one assignment problem $\mathbf{c} = (c_{ij})$ for which the algorithm, beginning at a level $n-1$ tree, necessarily takes this number of pivot steps before finding a tree that admits an assignment: namely, $c_{ij} = (m-i)(j-1)$. For, in this case, if $i < h$ and $j < k$, it is impossible for both (i, k) *and* (h, j) to be in a tree T (Figure 6). If both did belong, then $u_i + v_k = (m-i)(k-1)$ and $u_h + v_j = (m-h)(j-1)$, and also $u_i + v_j \leq (m-i)(j-1)$ and $u_h + v_k \leq (m-h)(k-1)$. Therefore,

$$(m-i)(k-1) + (m-h)(j-1) \leq (m-i)(j-1) + (m-h)(k-1),$$

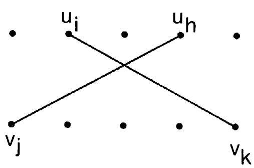

Figure 6. A "crossing" $(i < h, j < k)$.

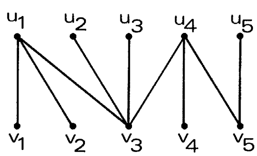

Figure 7. A "no crossing" tree.

a contradiction. It follows that the trees for this problem are only those that have "no crossings" (Figure 7). Therefore, each pivot from a tree with signature $\mathbf{a}$ to one with signature $\mathbf{a}'$ is an "adjacent transfer": $a_k' = a_k - 1$ implies either $a_{k+1}' = a_{k+1} + 1$ or $a_{k-1}' = a_{k-1} + 1$. The tree "closest" to $(n, 1, 1, \cdots, 1)$ that admits an assignment is $(2, 2, \cdots, 2, 1)$. Its distance is $(n-1)(n-2)/2$, since this many adjacent transfers are necessary. Any nearer tree does not admit an assignment: T having signature $\mathbf{a}$ with $a_1 \geq 3$ does not; $a_1 = \cdots = a_k = 2$ and $a_{k+1} \geq 3$ does not.

4. REMARKS

The idea for this approach via signatures came from work on dual transportation polyhedra:

$$D_{m,n}(\mathbf{c}) = \{\mathbf{u}, \mathbf{v}; u_i + v_j \leq c_{ij}, u_1 = 0\}.$$

Signatures—here m partitions (or n partitions) of $m + n - 1$—uniquely characterize the feasible bases of nondegenerate $D_{m,n}(\mathbf{c})$. This fact immediately implies that any $D_{m,n}(\mathbf{c})$ has at most $\binom{m+n-2}{m-1}$ extreme points; less directly, it implies that all nondegenerate $D_{m,n}(\mathbf{c})$ have exactly the same number of faces of all dimensions (Balinski and Russakoff).

Every problem $\mathbf{c}$ has *at least n* trees T that admit an optimal assignment, one for each possible level 1 tree. The following three problems each have six trees in all. All six trees of $\mathbf{c}^1$ admit optimal assignments; four of $\mathbf{c}^2$ do; and three of $\mathbf{c}^3$. It seems that almost anything can happen.

$$\mathbf{c}^1 = \begin{array}{|ccc|} \hline 3 & 3 & 6 \\ 4 & 1 & 6 \\ 4 & 2 & 3 \\ \hline \end{array} \qquad \mathbf{c}^2 = \begin{array}{|ccc|} \hline 3 & 3 & 6 \\ 3 & 2 & 4 \\ 4 & 2 & 3 \\ \hline \end{array} \qquad \mathbf{c}^3 = \begin{array}{|ccc|} \hline 3 & 4 & 4 \\ 3 & 2 & 5 \\ 4 & 2 & 3 \\ \hline \end{array}$$

The signature approach may also be used to show that the Hirsch conjecture is true for any $D_{m,n}(\mathbf{c})$. That is, the diameter of the graph consisting of the extreme points and edges of the unbounded polyhedron $D_{m,n}(\mathbf{c})$ is at most $(m-1)(n-1)$ (Balinski [1984]).

REFERENCES

BALINSKI, M. L. 1983. Signatures des points extrêmes du polyèdre dual du problème de transport. *Compt. Rend. Acad. Sci. Paris*, **296**, Ser. I, 456–459.

BALINSKI, M. L. 1984. The Hirsch Conjecture for Dual Transportation Polyhedra. *Math. Opns. Res.* **9**, 629–633.

BALINSKI, M. L., AND R. E. GOMORY. 1964. A Primal Method for the Assignment and Transportation Problems. *Mgmt. Sci.* **10**, 578–593.

BALINSKI, M. L., AND A. RUSSAKOFF. 1984. Faces of Dual Transportation Polyhedra. *Math. Program. Stud.* **22**, Mathematical Programming at Oberwolfach II, B. Korte and K. Ritter, eds., 1–8.

BARR, R., F. GLOVER, AND D. KLINGMAN. 1977. The Alternating Path Basis Algorithm for Assignment Problems. *Math. Program.* **13**, 1–13.

CUNNINGHAM, W. H. 1976. A Network Simplex Method. *Math. Program.* **11**, 105–116.

CUNNINGHAM, W. H. 1979. Theoretical Properties of the Network Simplex Methods. *Math. Opns. Res.* **4**, 196–208.

CUNNINGHAM, W. H. 1983. Private letter of Sept. 19. (I am indebted also for several suggestions concerning the manuscript in addition to the method of counting.)

EDMONDS, J., AND R. M. KARP. 1972. Theoretical Improvements in Algorithmic Efficiency for Network flow Problems. *J. Assoc. Comput. Mach.* **19**, 248–264.

GOLDFARB, D. 1983a. Efficient Dual Simplex Algorithms for the Assignment Problem (preprint, July).

GOLDFARB, D. 1983b. Private communication (October).

HUNG, M. S. 1983. A Polynomial Simplex Method for the Assignment Problem. *Opns. Res.* **31**, 595–600.

MUNKRES, J. 1957. Algorithms for the Assignment and Transportation Problems. *J. Soc. Indust. Appl.* **5**, 32–38.

ROOHY-LALEH, E. 1981. Improvements to the Theoretical Efficiency of the Network Simplex Method. Ph.D. thesis, Carleton University.