CENTRE DE RECHERCHE SUR LES TRANSPORTS
(C.R.T.)
CENTRE FOR RESEARCH ON TRANSPORTATION

UNIVERSITÉ DE MONTRÉAL
ÉCOLE DES HAUTES ÉTUDES COMMERCIALES
ÉCOLE POLYTECHNIQUE

CRT-966

A Simplex Solution to the Minimum Cost Nonsimultaneous Multi-Commodity Flow Problem

by

Patrick Rioux
Benjamin T. Smith[1]
Krishnaiyan Thulasiraman[2]

CENTRE DE RECHERCHE SUR LES TRANSPORTS
UNIVERSITÉ DE MONTRÉAL
C.P. 6128, SUCCURSALE A
MONTRÉAL
CANADA H3C 3J7

TÉLÉPHONE : (514) 343-7575
EMAIL : crt@crt.umontreal.ca
TÉLÉCOPIEUR (FAX) : (514) 343-7121

CRT-966

A Simplex Solution to the Minimum Cost Nonsimultaneous Multi-Commodity Flow Problem

by

Patrick Rioux
Benjamin T. Smith[1]
Krishnaiyan Thulasiraman[2]

[1] École Polytechnique and Centre de recherche sur les transports, Université de Montréal, C.P. 6128, succursale centre-ville, Montréal (Québec), Canada, H3C 3J7.
[2] Department of Electrical and Computer Engineering, Concordia University, 1455 bd de Maisonneuve Ouest, Montréal (Québec), Canada, H3G 1M8.

Centre de recherche sur les transports — Publication #966

February 1994

Dépôt légal – Bibliothèque nationale du Québec,
Bibliothèque nationale du Canada, 1994

ABSTRACT

In the planning of dynamic architecture telecommunications networks, considerations of sizing of the links lead to a multi-commodity, multi-period flow problem on the underlying graph. A formulation based on the edge-chain representation of the network reveals a structure which can be exploited in two steps. First the generalized upper bounding (GUB) technique is used to take advantage of the multi-commodity nature of the problem by reduction to a working basis. Then the multi-period aspect of the problem is further exploited by a decomposed triangular factorization of the working basis. In each step of the algorithm the problem is decomposed into as many subproblems as there are periods, where each subproblem has size equivalent to that of a single period problem.

Keywords: telecommunications, nonsimultaneous multi-commodity flows, GUB decomposition, working basis.

RESUME

Lors de la planification d'un réseau de télécommunications à architecture dynamique, la considération de la capacité des liens conduit à un problème de multiflot à plusieurs périodes sur le graphe sous-jacent. Une formulation basée sur la représentation arête-chaîne du réseau révèle une structure qui peut être exploitée en deux étapes. D'abord, la technique de décomposition GUB (Generalized Upper Bounds) est utilisée pour profiter du caractère multiflot du problème par la réduction à une base de travail. Ensuite l'aspect multi-période du problème est encore exploité par une factorisation triangulaire décomposée de la base de travail. A chaque étape de l'algorithme le problème est décomposé en autant de sous-problèmes qu'il y a de périodes, où chaque sous-problème a une taille équivalente à celle d'un problème à une période.

Mots-clés : Télécommunications, multiflots non-simultanés, décomposition par bornes généralisées, base de travail.

1. INTRODUCTION

The introduction of digital computers to the technology of telecommunications networks has made possible the dynamic reconfiguration of the architecture of these networks. This means that the topology of the network can be changed over time in order to satisfy known point-to-point demands which vary over a number of periods.

In order to satisfy all demands in all periods, one must allocate enough capacity to each link of the underlying graph so that the maximum number of circuits over all periods routed along a given edge can be accomodated. Since the capacity of an edge is in fixed proportion to the maximum number of circuits using the edge, we may identify the capacity of an edge with this number.

The telecommunications network considered here may be represented as a non-directed graph where the nodes correspond to the switches and the edges correspond to the possibility of establishing a physical link between two switches for the purpose of creating a circuit between them. Circuit demands between any two switches i and j can be thought of as integral flows of a commodity in the graph which may use any one or several of the chains connecting nodes i and j in the graph.

If we assume that the only relevant cost involved in such a network is a non-negative linear cost g_u applied to the maximum flow in any edge u, then the problem is to choose patterns of flow for the circuit demands between all origin-destination (OD) pairs over all periods such that the total cost of capacity is minimized. Minoux(1989) has referred to this problem as a network flow model with nonsimultaneous multi-commodity flow requirements, as the demands for each period are satisfied independently of the demands for all other periods.

If there were just a single period, then the optimal solution would be achieved by routing the demand for each OD pair i and j along a minimum chain connecting i and j in the graph with lengths of edges given by the g_u (it is assumed that the underlying graph is connected, but not necessarily complete).

When there are several time periods of OD pair demands, owing to different busy periods during a day or to time zone changes for example, then the linear cost g_u as defined above will be applied to the maximum flow over all time periods in edge u. In this case, using the minimum chain solution for each period before calculating costs is no longer optimal as the following two-period example shows.

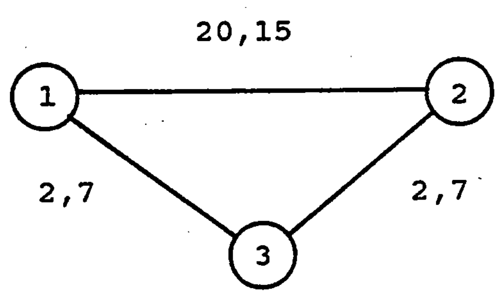

Figure 1. A three node, two period example.

Figure 1 shows a three node graph with demands for periods one and two, respectively, between nodes i and j indicated next to the edge (i,j) and cost g_u = 1 for all u. If the shortest chain solution is used for each period and then the maximum flow is chosen in each edge, then there is a total cost of

$$20 + 7 + 7 = 34.$$

However, if 5 circuits are reassigned from chain (1,2) to the chain (1,3), (3,2), then the total cost becomes

$$15 + 7 + 7 = 29,$$

a savings of 14.7%.

Other authors have considered variants of this problem. McCallum (1977) considers the single period problem where the capacities are fixed quantities and there is a cost of constructing additional capacity plus a non-zero cost for the routing of circuits along chains in the objective function. His approach uses the special structure of the linear programmming formulation of the problem to apply the generalized upper bounding technique (GUB) of Dantzig and Van Slyke (1967), thereby reducing the dimension of the problem to that of a smaller "working basis".

Kortanek and Polak (1985) address the multiperiod problem which they call the Deterministic Dynamic Routing problem (DDR). Their problem is slightly more general, however, as they consider flow through nodes as well as edges via the node-chain incidence matrix and add a cost for this flow in the objective function. The main difference between their problem and the present one is in the nature of the demands. The present demands are constants, whereas theirs are random variables. They proceed to solve the Stochastic Dynamic Routing problem (SDR) by Dantzig-Wolfe decomposition.

The only authors to consider explicitly the same problem as ours are Minoux and Serreault (1981). Their problem formulation was different in that the topology of the underlying graph as well as the demands were allowed to vary from period to period. It will be noted that the method proposed in this paper applies without change to the case of varying topology. Minoux and Serreault chose not to solve the problem directly by the simplex algorithm, but to solve an equivalent problem using Lagrangean relaxation and subgradient algorithms to optimize the dual problems. Thus their method, while computationally efficient, is not exact.

Finally, the problem presented in this paper can be considered as a multi-commodity generalization of the network synthesis problem studied by Gomory and Hu (1961).

This paper is organized as follows. Section 2 formulates the problem as a large-scale linear program and proves a proposition which will provide useful information for the subsequent decomposition of the problem. Section 3 briefly reviews the GUB procedure, and then shows how GUB applies to the problem. In section 4 it is shown that the working basis of GUB can be further decomposed in the light of the proposition of section 2. Section 5 shows in detail how the method is adapted to each step of the revised simplex. A conclusion dealing with the complexity of the proposed method and possible extensions is the subject of section 6.

2. FORMULATION

Let G, the underlying graph of the telecommunications network, be an undirected graph on N nodes having M edges and let n be the number of chains connecting the $K = N(N-1)/2$ distinct Origin- Destination (OD) pairs in G. The problem, as defined in Section 1, is a multiperiod (nonsimultaneous), multi-commodity flow problem. More precisely, we seek chain flows f^h exactly satisfying demands between all OD pairs in H periods, such that a non-negative weighted sum of the maximum flow on each edge over all periods is minimized. This problem may be formulated as

$$\min g^T c$$

$$\left.\begin{array}{ll} Af^h & \le c \\ Ef^h & = d^h \end{array}\right\} \quad \text{for } h = 1,\ldots,H \qquad (1)$$

$$f_h \ge 0 \text{ and integer.}$$

where A is the Mxn edge-chain matrix of G, E is a Kxn matrix of ones and zeroes in staircase form, c is the vector of capacity variables, g^T denotes the transpose of the column vector g, and d^h is the Kx1 column vector of OD demands for period h.

Although the problem as stated is an integer LP, we choose to ignore the integrality constraint for the following reason. The problem is known as the "sizing" problem in the telecommunications literature (see Rioux(1988)). This means that the optimal capacity variable values will constitute benchmark values for subsequent routing algorithms which will use other optimality criteria, and hence that their optimal integer values may be approximated by their optimal LP values rounded up.

After addition of slack variables the LP corresponding to (1) may be written as

$$\min \sum_{i=1}^{M} g_i c_i$$

$$\sum_{k=1}^{K} \sum_{j\in k} a_{ij} f_{kj}{}^{h} - c_i + s_i{}^{h} = 0 \qquad \text{for } i = 1,\ldots,M \quad \text{for } h = 1,\ldots,H$$

$$\sum_{j\in k} f_{kj}{}^{h} = d_k{}^{h} \qquad \text{for } h = 1,\ldots,H \quad \text{for } k = 1,\ldots,K \qquad (2)$$

$$f_{kj}{}^{h}, s_i{}^{h} \geq 0, \quad \text{for all } i,j,h,k$$

where $A = (a_{ij})$ and k' is the set of flow indices corresponding to chains connecting OD pair k in G. $f_{kj}{}^{h}$, c_i, and $s_i{}^{h}$ are referred to as the flow, capacity, and slack variables respectively.

The solution presented in Section 1 constitutes an initial basic feasible solution to (1) and (2) and consists in sending the demand for each OD pair k in each period along the shortest chain, say j(k), in G (with respect to the lengths g_i on the edges). This gives HK flow variables in the basis. The other HM variables are the M capacity variables whose values are

$$c_i = \max_{h=1\ldots H} \{ \sum_{k=1}^{K} a_{ij(k)} f_{kj(k)}{}^{h} \} = \sum_{k=1}^{K} a_{ij(k)} f_{kj(k)}{}^{h_i}, \quad i = 1,\ldots,M$$

where h_i is the period supplying the maximum flow on edge i, and

(H-1)M of the slack variables whose values are

$$s_i{}^h = c_i - \sum_{k=1}^{K} a_{ij(k)} f_{kj(k)}{}^h \qquad \begin{array}{l} i = 1,\ldots,M \\ h = 1,\ldots,H \end{array}$$

Since the $s_i{}^{hi} = 0$, $i= 1,\ldots,M$, these slack variables are declared non-basic.

Note that the above basic feasible solution is optimal to (1) in two special cases: the single period case ($H = 1$) and the multiple period case with a dominant period h, that is, $d_k{}^h \geq d_k{}^r$ for all k and $r = 1,\ldots,H$, $r \neq h$. For multiple and undominated periods, however, problem (1) is non-trivial.

The above construction has shown that there is an initial basic feasible solution containing all the capacity variables. In fact, the capacity variables may be left in the basis as the following proposition shows.

<u>Proposition</u>: At each iteration of the simplex method, if a capacity variable is eligible to leave the basis, then so is a flow or slack variable.

Proof: First, suppose a basis B of (2) contains capacity variable c_i at a positive level. Then according to the H equations

$$\sum_{k=1}^{K} \sum_{j \in k'} a_{ij} f_{kj}{}^h - c_i + s_i{}^h = 0, \qquad h = 1,\ldots,H$$

at least one of the $f_{kj}{}^h$ or $s_i{}^h$ must also be in the basis at a non-zero level for each period h. Hence, if c_i were eligible to leave the basis and become zero, then so would be these slack and flow variables for each period h.

Suppose now that the basis contains capacity variable c_i at a zero level and that c_i is the r^{th} basic variable. Suppose also that non-basic column a_t has been chosen to enter the basis and that y_t is the transformed column, that is,

$$a_t = By_t = \sum_{j=1}^{HM+HK} y_{jt} b_j ,$$

where the b_j are the columns of the basis matrix B. Write the p^{th} equation in this expression for $p = i, i + M, \ldots, i + (H-1)M$, and observe that

$$a_{pt} = -y_{rt} + \sum_{\substack{j=1 \\ j \neq r}}^{HM+HK} y_{jt} b_{pj} .$$

It must be shown that there exists at least one $y_{jt} > 0$ for a flow variable or a slack variable corresponding to c_i. Now $a_{pt} \geq 0$ and $y_{rt} > 0$ since c_i is eligible to leave the basis. Therefore,

$$\sum_{\substack{j=1 \\ j \neq r}}^{HM+HK} y_{jt} b_{pj} > 0,$$

and since all $b_{pj} \geq 0$, we have that $y_{jt} > 0$ and $b_{pj} > 0$ for at least one j corresponding to a flow variable or slack variable. This slack or flow variable whose value in the basis is necessarily zero may then be chosen to leave tha basis in the place of c_i ∎ .

Next the variables may be scaled by dividing each demand constraint by the corresponding demand. This gives

$$\min \sum_{i=1}^{M} g_i c_i$$

$$\sum_{k=1}^{K} \sum_{j \in k'} d_k{}^h a_{ij} \bar{f}_{kj}{}^h - c_i + s_i{}^h = 0 \qquad \begin{array}{l} \text{for } i = 1,\ldots,M \\ \text{for } h = 1,\ldots,H \end{array}$$

$$\sum_{j \in k'} \bar{f}_{kj}{}^h = 1 \qquad \begin{array}{l} \text{for } h = 1,\ldots,H \\ \text{for } k = 1,\ldots,K \end{array} \tag{3}$$

$$\bar{f}_{kj}{}^h,\ s_i{}^h \geq 0, \quad \text{for all } i,j,h,k$$

where $\bar{f}_{kj}{}^h = f_{kj}{}^h / d_k{}^h$ are the scaled variables.

The revised simplex is carried out on (3), but first note that its structure is amenable to the generalized upper bounding technique of Dantzig and Van Slyke (1967). The following approach to GUB resembles that of Lasdon (1972) except that it works with the triangularized form of a certain basis matrix and not its inverse.

3. GENERALIZED UPPER BOUND (GUB) DECOMPOSITION.

Any basis of the system (3) must include at least one variable ${f_{kj}}^h$ for each h and k. Choose one of these variables for each pair h and k and call it a <u>key variable</u>. Then a basis B of the full system (3) takes the form

$$B_{(HM+HK)x(HM+HK)} = \begin{bmatrix} \text{key columns} & \text{non-key columns} \\ \bar{A}_{HMxHK} & \bar{B}_{HMxHM} \\ I_{HKxHK} & C_{HKxHM} \end{bmatrix},$$

B can be made upper block triangular by subtracting suitable key columns from the non-key columns so as to reduce the matrix C to a matrix of zeroes. This subtraction corresponds to multiplying B on the right by the matrix

$$T = \begin{bmatrix} I_{HKxHK} & -C_{HKxHM} \\ O_{HMxHK} & I_{HMxHM} \end{bmatrix}$$

giving

$$B' = BT = \begin{bmatrix} \bar{A}_{HMxHK} & WB_{HMxHM} \\ I_{HKxHK} & O_{HKxHM} \end{bmatrix},$$

where the non-singular HMxHM matrix $WB = -\bar{A}C + \bar{B}$ is called the <u>working basis</u>.

The revised simplex solves two systems involving the basis matrix:

(i) the representation of the entering column in terms of the current basis, and

(ii) the computation of the dual variables.

We show now how GUB as applied to the problem of section 1 may be used to simplify the calculations involved in solving these systems.

(i) If a is a non-basic column eligible to enter the basis, then we must solve By = a. Now T is non-singular since det T = 1; therefore, if $T^{-1}y = z$, then the system By = a is equivalent to BTz = B'z = a, which is solved by

$$z_i = a_{HM+i} \qquad i=1,\ldots,HK$$

$$\text{and} \quad WB\begin{bmatrix} z_{HK+1} \\ \cdot \\ \cdot \\ z_{HK+HM} \end{bmatrix} = \bar{a} - \sum_{i=1}^{HK} z_i \bar{a}_i, \tag{4}$$

where the $\bar{a}_i$ and $\bar{a}$ are the vectors of the first HM components of key column a_i and a respectively.

(ii) If π is the vector of dual variables and δ_B the vector of basic costs, then we must solve

$$\pi^T B = \delta_B{}^T. \tag{5}$$

Now, $\delta_B = (\delta_{B1}, \delta_{B2})$ where $\delta_{B1} = 0$ is an HK+(H-1)M-vector and $\delta_{B2} = g$ is an M-vector (since we may suppose that the capacity variables constitute the last M basic variables according to the proposition); hence (5) is equivalent to

$$\pi^T BT = \delta_B{}^T T.$$

Now,
$$\pi^T BT = \pi^T B' = (\pi_1{}^T, \pi_2{}^T)\begin{bmatrix} \overline{A} & WB \\ I & 0 \end{bmatrix}$$

$$= \delta_B{}^T T = (\delta_{B1}{}^T, \delta_{B2}{}^T)\begin{bmatrix} I & -C \\ 0 & I \end{bmatrix}$$

$$= (\delta_{B1}{}^T, \delta_{B2}{}^T - \delta_{B1}{}^T C)$$

$$= (0, \delta_{B2}{}^T).$$

Hence, $\pi_1{}^T \overline{A} + \pi_2{}^T = 0$, or

$$\pi_2{}^T = -\pi_1{}^T \overline{A} \tag{6}$$

and

$$\pi_1{}^T WB = \delta_{B2}{}^T \tag{7}$$

In both cases (i) and (ii), a system involving the matrix WB of order HM is sufficient for solving the larger systems of order H(M+K). This already represents a substantial saving in computation as K = N(N-1)/2 is usually large compared with H and M. Systems (4) and (7) are referred to as reduced systems, and it is with them that the steps of the revised simplex will be carried out.

4.TRANSFORMATION OF THE WORKING BASIS

Look more closely now at the working basis WB. By permuting columns of WB such that flow and slack variables which correspond to the same period are grouped together and such that capacity variables correspond to the last M columns of WB, we obtain a matrix of the form

$$WB = \begin{bmatrix} A_1 & & & & -I \\ & A_2 & & & -I \\ & & \ddots & & \vdots \\ & & & A_H & -I \end{bmatrix}.$$

where the A_i are of order Mxq_i, $i = 1,\ldots,H$, $\sum_{i=1}^{H} q_i = (H-1)M$, and I is the identity matrix of order M. Each A_i is of rank q_i since WB is non-singular.

Note that WB could be completely decomposed into H independent submatrices were it not for the columns representing capacity variables. On the other hand, the form of WB suggests that a triangular factorization would be easier to obtain and update than the calculation and update of the inverse of WB. Indeed, considerations of speed and accuracy for the general LP suggest the use of triangular factorization of the basis matrix as opposed to the use of its inverse (Chvatal (1983), chapter 4).

In what follows, a transformation effecting a compromise between complete decomposition and complete triangularization of WB is used. The transformation is carried in two steps.

First multiply

$$QWB = \begin{bmatrix} Q_1 & & & \\ & Q_2 & & \\ & & \ddots & \\ & & & Q_H \end{bmatrix} \begin{bmatrix} A_1 & & & & -I \\ & A_2 & & & -I \\ & & \ddots & & \vdots \\ & & & A_H & -I \end{bmatrix}$$

$$= \begin{bmatrix} R_1 & & & & -Q_1 \\ & R_2 & & & -Q_2 \\ & & \ddots & & \vdots \\ & & & R_H & -Q_H \end{bmatrix},$$

where the Q_i, $i = 1,\ldots,H$ are MxM non-singular matrices triangularizing A_i, that is, the R_i are Mxq_i upper triangular matrices with the last $M-q_i$ rows consisting of zeroes, and each Q_i is the product of q_i elementary Householder matrices used in the QR factorization of A_i (Strang(1986), chapter 5). The zero rows may be permuted to the bottom of the matrix by multiplying on the left by a suitable permutation matrix P giving

$$U = PQWB = Q'WB = \begin{bmatrix} U_1 & & & & -Q_1' \\ & U_2 & & & -Q_2' \\ & & \ddots & & \vdots \\ & & & U_H & -Q_H' \\ & & & & D^T \end{bmatrix},$$

where the U_i, $i = 1,\ldots,H$ are $q_i x q_i$ upper triangular matrices, the $-Q_i'$ are $q_i xM$ matrices equal to the $-Q_i$ matrices without their last $M-q_i$ rows and D^T is an MxM invertible matrix. Note that U is not in general upper triangular as D^T is not in general upper triangular, so that the factorization of WB is partial. A full factorization of WB is not attempted at this stage as this would alter the structure of the matrix D^T which summarizes the information concerning the periods.

It will be seen that D plays the role of the basis matrix for the revised simplex procedure in that when entering and leaving columns are exchanged in the basis B of the larger problem, then corresponding leaving and entering columns are exchanged in D, and that the work involved in solving the initial problem of rank HM + HK is essentially that involved in solving H problems of rank M.

5. THE REVISED SIMPLEX

The revised simplex procedure consists of the five steps:

A. Calculation of the dual variables;

B. Choice of the entering column;

C. Calculation of the transformed entering column;

D. Choice of the leaving column;

E. Update of the basis matrix.

We consider each of these steps as they apply to problem (3) while using the reduction of the working basis given in sections 3 and 4 in steps A, C, and E.

A. Calculation of the dual variables.

As shown in section 3 , the equation

$$\pi^T B = \delta_B{}^T$$

reduces to

$$\pi_1{}^T WB = \delta_{B2}{}^T = (0, g^T)$$

and

$$\pi_2{}^T = -\pi_1{}^T \bar{A} \tag{8}$$

Now set

$$\pi_1{}^T Q'^{-1} = v^T, \tag{9}$$

where Q' is, as above, the invertible matrix of order HM transforming WB into U. Then,

$$\pi_1{}^T WB = v^T Q' WB = v^T U = (0, g^T).$$

After partitioning v into variables corresponding to the rows of the U_i and D^T, we have

$$v^T U = (v_1^T, v_2^T, \ldots, v_H^T, v_D^T) \begin{bmatrix} U_1 & & & & -Q_1' \\ & U_2 & & & -Q_2' \\ & & \ddots & & \vdots \\ & & & U_H & -Q_H' \\ & & & & D^T \end{bmatrix} .$$

But $v_i^T U_i = 0$ implies $v_i = 0$, $i = 1,\ldots,H$, and $v_D^T D^T = g^T$ implies $Dv_D = g$.

Finally, the system

$$Dv_D = g \tag{10}$$

can be solved by triangularization of D. It is assumed that the initial matrix D has been triangularized and that this factorization is updated at each iteration.

B. Choice of the entering column.

For a minimization problem, the sufficient optimality criterion is that all non-basic reduced costs be non-negative. Since according to the proposition, all non-basic variables are flow or slack variables, this criterion becomes

$$-\pi^T a_t \geq 0$$

for all t non-basic where a_t is a flow or slack column of the constraint matrix of formulation (3) of section 2.

Let

$$\pi = (\pi_1, \pi_2) = (\pi_1^{\,1}, \pi_1^{\,2}, \ldots, \pi_1^{\,H}, \pi_2^{\,1}, \pi_2^{\,2}, \ldots, \pi_2^{\,H}) ,$$

where

$$\left.\begin{aligned} \pi_1^{\,h} &= (\pi_{11}^{\,h}, \pi_{12}^{\,h}, \ldots, \pi_{1M}^{\,h}) \\ \pi_2^{\,h} &= (\pi_{21}^{\,h}, \pi_{22}^{\,h}, \ldots, \pi_{2K}^{\,h}) . \end{aligned}\right. \qquad h = 1,\ldots,H$$

$$\pi_2^h = (\pi_{21}^h, \pi_{22}^h, \ldots, \pi_{2K}^h) \quad h = 1,\ldots,H$$

At optimality , all the variables π_1 will be non-positive, as the optimality criterion applied to the slack variables gives

$$-\pi^T s_i^h = -\pi_{1i}^h \geq 0$$

$$\text{or} \quad \pi_{1i}^h \leq 0,$$

for slack variables s_i^h, $h = 1,\ldots,H$ and $i = 1,\ldots,M$.

If the current basis is not optimal, then at least one value $-\pi^T a_t < 0$. A column generation method that will find such a value if it exists, and recognize optimality when it does not exist is now proposed. It will be noted that the basis entrance criterion is not the usual one of choosing the minimum reduced cost column.

Start by looking at reduced costs corresponding to flow variables.

$$-\pi^T a_t = d_k^h \sum_{i \in k(j)} (-\pi_{1i}^h) - \pi_{2k}^h ,$$

where column a_t corresponds to flow variable $\overline{f}_{kj}^h$ and $k(j)$ is the set of edges i in chain j connecting OD pair k. Thus, each reduced cost for a non-basic flow variable $\overline{f}_{kj}^h$ is equal (up to a positive affine transformation) to the length of a chain in the graph G connecting OD pair k.

For each period h assign lengths $-\pi_{1i}^h$ to every edge i such that $\pi_{1i}^h \leq 0$. If $\pi_{1i}^h > 0$, then assign to edge i a large positive length $L > \sum_{i=1}^{M} |\min(\pi_{1i}^h, 0)|$. Then for each period h and each OD-pair k solve a minimum chain problem between the nodes of OD-pair k; call this value MC_k^h. Then calculate

$$MC = \min_{i,h,k}(d_k^h MC_k^h - \pi_{2k}^h, \ -\pi_{1i}^h \ ),$$

and if $MC \geq 0$, then optimality is attained.

If $MC = d_{k*}^{h*} MC_{k*}^{h*} - \pi_{2k*}^{h*} < 0$, then choose the flow variable for period h* and OD pair k* to enter the basis. If $MC = -\pi_{1i}^{h*} < 0$, then choose the slack variable s_{i*}^{h*} to enter the basis.

Note that if there are positive π_{1i}^{h}, then the shortest chain calculation will not consider the true value of some reduced costs, as the corresponding chains have artificially inflated edge lengths. This means that the smallest non-basic reduced cost for flow variables is not necessarily calculated at a given iteration. However, in this case, there is always a slack variable eligible to enter the basis.

The reason that a slack variable is not immeditely entered in the basis when a $\pi_{1i}^{h} > 0$ is that the initial basis contains all but M of the slack variables and the minimum number of flow variables, and this criterion permits flow variables to enter more quickly. Of course, when all π_{1i}^{h} are non-positive, then the above calculation picks the smallest reduced cost and when this reduced cost is non-negative, then optimality is achieved.

C. Calculation of the transformed entering column.

Suppose column a_t corresponding to a flow or slack variable has been chosen to enter the basis. We seek the transformed column y_t satisfying the system

$$By_t = a_t \tag{11}$$

From section 3 equation (4), recall that system (11) is equivalent to the system

$$WBz_t = \bar{a}_t - \sum_{i=1}^{HK} a_{HM+it}\bar{a}_i \;, \tag{12}$$

where $\bar{a}_t$ and the $\bar{a}_i$ are vectors of the first HM components of a_t and the i^{th} key column respectively, and z_t is the last HM components of the vector $z_t = T^{-1}y_t$.

If a_t corresponds to flow in period h connecting OD pair k, then it can be written

$$a_t = \begin{bmatrix} \bar{a}_t \\ \hline e_{(h-1)K+k} \end{bmatrix},$$

where $e_{(h-1)\ K+k}$ is the $((h-1)K+k)^{th}$ unit HK-vector, that is,

$$a_{HM+it} = \begin{cases} 1 \text{ if } i = (h-1)K + k \\ 0 \text{ otherwise} \end{cases}$$

In this case, equation (12) can be written

$$WBz_t = \bar{a}_t - \sum_{i=1}^{HK} a_{HM+it}\bar{a}_i = \bar{a}_t - \bar{a}_{(h-1)K+k} := \hat{a}_t \qquad (13)$$

that is, $\bar{a}_t$ is reduced by the first HM components of its corresponding key variable in the basis.

If a_t corresponds to the i^{th} slack variable of period h, then

$$a_t = \begin{bmatrix} e_i \\ \hline 0 \end{bmatrix},$$

where e_i is the i^{th} unit HM-vector. Consequently, in this case, $a_{HM+it} = 0$, i = i,...,HK, and equation (12) can be written

$$WBz_t = \bar{a}_t - \sum_{i=1}^{HK} a_{HM+it}\bar{a}_i = \bar{a}_t := \hat{a}_t. \qquad (14)$$

For both cases, if equation (12) is multiplied on the left by matrix Q', derived in section 4, one obtains

$$Uz_t = Q'WBz_t = Q'\hat{a}_t,$$

that is,

$$\begin{bmatrix} U_1 & & & & -Q_1' \\ & U_2 & & & -Q_2' \\ & & \cdot & & \cdot \\ & & & \cdot & \cdot \\ & & & U_H & -Q_H \\ & & & & D^T \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ \cdot \\ \cdot \\ z_H \\ z \end{bmatrix} = \begin{bmatrix} 0 \\ \cdot \\ 0 \\ a \\ 0 \\ \cdot \\ \beta \end{bmatrix} \begin{matrix} \\ \\ \\ q_h\text{-vector} \\ \\ \\ M\text{-vector} \end{matrix}$$

This system is equivalent to

$$D^T z = \beta \qquad \text{or} \qquad z^T D = \beta^T \text{ and}$$

$$U_i z_i = Q_i' z, \; i = 1, \ldots, H, \; i \neq h$$

$$U_h z_h = Q_h' z + a.$$

It is supposed, as usual, that D is triangularized initially, and that the factorization of this triangularization is updated at each iteration.

D. Choice of the leaving column.

Given the column a_t chosen by step B to enter B', choose the leaving column a_r of B' by the usual minimum ratio criterion using the current value of the basic variables and the components of y_t calulated in step C. We may assume, according to the proposition of section 2, that a_r is a slack or flow column.

E. Update of the basis and the basic solution

Recall from section 3 that a basis B of the full system (3) can be transformed into the equivalent matrix

$$B' = \begin{bmatrix} \bar{A} & WB \\ I & 0 \end{bmatrix},$$

where

$$WB = \begin{bmatrix} A_1 & & & & -I \\ & A_2 & & & -I \\ & & \ddots & & \vdots \\ & & & A_H & -I \end{bmatrix},$$

and that WB can be further transformed by row operations into the matrix

$$U = \begin{bmatrix} U_1 & & & & -Q_1' \\ & U_2 & & & -Q_2' \\ & & \ddots & & \vdots \\ & & & U_H & -Q_H' \\ & & & & D^T \end{bmatrix}.$$

The update is carried out on B', but as it has been shown that the steps of the revised simplex depend on WB and more specifically on D, the effect of the exchange of entering and leaving columns on the R_i, the $-Q_i$, and D will be examined.

It is assumed that at each iteration of the revised simplex the matrix D is in triangular form U_D, and that if there is an exchange of columns in D, then update of the triangular factorization of D is carried out using one of the usual techniques (see for example Chvatal (1983), chapter 24). The D of the initial solution given in section 2 can be shown to correspond to the negative of a permutation matrix and can be easily factorized into the negative of the identity matrix of order M.

The notation $\hat{a}_t$ to represent the vector of the first HM components of the (HM+HK)-vector a_t after subtraction of the first HM components of its key variable has been introduced in step C. If $\hat{a}_t$ is a slack or flow column from period h entering or leaving WB, then by an abuse of notation we say that $\hat{a}_t$ enters or leaves A_h, even though it is the M components in locations (h-1)M+1 through hM of $\hat{a}_t$ which do so.

Recall that the first HK columns of B' are the key columns, flow vectors representing each period h and each OD pair k, h=1,...,H and k=1,...,K. Steps B and D of the revised simplex have determined columns a_t and a_r as entering and leaving columns respectively for the present iteration.

i) Removal of column a_r from B'

If a_r is a key variable representing period h and OD pair k, then since a_r is to leave the basis, it must be replaced in the basis by another key variable in order to maintain the canonical form of the basis matrix. Let $S = \{a_j | a_j = a_i + a_r,\ \hat{a}_i \in A_h$ and a_j is a basic variable representing a flow in period h between OD pair k}.

If $S = \phi$, then a_r is the only column of B' representing the demand $d_k{}^h$ between OD pair k in period h. Therefore, the entering column a_t must also represent OD pair k and period h; otherwise the updated basis matrix would contain a zero row and be singular. In this case, simply replace column a_r by a_t in B' and there is no effect on WB.

If $S \neq \phi$, then before taking a_r out of the basis, we replace a_r by one of the $a_{j*} \in S$. This amounts to replacing $\hat{a}_{j*} - \hat{a}_r$ in

A_h by $\hat{a}_r - \hat{a}_{j*} = -(\hat{a}_{j*} - \hat{a}_r)$ and for all other $a_j \in S$, to replacing $\hat{a}_j - \hat{a}_r$ in A_h by $\hat{a}_j - \hat{a}_{j*} = \hat{a}_j - \hat{a}_r - (\hat{a}_{j*} - \hat{a}_r)$.

Since the matrix U is obtained from WB by row operations only, then the effect on U of making a_{j*} a key variable in place of a_r is to perform the same column operations on U_h as just performed on A_h. In order to keep U_h upper triangular after these column operations, choose $a_{j*} \in S$ such that $\hat{a}_{j*} - \hat{a}_r$ is the leftmost column of A_h with $a_{j*} \in S$. Then subtraction of $\hat{a}_{j*} - \hat{a}_r$ from $\hat{a}_j - \hat{a}_r$ for all $a_j \in S$ leaves U_h in upper triangular form. Now remove $\hat{a}_r - \hat{a}_{j*}$ from the basis as described below.

Without loss of generality, one may assume now that the leaving column a_r corresponds to a flow or slack variable such that $\hat{a}_r$ is in A_h for some period h. In this situation $\hat{a}_r$ will leave A_h and $\hat{a}_t$ representing a flow or slack variable from some period i, $1 \le i \le H$ will enter A_i.

Suppose $\hat{a}_r$ is the j^{th} column of A_h, $1 \le j \le q_h$. Then removing $\hat{a}_r$ from A_h makes A_h of order $Mx(q_h - 1)$ and corresponds to removing the j^{th} column from R_h and thus to removing one row from the new $-Q_h'$ obtained from retriangularization of R_h . This row then becomes the entering row of the matrix D^T, that is, the entering column of D.

Specifically, the changes involved are the following : first remove the j^{th} column $\hat{a}_r$ from R_h giving the matrix $\bar{R}_h$. Then permute row j of $\bar{R}_h$ to the ${q_h}^{th}$ row while moving up by one row the rows $j+1, j+2, \ldots, q_h$; this is summarized by multiplication of $\bar{R}_h$ on the left by a permutation matrix P_h. Now make the q^{th} row of the resulting matrix zero by multiplication on the left by a lower triangular matrix L_h. Thus, $\hat{R}_h = L_h P_h R_h$ is the new upper triangular matrix for period h and the corresponding MxM matrix in the last M columns of WB is $-\hat{Q}_h = L_h P_h(-Q_h)$. Indentify the q_hth row of $-\hat{Q}_h$ as a possible entering row of D^T (rows q_h+1, $q_h+2,\ldots,M$ of $-\hat{Q}_h$ are already rows of D^T). It might not enter D^T as will be seen below.

ii) Entry of column a_t into B'.

a_t being a column from period i to enter B', enter $\bar{a}_t' = Q_i \hat{a}_t$ after the last column of R_i, making the new matrix $[R_i, a_t']$ of rank q_i+1 . Consider two cases depending on whether the entering and leaving columns are from different or the same periods.

(a) Suppose $i \neq h$. Then one of the components q_i+1, $q_i+2,\ldots,M$ of $\bar{a}_t'$must be non-zero; otherwise the matrix $[R_i, \bar{a}_t']$ would have rank q_i instead of q_i+1. By definition, the rows q_i+1, $q_i+2,\ldots,M$ of R_i correspond to the same numbered rows of $-Q_i$, which are also rows of D^T. Thus, retriangularization of $[R_i, \bar{a}_t']$, which involves permuting a non-zero pivot element in $\bar{a}_t'$ to the $(q_i+1)^{st}$ place and then subtracting multiples of row q_i+1 from rows $q_i+2,\ldots,M$, effects the same row operations on $-Q_i$ and D^T and corresponding column operations on D and U_D, the triangular factorization of D.

Let $P = \{j \mid q_i+1 \leq j \leq M, \bar{a}_{jt}' \neq 0\}$. Then in order to keep U_D in upper triangular form, we choose the pivot element in $\bar{a}_t'$ such that its corresponding column k in U_D is leftmost (after permutation of the pivot element in $\bar{a}_t'$) among all columns of U_D corresponding to a $j \in P$. Then the column operations on U_D which reflect the retriangularization of $[R_i, \bar{a}_t']$ amount to subtracting multiples of column k of U_D from columns to its right, thus preserving the upper triangular form of U_D.

At this point there is an entering row of D^T in row q_h of $-\hat{Q}_h$ and a leaving row of D^T in row q_i +1 of $-\hat{Q}_i$ ($-Q_i$ after the above row operations). Exchange of these columns in D is brought about by the update of the current triangular factorization of D.

(b) Suppose now that i = h (the entering and leaving columns of B' represent the same period). Then $L_h P_h \bar{a}_t'$ is added as column q_h to the matrix $\hat{R}_h$ obtained in i) above.

If the components $q_h+1, q_h+2, \ldots, M$ of $L_h P_h \bar{a}_t'$ are all zero, then the $q_h{}^{th}$ component of this vector must be non-zero or otherwise the matrix $C_h = [\hat{R}_h, L_h P_h \bar{a}_t']$ would be of rank q_h-1 instead of q_h. Hence, C_h is already upper triangular so that the matrix $-\hat{Q}_h$ is unchanged and its $q_h{}^{th}$ row which we identified above as being a possible addition to D^T remains in its place and is identified as the leaving row of D^T also. In this case, update of B' amounts to the deletion and addition of columns in A_h with resultant changes to R_h and $-Q_h$ but to no change of D or U_D.

If one of the components $q_h+1, q_h+2, \ldots, M$ of $L_h P_h \bar{a}_t'$ is non-zero (the $q_h{}^{th}$ component may be zero or non- zero), then proceed to retriangularize C_h as in step (a) above. The leaving row of D^T is found in row q_h of the updated matrix $-\hat{Q}_h$ and the entering row of D^T is the former $q_h{}^{th}$ row of $-\hat{Q}_h$ after transformation by the retriangularization of C_h. Again exchange these rows in D^T by updating the current triangular factorization of D.

iii) Update of the basic solution.

Replace the current basic solution x_D by $x_D - \theta y_t$ and $x_t = \theta$, where

$$\theta = \min_{i=1,\ldots,(H-1)M+HK}[x_{Di}/y_{it},\ y_{it} > 0]$$

was calculated in step (D) of the revised simplex.

6. CONCLUSION

The two-fold decomposition procedure outlined in this paper addresses an LP problem with a basis of size HK+HM and solves it as if it were H problems each with a basis of size M. The reduction in computational complexity is made evident by comparison of the number of operations necesary to carry out the steps of the revised simplex, first on the GUB working basis WB, and then on WB using the partial decomposition of this paper.

First note that step D, pivoting, is the same in both cases as this step does not involve the basis matrix. Without taking into consideration the structure of WB, the factorization of the initial basic solution of section 2 would be of order O(HM). Steps A and C, both require $O(H^2M^2)$ calculations, and update of the triangular factorization of WB would be of complexity $O(H^2M^2)$ also.

By contrast, the procedure outlined in this paper would initially factorize H rectangular matrices A_h of order Mxq_h, and the negative of a permutation matrix of order M taking $O(HM^3)$ operations. In step A, equation (10), $O(M^2)$ operations are involved in calculating the vector v_D and $O(HM^2)$ in recovering the dual variables π_1 in equation (9). The dual variables π_2 are recovered in equation (6) by $O(HKM) = O(HM^2)$ calculations owing to the special structure of the matrix A. Step C requires $O(HM^2)$ calculations to obtain the vector z_t from equation (13) and O(HM) additions to recover y_t from the equation $y_t = Tz_t$, owing to the special structure of the matrix T.

As for the basis update, all row and column operations are carried out on a maximum of three matrices of order at most M. Moreover, matrix multiplication is restricted to multiplication of columns on the left by lower triangular matrices differing from the identity matrix I_M by one row whose elements can be calculated in $O(M^2)$ operations. Hence, total computations for the update require $O(M^2)$ operations.

Periodic refactorization of WB is necessary in order to avoid excessive cumulative round off errors. Refactorization of WB without partial decomposition into periods has complexity $O(H^3M^3)$ whereas the method of this paper refactorizes H + 1 smaller matrices of size at most MxM for a complexity of $O(HM^3)$ per refactorization. Moreover, since at most 3 such submatrices out of H + 1 are changed per iteration, at least $v_1 = (H + 1)/3$ iterations are necessary before all H + 1 submatrices are changed at least once. If refactorization is done every v_1 iterations with partial decomposition and every v_2 iterations without it, then the number of operations due to refactorization over v_2 iterations is $(v_2/v_1)O(HM^3) = O(M^3)$ for partial decomposition versus $O(H^3M^3)$ without it.

Finally, note that if one did not have the result of the proposition of section 2, then testing of optimality (column generation) for each of the capacity variables would require O(HM) more calculations than the proposed method.

Of course, as for any algorithm, the real test of this method will be its actual performance as a computer-coded program.

Although this coding has not yet been carried out, it is believed that the increase in calculation due to the extra bookkeeping required, especially in the update step of the revised simplex, will be more than compensated for by the computational savings obtained in the decomposition of the working basis.

Note that the technique of this paper would apply with little modification to a problem where not only the capacity variables, but also the flow variables have non-zero costs. Only the column generation scheme would have to be slightly modified.

This technique is made possible by two observations: the capacity variables can be left in the basis, and triangularization of the basis matrix leaves intact the multi-period structure of the problem, whereas inversion of the basis matrix would destroy such structure.

It is hoped that this approach can be applied, perhaps in modified form, to the multi-period version of the problem proposed by McCallum (1977). This problem could be formulated as

$$\min \sum_{h=1}^{H} \sum_{j=1}^{n} k_j f_j^h + \sum_{i=1}^{M} g_i z_i$$

$$Af^h + s^h - z = c$$

$$Ef^h = d^h \qquad h = 1,\ldots,H$$

$$f^h,\ s^h,\ z \geq 0 \quad \text{all } h$$

where k_j is the unit cost attached to flow along chain j in G, the z_i are the capacity expansion variables for edge i, and all other notation is the same as in this article, except that c is a vector of constants. Here the proposition, at least as proved in section 2, would not apply to the capacity expansion variables, and thus the last M columns of a basis matrix might not keep a constant structure. However, such a problem may yet lend itself to a similar partial decomposition of its basis matrices, given that the structure of its matrix of constraints is the same as in this paper.

7. ACKNOWLEDGEMENTS

This research has been partially supported by the Natural Sciences and Engineering Research Council of Canada and the Fonds pour la formation de chercheurs et l'aide à la recherche du Québec.

REFERENCES

1. Chvatal, V.,(1983) Linear Programming , Freeman, New York.
2. Dantzig, G. and Van Slyke, M. (1967) "Generalized Upper Bounding Techniques," J. Comp. Systems Science, 1, 213-226.
3. Gomory, R.E., and Hu, T.C., (1961) "Multi-Terminal Network Flows", J. Soc. Indust. /appl. Math. 9, No. 4, 551-570.
4. Kortanek K. and Polak, G., (1985) "Network Design and Dynamic Routing under Queuing Demand," Zeitschrift für Operations Research, 29, 141-160.
5. Lasdon, L., (1972) Optimization Theory for Large Systems, Macmillan, New York.
6. McCallum, C ., (1977) "A Generalized Upper Bounding Approach to a Communications Network Planning Problem," Networks, 7, 1-23.

7. Minoux, M., (1989) "Network Synthesis and Optimum Network Design Problems: Models, Solution Methods and Applications", Networks, 19, 313-360.

8. Minoux, M. and Serreault, J.Y., (1981) "Subgradient Optimization and Large Scale Programming: an Application to Optimum Multicommodity Network Synthesis with Security Constraints", R.A.I.R.O. Recherche opérationnelle/Operations Reasearch 15, No. 2, 185-203.

9. Rioux, P., (1988) "Minimum Cost Sizing of Rearrangeable Networks with Multi-Period Demands," unpublished PhD thesis, Concordia University, Montréal, Quebec, Canada.

10. Strang, G.,(1986) Introduction to Applied Mathematics, Wellesley-Cambridge Press, Cambridge Mass.