LIBERTAS MATHEMATICA, VOL. 2, 1982

A TRAFFIC FLOW MODEL AND ITS SOLUTION BY MEANS OF THE GENERALIZED NUCLEOLUS

Irinel Dragan

A traffic flow model which is a multiobjective linear programming problem of special structure is given. An algorithm for finding the generalized nucleolus of a set in R^n defined by linear restrictions with respect to a set of linear functions (see [2]) leads to a particular algorithm being able to solve easily the model. The main purpose of this paper is that of stating the model and showing how it can be solved; a small example will be given, but the practical use of the model will be reported in a further paper.

1. The model. The problem discussed below could be stated nicely as a minimum cost multicommodity flow problem, but in such a case the size of the smallest practical problem would become prohibitive; this was the reason of introducing another model able to handle practical problems. We would like to mention that any real world problem connected to the traffic is always complex enough to enable the criticism of any model trying to solve it from some point of view. Therefore, the multiobjective linear programming problem given here as such a model, with the concept of solution defined below, is thought of only as a possible way of improving a real situation, without claiming that one can not do better.

Let $G = (X,E)$ be directed graph representing a network of streets in some area, X is the set of nodes and E is the set of arcs. Denote $|E| = m$ and suppose that the set of arcs is numbered in some way: $E = \{e_i \mid i = \overline{1,m}\}$. Each arc $e_i \in E$ has a capacity c_i which is a positive number. The case where the nodes would have capacities can be handled, too; some nodes of G can be substituted by some graphical configurations with capacities on the arcs.

Now, suppose that some pairs of nodes of G are chosen; any pair is an ordered pair with an initial and a final point. Denote P the sets of pairs and $|P| = n$ and suppose that the set of pairs is numbered in some way: $P = \{p_k \mid k = \overline{1,n}\}$. For instance, such pairs might be all the pairs formed by the entrance-exit points of the town, if the transit flow of the town is studied. Of course, we do not exclude identical pairs as well as pairs formed by two nodes taken in both possible orderings.

For each pair $p_k \in P$ we intend to send a flow f_k through G consistent with the given capacities of the arcs. The levels of all these flows have to be "as high as possible." Of course, this is a multiobjective programming problem and we shall state it mathematically by accepting a quite strong hypothesis.

Let us suppose that for each $p_k \in P$ one can use only one path D^k in order to carry the flow f_k and this path is known. Of course, if we are obliged to accept several paths for the same pair we can include in P the pair one time for each path, but the model has to keep a reasonable size of P. Our hypothesis assumes numerically that a $m \times n$ arc-path matrix $D = (d_{ik})$ is given: an element $d_{ik} > o$ says that the arc e_i belongs to the path D^k, eventually with some weight, if $d_{ik} \neq 1$, and $d_{ik} = o$ means that e_i does not belong to D^k.

We can write the capacity constraints and the nonnegativity constraints as: $D_i f \leq c_i$, $i = \overline{1,m}$, $f > o$, where D_i is the row i of D. Thus, the set of feasible flows is the set

$$\Omega = \{f \mid f \in R^n:\ D_i f \leq c_i,\ i = \overline{1,m}:\ f \geq o\} . \tag{1}$$

As we try to maximize the vector f, that is to minimize $-f$, we have the following set of objective functions to be minimized

$$U = \{u_k(f) \mid u_k(f) = -f_k,\ k = \overline{1,n}\} . \tag{2}$$

Hence, we stated the following multiobjective linear programming problem (MOLP) that represents our model: minimize the objective functions U on the feasible set Ω.

It is well known that for solving such a problem one has to define first what is meant by a solution of the problem; there are many possible definitions of such concepts, but we shall use the concept defined below.

Consider any nonempty set $\Omega \in R^n$ and any set of functions $U = \{u_k(x) \mid k = \overline{1,p}\}$ defined on Ω with real values. For each $\bar{x} \in \Omega$ a p-vector $\theta(\bar{x})$ is uniquely defined by taking the numbers $u_k(\bar{x})$, $k = \overline{1,p}$, in a nonincreasing ordering. The generalized nucleolus of Ω with respect to the set of functions U is the subset of Ω defined by

$$N(\Omega) = \{x \mid x \in \Omega,\ \theta(x) \underset{L}{\leq} \theta(y),\ \forall\, y \in \Omega\}, \tag{3}$$

where $\underset{L}{\leq}$ is the lexicographical ordering in R^p. The concept was introduced by M. Justman ([4]) as a natural extension of the concept of nucleolus met in game theory and due to D. Schmeidler ([6]). In a previous paper ([2]) an algorithm for finding the generalized nucleolus was given in the case where Ω is a nonempty compact set defined by linear inequalities and U is a set of linear functions. This algorithm was suggested by A. Kopelowitz' algorithm for finding the nucleolus ([5]); connected papers are ([1],[3]).

Now, consider the multiobjective programming problem: minimize a set of functions U on a feasible set Ω. We define as a solution of this problem any element of the generalized nucleolus. Thus, in order to solve the problem we have to find an element of this set. In the linear case this can be done by the algorithm mentioned above, but we shall apply this algorithm to our model, which is a particular MOLP problem: therefore, this algorithm becomes more efficient. As our model leads to a special version of the algorithm, the aim of the paper is to give the theoretical results justifying the particular algorithm and the corresponding statement.

2. Results. Let us consider the MOLP problem defined by (1), (2).

Theorem 1. The set of solutions of the MOLP problem (1), (2) consists of one and only one point.

Proof. As $c_i > o$, $i = \overline{1,m}$, we have $\Omega \neq \phi$, because $f = o$ is a feasible solution. For each $k = \overline{1,n}$, any component f_k of a feasible solution has to satisfy

$$o \leq f_k \leq \min_{i \mid d_{ik} \neq o} c_i/d_{ik} \tag{4}$$

hence Ω is a compact set. Thus the generalized nucleolus defined by (1), (2), (3) is a nonempty set according to the constructive proof given in ([2]) for the linear case. On the other part, taking account on the definition of the functions U in our case and on Justman's results ([4] , Prop. 4.5, Cor. 4.6, pg. 201), the generalized nucleolus of Ω consists in our case of only one point.

The Theorem 1 shows that the algorithm to be given below has to compute always successfully an unique solution.

Let us recall the operations of a step of the algorithm for finding the generalized nucleolus in the linear case given in ([2]). Now Ω is considered defined by a set of linear inequalities and the objective functions are linear. Consider the LP problem:

$$t = \min., \quad x \in \Omega, \quad u_k(x) \leq t, \quad k = \overline{1,p}. \tag{P}$$

If Ω is a nonempty compact set as we suppose, (P) has certainly an optimal solution; in fact, this happens under weaker conditions. Let t^* be the optimal value and define the set

$$S(P) = \{x \mid (t^*,x) = \text{optimal solution of } (P)\}.$$

A key result of ([2]) was: there exists a set of indices $\pi = \{k_1,\dots,k_{p^*}\}$ such that we have for all $x \in S(P)$

$$u_k(x) = t^* \quad \text{for all} \quad k \in \pi, \tag{5}$$

(lemma 1, pg. 6). In general such a set π can be found by assigning to π any index of a constraint in (P) corresponding to a positive dual variable in an optimal solution of the dual of (P); however, we shall find bellow such a set π by the direct use of its definition. Let us define the set

$$\Omega^* = \{x \mid x \in \Omega,\ u_k(x) = t^*,\ \forall\ k \in \pi\}; \tag{6}$$

it is a nonempty compact set. Now, if $|\pi| = p$, then $N(\Omega) = \Omega^*$, (Th. 2, pg. 10, [2]) and the problem is solved, if $|\pi| \neq p$, then $N(\Omega^*) = N(\Omega)$, (Th. 3, pg. 12, [2]), and new steps are needed. The set Ω is substituted by Ω^* and the set U is substituted by

$$U^* = \{u_k(x) \mid k = \overline{1,p},\ \forall\ k \notin \pi\}, \tag{7}$$

hence the new step will be done in similar conditions, but the number of objective functions has been certainly reduced.

The operations of a step of the algorithm are:

(A) Find the optimal value t^* of the LP problem (P);

(B) Find a set of indices π having the property (5);

(C) If $|\pi| = p$, STOP, the generalized nucleolus consists of $S(P)$; if $|\pi| < p$, add to the constraints of (P) those defined by π, in order to construct Ω^* and cancel from U the objective functions defined by these constraints, then pass to a new step

Consider our particular problem in order to prove the results leading to a special variant of the algorithm described above. Consider the LP problem to be solved by the operation (A):

$$t = \min.;\quad D_i f \leq c_i,\quad i = \overline{1,m};\quad f \geq o;\quad -f \leq te; \tag{P*}$$

where $e = (1,\ldots,1) \in R^n$

The following results show that we can perform all the operations of the above algorithm without using some code for solving LP problems.

Theorem 2. The optimal value of the problem (P) is*

$$t^* = - \min_{j/D_j e \neq o} c_j / D_j e \tag{8}$$

Proof. We have to show that t^* is a feasible value and any other $\bar{t} < t^*$ is unfeasible. Consider (f^*, t^*) defined by (8) and $f^* = -t^* e$. Obviously, $f^* \geq o$. If for some i we have $D_i e = o$, then we get $D_i f^* - c_i = -c_i < o$; if $D_i e \neq o$ we get

$$\begin{aligned} D_i f^* - c_i &= -t^* D_i e - c_i = -D_i e(t^* + c_i/D_i e) = \\ &= -D_i e(c_i/D_i e - \min_{j/D_j e \neq o} c_j/D_j e) \leq o; \end{aligned} \tag{9}$$

thus, (f^*, t^*) is feasible and t^* is a feasible value. Consider any feasible solution $(\bar{f}, \bar{t})$ of (P^*) and suppose $\bar{t} < t^*$. Let i^* be defined by (8) such that

$$t^* = -c_{i^*} / D_{i^*} e \tag{10}$$

where $D_{i^*} e \neq o$. Consider the constraint i^*, taking account on the fact that $-\bar{f} \leq \bar{t} e < t^* e$, i.e. we have $\bar{f}_k > -t^*$, $\forall\, k = \overline{1,n}$; we have

$$D_{i^*} \bar{f} - c_{i^*} > -t^* D_{i^*} e - c_{i^*} \tag{11}$$

and the equality (10) shows that the constraint i^* can not be satisfied; the hypothesis $\bar{t} < t^*$ was false and the theorem follows.

Theorem 2 shows that the rule (8) is a good rule for achieving the operation (A) of the algorithm.

Theorem 3. If $I \subset \{1,\ldots,n\}$ is the set of indices defined by

$$I = \{i \mid D_i e \neq o,\ c_i / D_i e = -t^*\}, \tag{12}$$

where t^* is given by (8), then we have

$$f_k^* = -t^* \tag{13}$$

for all the indices k *in the set*

$$K(I) = \{k \mid \exists\ i \in I \text{ such that } d_{ik} \neq o\} \tag{14}$$

in all the optimal solutions (f^*,t^*) *of* (P^*).

Proof. Suppose that for some $i^* \in I$ there is $k^* \in K(I)$, i.e. $d_{i^*k^*} \neq o$, such that $f^*_{k^*} > -t^*$ and consider the constraint i^*; taking account on $f_k^* \geq -t^*$, $k = \overline{1,n}$, and $f^*_{k^*} > -t^*$, we get

$$D_{i^*}f^* - c_{i^*} > -t^*D_{i^*}e - c_{i^*} = o; \tag{15}$$

thus the constraint i^* can not be satisfied; the hypothesis $f^*_{k^*} > -t^*$ was false and the theorem follows.

Theorem 3 shows that the rule (12, (13, (14) is a good rule for determining a set of indices π having the property (15): we take $\pi = K(I)$ and the operation (B) of the algorithm will be done.

Let us remark still that after the operation (B) all the variables f_k, $k \in K(I)$, will be fixed at the value $-t^*$ given by (8) and if $|K(I)| = n$, then the problem is solved. If $|K(I)| \neq n$ then we can easily pass to a new step after computing the residual capacities of the unsaturated arcs

$$c_i' = c_i + t^* \sum_{k \in K(I)} d_{ik},\ \forall\, i \in I. \tag{16}$$

Thus the rows corresponding to the saturated arcs and the columns corresponding to the assigned flows can be cancelled after computing the new right hand sides.

The above results justify the following algorithm for solving the model: let $A = (D/c)$ be the matrix of the model; perform the following operations:

1. Compute De and the rates

$$r_i = c_i / D_i e \quad i = \overline{1,m}, \tag{17}$$

for all i's such that $D_i e \neq o$; among these numbers determine the minimum rate r_o and record $t^* = -r_o$ as well as the set I of all the indices of the rates equal to r_o;

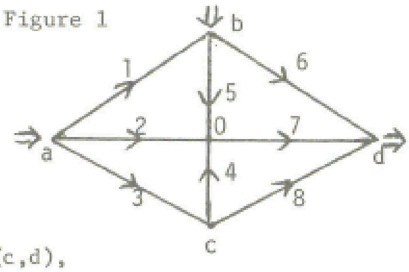
2. For each $i \in I$ determine those k's such that $d_{ik} \neq o$ and record them in $K(I)$ until the whole set I is exhausted; then, take on $f_k = r_o$, $\forall\, k \in K(I)$.

3. If $|K(I)| = n$, STOP, the problem is solved; otherwise, cancel all rows in I, compute c_i', $\forall\, i \notin I$, cancel all the columns in $K(I)$ and pass to a new step.

Let us remark that other rows might also be cancelled if they correspond to unsaturated arcs belonging to the already considered paths, in the case where the remaining paths do not pass through these arcs.

3. Example. Consider the directed graph G given in the figure 1 and suppose that we intend to pass a maximal flow through G, the entrance points being a and b and the exit point being d.

Figure 1



Suppose also that the flows can pass only through the paths

$D^1 = [a,b,o,d]$, $D^2 = [a,o,d]$, $D^3 = [a,c,o,d]$, $D^4 = [a,c,d]$, $D^5 = [a,b,d]$, $D^6 = [b,d]$, $D^7 = [b,o,d]$.

If the arcs are denoted in the ordering

(a,b), (a,o), (a,c), (c,o), (b,o), (b,d), (o,d), (c,d),

then the capacity function of the network is given by the vector (6,3,4,4,6,6,8,3). The augmented matrix of our problem which contains the arc-path matrix and the capacities is

arc	D^1	D^2	D^3	D^4	D^5	D^6	D^7	c
1	1				1			6
2		1						3
3			1	1				4
4			1					4
5	1						1	6
6					1	1		6
7	1	1	1				1	8
8				1				3

The problem can be solved in two steps. The first step gives

$$\min_i \; c_i / D_i e = 2, \quad I = \{3,7\}, \quad K(I) = \{1,2,3,4,7\}, \tag{18}$$

hence we have

$$t^* = -2, \quad f_1 = f_2 = f_3 = f_4 = f_7 = 2, \tag{19}$$

and the rows 3,7 as well as the columns 1,2,3,4,7, have to be cancelled; the new tableau with the updated capacities can contain only the following

arc	D^5	D^6	c
1	1		4
6	1	1	6

.

One can remark that the rows 2,4,5,8 were unuseful because they correspond to arcs belonging only to already considered paths. The second step gives

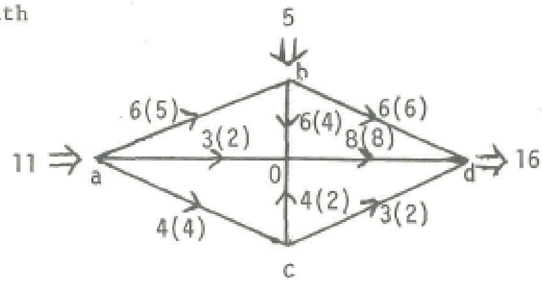
$$\min \; c_i / D_i e = 3, \quad I = \{6\}, \quad K(I) = \{5,6\} \tag{20}$$

hence we have

$$t^* = -3, \quad f_5 = f_6 = 3, \tag{21}$$

and the flow defined by the model can be read in the figure 2 near the capacities; of course, the model imposes some homogeneity in the use of the network, because every path has to carry at least two units of flow.

Figure 2



REFERENCES

[1] Behringer, F. *Lexicographic Quasiconcave Multiobjective Programming* Z.O.R., *21*, 1977, pp. 102-116.

[2] Dragan, I. "A game theoretic approach for solving the multiobjective linear programming problems; an application to a traffic problem," Tech. Rep. Univ. Pisa, Feb. 1981.

[3] Heindl, G. *G-optimale Entscheidungen und ihre grundlegenden Eigenschaften*, *OR Verfahren*, *26*, 1976, pp. 672-688.

[4] Justman, M. "Regulative frameworks for iterative negotiations," RM the Univ. of Jerusalem 1973 and "Iterative processes with nucleolar restrictions," *Int. Journal of Game Theory*, *6*, *4*, 1977, pp. 189-212.

[5] Kopelowitz, A. "Computation of the kernels of simple games and the nucleolus of n-person games," RM 31 the Univ. of Jerusalem 1967.

[6] Schmeidler, D. "The nucleolus of characteristic function game," *SIAM J. APPL. MATH.*, *17*, *6*, 1969, pp. 1163-1170.