NETWORK SIMPLEX METHOD

for the

TRANSSHIPMENT PROBLEM

C&O 370 WINTER 1995

The transshipment problem on the digraph $G = (V(G), E(G))$ is the linear program

$$(TP) \qquad \min z = cx \text{ s.t. } Ax = b, x \geq 0.$$

In (TP), $A = \{a_{ve} : (v,e) \in V(G) \times E(G)\}$ is the *incidence matrix of* G, defined by

$$a_{ve} = \begin{cases} -1 & \text{if} \quad v = t(e) \\ 1 & \text{if} \quad v = h(e) \\ 0 & otherwise. \end{cases}$$

$c = \{c_e : e \in E(G)\}$ is a given real *cost* vector, and $b = \{b_v : v \in V(G)\}$ is a given real *demand* vector.

A tree solution of (TP) is a vector $x^0 = \left\{x_j^0 : j \in E(G)\right\}$ which is the (unique) solution of the equality constraints of (TP) together with $x_j = 0, j \in E \backslash T$, where edge-set of some spanning tree of G. (Henceforth, we abbreviate *edge-set of a spanning tree of* G to *tree of* G).

If a tree solution x^0 of (TP) is also nonnegative, then x^0 is a <u>feasible tree solution</u>. Any tree determining a feasible tree solution is a <u>feasible tree</u>. Given such a tree T and edge set $e \in E \backslash T$, let $C(T, e)$ be the unique circuit of G satisfying:

1. every edge of $C(T, e)$ is in $T \cup \{e\}$

2. e is a forward edge of $C(T, e)$.

The <u>cost</u> (with respect to c) of $C(T, e)$ is defined to be

$$\begin{aligned} &\textstyle\sum\{c_j : j \text{ is a forward edge of } C(T,e)\} \\ - \quad &\textstyle\sum\{c_j : j \text{ is a reverse edge of } C(T,e)\} \end{aligned}$$

(The cost of a path is defined analogously.)

For $e \in E$, we have $c_e(T)$ is the cost of $C(T, e)$ if $e \in E \backslash T$ and is 0 if $e \in T$.

NETWORK SIMPLEX METHOD

Step 0

Find a feasible tree T with associated solution x^0, or show that no feasible tree exists.

Step 1

If $c_j(T) \geq 0$, for all $j \in E \backslash T$, then STOP; x^0 is an optimal solution. Otherwise, choose $e \in E \backslash T$ with $c_e(T) < 0$.

Step 2

If $C(T, e)$ has no reverse edge, the STOP; (TP) is unbounded.

Step 3

Let

$\theta = \min\{x_j^0 : j$ is a reverse edge of $C(T, e)\}$.

Let

$F = \{j : j$ is a reverse edge of $C(T, e) \& x_j^0 = 0\}$.

Define $x^1 = \{x_j^1 : j \in e\}$ by

$$x_j^1 = \begin{cases} x_j^0 + \theta & \text{if } j \text{ forward in } C(T, e) \\ x_j^0 - \theta & \text{if } j \text{ reverse in } C(T, e) \\ x_j^0 & otherwise. \end{cases}$$

Choose $f \in F$, and let $T^1 = \{T \cup \{e\}\} \backslash \{f\}$. Replace T by T^1 and x^0 by x^1 and go to Step 1.

Comments

1. Step 0 requires a *Phase 1* subroutine.

2. The numbers $c_j(T)$ of Step 1 can be evaluated by maintaining the vector $y(T) = \{y_v(T) : v \in V\}$, where relative to a designated root node r, $y_v(T)$ is the cost of the rv-path in T. Then, it can be shown that

$$c_j(T) = c_j + y_{t(j)}(T) - y_{h(e)}(T), \forall j \in E.$$

3. The termination criterion of Step 1 is valid. This can be demonstrated by showing that $y(T)$ is dual feasible and satisfies complementary slackness with x^0.

4. The termination criterion of Step 2 is valid, since x^1 defined in step 3 is feasible for θ arbitrarily large and has $cx^1 = cx^0 + \theta c_e(T)$.

5. The following refinement due to R. Bland does not admit cycling:

   Let $e_1, e_2, \ldots, e_n$ be an arbitrary, fixed ordering of $E(= E(G))$. Then,

   (a) choose e in step 1 to have the least index possible

   (b) choose f in step 3 to have the least index possible.