Journal of the Operations Research
Society of Japan
Vol. 30, No. 3, September 1987

A POLYNOMIAL-TIME DUAL SIMPLEX ALGORITHM FOR THE MINIMUM COST FLOW PROBLEM

Akira Nakayama
Otaru University of Commerce

(Received October 11, 1985; Final May 11, 1987)

Abstract Recently, Ikura and Nemhauser proposed a polynomial-time algorithm for the transportation problem of the Hitchcock type by the dual simplex method. The Ikura-Nemhauser algorithm can also solve the general minimum cost flow problem by reducing the minimum cost flow problem to a Hitchcock problem. In this paper, instead of such a reduction, we show a polynomial-time dual simplex algorithm for finding a minimum cost flow in a two-terminal capacitated network by applying Ikura and Nemhauser's idea directly to general two-terminal networks.

0. Background

Dantzig [3] devised the simplex method for linear programming problems and showed a way of specializing the simplex method to the minimum cost flow problem, a special case of linear programming problems. Cunningham [2] recently developed a nice anticycling rule for the network simplex method. Later primal dual method for network flow problems has been proposed by Ford and Fulkerson [5] and Iri [10]. Fulkerson [6] also proposed an approach called out-of-kilter method to the minimum cost flow problem. Other algorithms, primal methods, to find a minimum cost flow are seen in the work by Balinski and Gomory [1] and Klein [11]. However, all of these methods are not polynomial-time algorithms. In 1972, Edmonds and Karp [4] gave a polynomial-time algorithm for the first time by introducing a new technique, i.e., capacity-scaling method. Lawler [12] also showed a polynomial-time algorithm by employing the out-of-kilter method with capacity scaling.

Recently, Ikura and Nemhauser [8,9] gave a polynomial-time dual simplex algorithm for the Hitchcock transportation problem. Furthermore, Tardos [15] has recently shown a strongly polynomial algorithm for the minimum cost flow problem. Namely, she has affirmatively solved the open problem posed by Edmonds and Karp in 1972. This is the problem of finding an algorithm for the minimum cost flow problem which requires polynomial time in the numbers of vertices and arcs of the underlying graph and independent of the sizes of the cost and capacity functions. Very recently, Orlin [13] and Fujishige [7] have devised faster algorithms than Tardos's.

In this paper, we present a polynomial-time dual simplex algorithm for finding a minimum cost flow in a two-terminal capacitated network by applying Ikura and Nemhauser's idea [8]. Our algorithm is not a strongly polynomial-time one, but it is expected that the proposed dual simplex method is a practically efficient one as is the simplex method for general linear programming problems.

Since the minimum cost flow problem can be reduced to a Hitchcock problem ([5]), it can be solved by the Ikura-Nemhauser algorithm by reducing it to a Hitchcock problem. It would, however, be pointless to do so from the computational point of view. It would be worth presenting a version of the Ikura-Nemhauser algorithm which can directly be applied to general two-terminal networks. As will be seen later, it is not so straightforward to devise such a version of the Ikura-Nemhauser algorithm and to estimate the computational complexity.

1. Minimum Cost Flow Problem

Let $G=(V,A,\partial^+,\partial^-)$ be a (directed) graph. V and A denote the set of vertices and the set of arcs of G, respectively. ∂^+ (resp. ∂^-) is a function from A to V and for each arc a, $\partial^+ a$ (resp. $\partial^- a$) denotes the initial (resp. terminal) vertex of a. For simplicity, we also write $G=(V,A)$ in place of $G=(V,A,\partial^+,\partial^-)$. For two graphs $G_i=(V_i,A_i)$ $(i=1,2)$, $G_1 \cup G_2$ is defined by $G_1 \cup G_2=(V_1 \cup V_2, A_1 \cup A_2)$, and $G_1 \subset G_2$ means that G_1 is the subgraph of G_2. For a graph $G=(V,A)$, the set R of real numbers, given nonnegative upper capacity function $b:A \to R$ and cost function $c:A \to R$, let $N=(G=(V,A),b,c)$ be a network. Throughout this paper, we will employ the following rule of expressions in algorithms, propositions and every part of this paper. When we write as $X^{\pm}=Y^{\mp} \pm Z^{\pm}$, we mean two relations $X^+=Y^-+Z^+, X^-=Y^+-Z^-$. Then the minimum cost flow problem (P) is formulated as follows.

$$\text{(P):}\quad \min\Sigma\{c(a)x(a):a\in A\},$$
$$\text{(1.1)}\quad \Sigma\{x(a):a\in\delta^{+}v\}-\Sigma\{x(a):a\in\delta^{-}v\}=0 \quad (\ v\in V\ ),$$
$$\text{(1.2)}\quad 0\leq x(a)\leq b(a) \quad (\ a\in A\ ).$$

Here, $\delta^{+}v$ (resp. $\delta^{-}v$) is defined as $\delta^{+}v=\{a\in A:\ \partial^{+}a=v\}$ (resp. $\delta^{-}v=\{a\in A:\ \partial^{-}a=v\}$). For a network N, a function $x:A\to R$ (or $(x(a):a\in A)$) satisfying (1.1) is called a circulation or a flow in N. If a circulation x also satisfies (1.2), then x is called feasible. Without loss of generality, we assume that $b(a)>0$ for all $a\in A$ and that G is strongly connected.

2. Dual Simplex Method for the Minimum Cost Flow Problem

2.1. Potential and Circulation

The vector $(p(v):v\in V)$ (or simply p) is called a potential, where the value $p(v)$ is given for each vertex $v\in V$. First, our particular potential p is determined by the next operation, which is called Procedure POT(T_r), where T_r is a spanning tree with root r.

Procedure POT(T_r)

Step 1: Let $p(r)=0$.

Step 2: Calculate p such that $c(a)+p(\partial^{+}a)-p(\partial^{-}a)=0$ for any $a\in A(T_r)$.

Procedure FLO(p,T_r) shown below finds a circulation in the network N, where p is the potential obtained from Procedure POT(T_r).

Procedure FLO(p,T_r)

Step 1: (1.i) If $A-A(T_r)=\Phi$, then we set $(x(a):a\in A)=0$ and stop.

(1.ii) For each arc $a\in A-A(T_r)$, if $c(a)+p(\partial^{+}a)-p(\partial^{-}a)\geq 0$, then we put $x(a)=0$. Otherwise, we put $x(a)=b(a)$.

Step 2: Decide $(x(a):a\in A(T_r))$ so that the condition (1.1) may hold by using $(x(a):a\in A-A(T_r))$ in Step 1.

Example-1: In Fig.2, the network $N=(G=(V,A),b,c)$ has $V=\{1,2,3,4\}$ and $A=\{a_i:1\leq i\leq 6\}$. For each arc a_i, the ordered pair attached to a_i is $(c(a_i),b(a_i))$. Let T_1 be a spanning tree in Fig.3 whose arcs consist of three waving arrows a_2,a_3,a_5, where vertex 1 is the root. By Procedures POT(T_r) and FLO(p,T_r), we can find a potential p and a circulation x of the network N in the same figure.

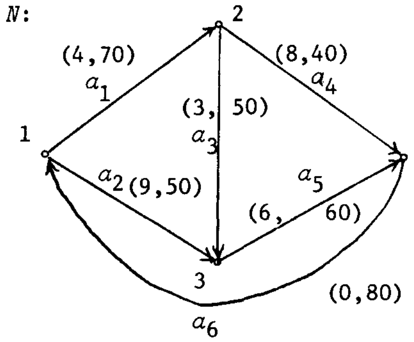

Fig.2

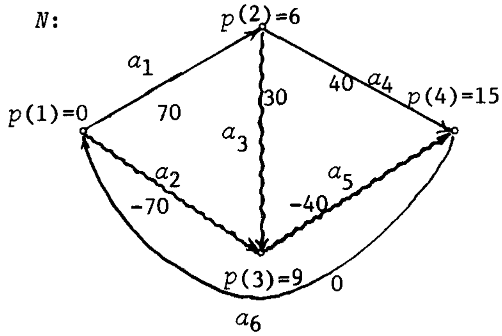

Fig.3

2.2. Pivot Operation

In order to describe the dual simplex method, we define the dual problem (DP) of the primal problem (P) as follows.

$$\text{(DP)}: \quad \max -\Sigma\{b(a)\gamma(a): a\varepsilon A\},$$
$$p(\partial^- a)-p(\partial^+ a)+\lambda(a)-\gamma(a)=c(a) \qquad (\ a\varepsilon A\ ),$$
$$\lambda(a)\geqq 0,\ \gamma(a)\geqq 0 \qquad (\ a\varepsilon A\ ),$$
$$p(v) \text{ is a free variable} \qquad (\ v\varepsilon V\ ).$$

Then we have the following complementary slackness condition (CS) for the primal (P) and the dual (DP):

(CS): For any $a\varepsilon A$,

$$\lambda(a)=0 \text{ if } x(a)>0,$$
$$x(a)=0 \text{ if } \lambda(a)>0,$$
$$x(a)=b(a) \text{ if } \gamma(a)>0.$$

The next proposition is well known.

Proposition 2.1. If the feasible solution $x(a)$ $(a\varepsilon A)$ of the primal problem (P) and the feasible one $\lambda(a)$, $\gamma(a)$ $(a\varepsilon A)$, $p(v)$ $(v\varepsilon V)$ of the dual (DP) satisfy condition (CS), then $x(a)$ $(a\varepsilon A)$ is an optimal solution of (P) and $\lambda(a)$, $\gamma(a)$ $(a\varepsilon A)$, $p(v)$ $(v\varepsilon V)$ is an optimal solution of (DP). Moreover the reverse is also true. □

Note that this fact still holds even if condition (CS) is replaced with the next one:

$(CS)^*$: For any $a\varepsilon A$,

$$p(\partial^- a)-p(\partial^+ a)\leqq c(a) \text{ if } x(a)=0,$$

$$p(\partial^- a)-p(\partial^+ a)=c(a) \text{ if } 0<x(a)<b(a),$$
$$p(\partial^- a)-p(\partial^+ a)\geq c(a) \text{ if } x(a)=b(a).$$

Now we show how the pivot operations of the dual simplex method are realized on the network N. A leaf branch B_a with respect to T_r and $a\in A(T_r)$ is defined as the connected component of T_r-a not containing root r. Arcs $a\in A-A(T_r)$, $a'\in A(T_r)$ are called cotree arc, tree arc, respectively. The two sets $A^+(T_r)$ and $A^-(T_r)$ are defined as

$$A^{\pm}(T_r)=\{a\in A(T_r):\partial^{\pm}a\in V(P_{r\partial^{\mp}a})\}$$

where $P_{r\partial^+ a}$ (resp. $P_{r\partial^- a}$) is the path joining root r and vertex $\partial^+ a$ (resp. $\partial^- a$) on tree T_r. Define E^+ and E^- by

$$E^{\pm}=\{a\in A^{\pm}(T_r):x(a)>b(a)\}\cup\{a\in A^{\mp}(T_r):x(a)<0\}.$$

For a number ρ and an arc $a\in E^{\pm}$, Operation 1 (p,T_r,ρ,a) given below determines a new potential denoted by p again.

Operation 1 (p,T_r,ρ,a):

For any vertex $v\in V(B_a)$, set $p(v)=p(v)+\rho$.

By the way, in order to complete the dual simplex method, this value must be determined so that the new potential p and the circulation x given by Procedure FLO(p,T_r) may satisfy the condition (CS)*. Here, for an arc $e\in A$ and a potential p, let $\tau(p,e)$ be the reduced cost given by $\tau(p,e)=c(e)+p(\partial^+ e)-p(\partial^- e)$, and for an arc $a\in E^{\pm}$, put

$$S_a^{\pm}=\{e\in H_a^{\pm}:\tau(p,e)\geq 0,\ x(e)=0\},$$
$$T_a^{\mp}=\{e\in H_a^{\mp}:\tau(p,e)\leq 0,\ x(e)=b(e)\}$$

where H_a^+ and H_a^- are defined as

$$H_a^{\pm}=\{e\in A-A(T_r):\partial^{\pm}e\in V(T_r)-V(B_a),\ \partial^{\mp}e\in V(B_a)\}.$$

Then Operation 2 $(a,S_a^{\pm},T_a^{\mp})$ given below decides the value of ρ.

Operation 2 $(a,S_a^{\pm},T_a^{\mp})$:

Find the value ρ by

$$\rho=\pm\min\{\min\{\tau(p,e):e\in S_a^{\pm}\},\ \min\{-\tau(p,e):e\in T_a^{\mp}\}\}$$

and let a_ρ be the arc which attains the minimum value ρ.

By each pivot operation, a tree arc breaking the capacity constraint (1.2) is to be replaced by some cotree arc so that the condition (CS)* is preserved after each pivot operation. When the potential p and the circulation x obtained in Section 2.1 are given for the network N, the dual simplex method is stated as follows.

Dual simplex method

Step 1: If $E^+\cup E^-=\phi$, then x is a minimum cost flow, and the algorithm terminates.

Step 2: Choose an arc $a\varepsilon E^{\pm}$. Then find the value ρ and an arc a_ρ by Operation 2 $(a, S_a^{\pm}, T_a^{\mp})$.

Step 3: (3.i) Renew the potential by Operation 1 (p, T_r, ρ, a).

(3.ii) Set $T_r = T_r - a + a_\rho$. Then renew the circulation by Procedure FLO(p, T_r), and return to Step 1.

3. Refinement of Dual Simplex Method

In Section 3.1, we introduce the procedure called Tree Partitioning which decides the tree arc taken in the pivot operation. This is a key operation for showing the polynomial running time. Section 3.2 gives a deficiency function used for proving polynomiality of our algorithm described in Section 3.3.2. Trimming operation in Section 3.3.1 is implemented as a preparation before pivot operations.

3.1. Tree Partitioning

We divide the rooted spanning tree T_r into a collection of disjoint subtrees, and partition them into three classes. This procedure, called Tree Partitioning, is implemented by Procedure TP $(T_r, (x(a): a\varepsilon A(T_r)))$ described below. In the dual simplex method stated in Section 2, pivot operations may be repeated infinitely. However, Tree Partitioning decides the right arc taken in the pivot operation so that the algorithm may run in polynomial time as we will see later. For the circulation x obtained in Section 2.1, we assume that the number z_r defined by

$$z_r = \Sigma\{x(a): a\varepsilon\delta^+ r\} - \Sigma\{x(a): a\varepsilon\delta^- r\}$$

is assigned to the root r. We have $z_r = 0$ at the present time. Then Procedure TP $(T_r, (x(a): a\varepsilon A(T_r)))$ is given as follows.

Procedure TP $(T_r, (x(a): a\varepsilon A(T_r)))$

(P0): Set $\underline{\mathcal{D}}^+ = \phi$, $\underline{\mathcal{D}}^- = \phi$ and $\underline{\mathcal{N}} = \phi$.

(P1): If $0 \le x(a) \le b(a)$ for each $a\varepsilon A(T_r)$, then put

$\underline{\mathcal{N}} = \underline{\mathcal{N}} \cup \{T_r\}$ if $z_r = 0$,

$\underline{\mathcal{D}}^- = \underline{\mathcal{D}}^- \cup \{T_r\}$ if $z_r > 0$,

$\underline{\mathcal{D}}^+ = \underline{\mathcal{D}}^+ \cup \{T_r\}$ if $z_r < 0$,

$\tilde{d}(T_r) = |z_r|$

and stop.

(P2): Choose an arc $a^* \varepsilon E^{\pm}$ and a leaf branch B_{a^*} of T_r such that $0 \le x(a) \le b(a)$ for any $a\varepsilon A(B_{a^*})$. Then set

$\underline{\mathcal{D}}^{\mp} = \underline{\mathcal{D}}^{\mp} \cup \{B_{a^*}\}$,

$$h=\begin{cases} -x(a^*) & \text{if } a^*\varepsilon(E^+\cap A^-(T_r))\cup(E^-\cap A^+(T_r)), \\ x(a^*)-b(a^*) & \text{if } a^*\varepsilon(E^+\cap A^+(T_r))\cup(E^-\cap A^-(T_r)). \end{cases}$$

(P3): If $\underline{\mathcal{D}}^+\cup\underline{\mathcal{D}}^-=\phi$, then we end this procedure. Otherwise, choose $Q\varepsilon\underline{\mathcal{D}}^\pm$, and set

$F^\pm=\{a\varepsilon A^\pm(T_r):x(a)<b(a)\}\cup\{a\varepsilon A^\mp(T_r):x(a)>0\}$,

$Z^\pm=\{a\varepsilon A^\pm(T_r):x(a)=b(a)\}\cup\{a\varepsilon A^\mp(T_r):x(a)=0\}$.

(P3.i) If $A(Q)\cap Z^\pm=\phi$, then put $\tilde{d}(Q)=h$, and go to (P4).

Otherwise, choose a leaf branch $B_{a''}$ of Q such that $A(B_{a''})\subset F^\pm$ for $a''\varepsilon A(Q)\cap Z^\pm$. Then we set

$\underline{\mathcal{N}}=\underline{\mathcal{N}}\cup\{B_{a''}\}$,

$\underline{\mathcal{D}}^\pm=\underline{\mathcal{D}}^\pm-\{Q\}$,

$Q = Q-V(B_{a''})$,

$\underline{\mathcal{D}}^\pm=\underline{\mathcal{D}}^\pm\cup\{Q\}$

and return to (P3.i).

(P4): For the arc $a^*\varepsilon E^\pm$ obtained in (P2), carry out ADJUST(T_r,a^*,$(x(a):a\varepsilon A(T_r))$,$z_r$) and return to (P1).

Here, the meaning of the function $\tilde{d}$ appearing in (P1) and (P3) will be made clear in Section 3.2, while ADJUST(T_r,a^*,$(x(a):a\varepsilon A(T_r))$,$z_r$) (or simply ADJUST-operation) in (P4) is as follows.

ADJUST (T_r,a^*,$(x(a):a\varepsilon A(T_r))$,$z_r$)

(1.i): For $a^*\varepsilon E^\pm$ define ν by

$$\nu=\begin{cases} -x(a^*) & \text{if } a^*\varepsilon A^\mp(T_r), \\ -b(a^*)+x(a^*) & \text{if } a^*\varepsilon A^\pm(T_r). \end{cases}$$

(1.ii): Let $T_r=T_r-V(B_{a^*})$ and P_{ru} be the path of the new tree T_r such that $u\varepsilon\{\partial^+a^*,\ \partial^-a^*\}$. Then adjust $(x(a):a\varepsilon A(T_r))$ and z_r in the following way.

$x(a)=x(a)-\nu$ if $a\varepsilon A^\pm(T_r)\cap A(P_{ru})$,

$x(a)=x(a)+\nu$ if $a\varepsilon A^\mp(T_r)\cap A(P_{ru})$,

$z_r=z_r\mp\nu$.

After carrying out Procedure TP(T_r,$(x(a):a\varepsilon A(T_r))$), for $\underline{\mathcal{C}}\ \varepsilon\{\underline{\mathcal{D}}^+,\underline{\mathcal{D}}^-,\underline{\mathcal{N}}\}$ we construct the new class $\mathcal{C}$ by doing the following JOIN($\underline{\mathcal{C}}$) (or simply JOIN-operation) which combines the members of $\underline{\mathcal{C}}$.

JOIN ($\underline{\mathcal{C}}$)

Step 1: If $|\underline{\mathcal{C}}|\leq 1$, then set $\mathcal{C}=\underline{\mathcal{C}}$ and stop.

Step 2: If there exist an arc $a\varepsilon A(T_r)$ and two distinct members Q,Q' of

$\underline{\mathcal{C}}$ connected by the arc a, then set

$\underline{\mathcal{C}} = \underline{\mathcal{C}} - \{Q, Q'\}$,

$\underline{\mathcal{C}} = \underline{\mathcal{C}} \cup \{Q \cup Q' + a\}$

and return to Step 1. Otherwise, put $\mathcal{C} = \underline{\mathcal{C}}$ and stop.

An example of these operations will be shown in Example-2 of the next section.

3.2. Deficiency Functions

When we implement Tree Partitioning, three classes $\underline{\mathcal{D}}^+$, $\underline{\mathcal{D}}^-$ and $\underline{\mathcal{N}}$ are obtained in Section 3.1. Moreover, for the set R_+ of nonnegative numbers, a function $\tilde{d}$: $\underline{\mathcal{D}}^+ \cup \underline{\mathcal{D}}^- \to R_+$ is introduced in Tree Partitioning. This function $\tilde{d}$ is called a deficiency function, which can be regarded as the measure of the primal infeasibility. We define a function d for the members of $\mathcal{D} = \mathcal{D}^+ \cup \mathcal{D}^-$, also called a deficiency function, by

(3.1) $\qquad d(Q^*) = \Sigma\{\tilde{d}(Q) : Q \in \underline{\mathcal{D}}^\pm,\ Q \subset Q^*\} \qquad (Q^* \in \mathcal{D}^\pm).$

The quantity $d(\mathcal{D})$ defined by

(3.2) $\qquad d(\mathcal{D}) = \Sigma\{d(Q^*) : Q^* \in \mathcal{D}\}$

is called the deficiency of the circulation x in network N.

Example-2: Fig.4 is the same as Fig.2, where the set of solid arcs constitutes a rooted spanning tree T_1, and the dotted arcs mean cotree arcs. The ordered pair attached to each tree arc a shows $(x(a), b(a))$. We perform Tree Partitioning by using the given T_1 and $(x(a) : a \in A(T_1))$ in Fig.3. In Procedure TP $(T_1, (x(a) : a \in A(T_1)))$, (P1) is skipped first. Going to (P2), we have $G \in \underline{\mathcal{D}}^+$ for graph $G = (\{4\}, \phi)$ because $x(a_5) = -40 < 0$. For this graph G, we have $\tilde{d}(G) = 40$ in (P3). Then proceeding to (P4), the values $x(a)$ $(a \in A(T_1'))$ will be adjusted by ADJUST-operation, where $T_1' = T_1 - \{4\}$. That is, we add to x a flow of 40 from root 1 to arc a_5 so that we have the subtree T_1' with the

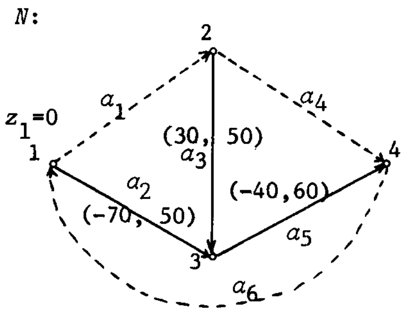

Fig.4

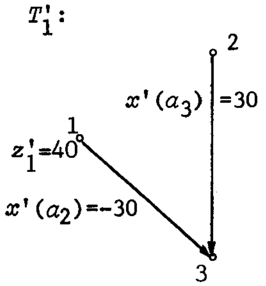

Fig.5

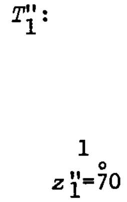

Fig.6

newly values $x'(a)$ $(a\varepsilon A(T_1'))$ and z_1' in Fig.5. We apply the above process to T_1' again. Since $x'(a_2)=-30$, we have $G'=(\{2,3\},\ \{a_3\})\varepsilon\ \mathcal{D}^+$ and $\tilde{d}(G')=30$. Let z_1'' be the value obtained by ADJUST $(T_1',a_2,(x'(a):a\varepsilon A(T_1')),z_1')$, then we have $z_1''=70$ as in Fig.6. When (P1) is repeated for $T_1''=T_1'-\{2,3\}$, we have $T_1''\varepsilon\ \mathcal{D}^-$ and $\tilde{d}(T_1'')=70$. These are summarized as follows: $\mathcal{D}^+=\{G,G'\}$, $\mathcal{D}^-=\{T_1''\}$, $\mathcal{N}=\phi$.
By JOIN-operations, we have $\mathcal{D}^+=\{G\cup G'+a_5\}$, $\mathcal{D}^-=\{T_1''\}$, $\mathcal{N}=\phi$.
The deficiencies are given by

$\tilde{d}(G)=40$, $\tilde{d}(G')=30$, $\tilde{d}(T_1'')=70$,

$d(G\cup G'+a_5)=\tilde{d}(G)+\tilde{d}(G')=70$, $d(T_1'')=\tilde{d}(T_1'')=70$.

Hence, the deficiency $d(\mathcal{D})$ of the circulation x in network N is 140.
Note that the deficiency $d(\mathcal{D})$ is determined by the rooted spanning tree T_1.

3.3. Description of Algorithm

We introduce the following Trimming operation as in [8] which improves the efficiency of the dual simplex method.

3.3.1. Trimming Operation

As will be seen later, we choose an arc a^*, as a pivot operation, such that $B_{a^*}=Q$ for some $Q\varepsilon\mathcal{D}$. Such an arc a^* is called a good arc. In this section, we show that some rooted tree with good arcs can be constructed, if necessary, by Trimming operation given below.

Let T_r be a rooted tree with no good arcs, then we can find $Q\varepsilon\mathcal{D}$ such that $Q_i\varepsilon\mathcal{N}$ $(2\leq i\leq k)$ where Q_i $(1\leq i\leq k)$ are components of $T_r-V(Q)$ satisfying $r\varepsilon V(Q_1)$. Such a Q is called an $\mathcal{N}$-surrounded graph while each Q_i $(2\leq i)$ a surrounding $\mathcal{N}$-graph of Q. If we can remove all the surrounding $\mathcal{N}$-graphs by pivot operations, then we have good arcs. Consider a pivot operation for arc a^i such that Q_i $(i\geq 2)$ equals a leaf branch B_{a^i}. Let a^* be the tree arc of T_r connecting Q_1 to Q and $\tilde{T}_r$ the new rooted spanning tree obtained after the pivot operation. Now, we check whether a^* is the good arc of $\tilde{T}_r$ or not. If a^* is a good arc, then we stop here. Otherwise we search for the $\mathcal{N}$-surrounded graph $\tilde{Q}$ of $\tilde{T}_r$ incident to a^* and a surrounding $\mathcal{N}$-graph $\tilde{Q}_i$ of $\tilde{Q}$. Then we continue the pivot operation for the arc $\tilde{a}^i$ such that $\tilde{Q}_i$ equals some leaf branch $\tilde{B}_{\tilde{a}^i}$ of $\tilde{T}_r$. That is, we repeat this process until the arc a^* becomes the good arc of the current rooted tree. This procedure is called Trimming operation, implemented so that we may have good arcs. Here, we have the next two problems to be considered.

(3.3) Is the number of pivot operations in Trimming operation finite?

(3.4) Can the pivot operations themselves be carried out?

First, we consider (3.4). Let $Q\varepsilon\mathcal{D}$ be an $\mathcal{N}$-surrounded graph and $B_{a'}$

the leaf branch of T_r which equals a surrounding $\mathcal{N}$-graph of Q. Then we have the next proposition, where p, x are the potential, the circulation obtained in Section 2.1, respectively.

Proposition 3.1. If $Q\varepsilon\mathcal{D}^{\pm}$, then we have an arc a_ρ and

$$x(a_\rho)=\begin{cases} b(a_\rho) & (\ \partial^{\pm}a_\rho\varepsilon V(B_{a'})\ ), \\ 0 & (\ \partial^{\mp}a_\rho\varepsilon V(B_{a'})\ ). \end{cases}$$

Proof: We only prove the case when $Q\varepsilon\mathcal{D}^{+}$.

Case 1: $a'\varepsilon A^{+}(T_r)$; Since $B_{a'}\varepsilon\mathcal{N}$ and $Q\varepsilon\mathcal{D}^{+}$, we have $x(a')=b(a')$ and $c(a')+p(\partial^{+}a')-p(\partial^{-}a')=0$. $p(\partial^{-}a')$ must be increased in order to satisfy $\tilde{x}(a')=b(a')$ for a new circulation $\tilde{x}$. This means that we may increase the potential of each vertex of $B_{a'}$. Hence, Operation 2 $(a', S^{+}_{a'}, T^{-}_{a'})$ is performed. If $S^{+}_{a'}\cup T^{-}_{a'}=\phi$, then we have

(3.5) $\qquad x(a)=b(a) \qquad (\ a\varepsilon H^{+}_{a'}\ ),$

(3.6) $\qquad x(a)=0 \qquad (\ a\varepsilon H^{-}_{a'}\ ).$

On the other hand, since x is a circulation, we have

(3.7) $\qquad \Sigma\{x(a):a\varepsilon H^{+}_{a'}\}+x(a')=\Sigma\{x(a):a\varepsilon H^{-}_{a'}\}.$

Rearranging the equation (3.7) by substituting (3.5) and (3.6), it follows that $b(a)=0$ for any $a\varepsilon H^{+}_{a'}\cup\{a'\}$, but this is impossible.

Case 2: $a'\varepsilon A^{-}(T_r)$; We have $H^{+}_{a'}=\phi$ as in Case 1. Though $H^{-}_{a'}\cup\{a'\}$ is a directed cut, this contradicts the assumption that the underlying graph of the network N is strongly connected. □

3.3.2. Algorithm

We present the algorithm for the problem (P), which is the dual simplex method accompanied by Tree Partitionings and Trimming operations.

Algorithm

Step 0: (Initialization)

(0.i): Find a spanning tree T_r with root r of the network N.

(0.ii): Determine the potential p by Procedure POT(T_r), and calculate the circulation x by Procedure FLO(p,T_r).

Step 1: (Implementation of Tree Partitioning)

(1.i): Find $\underline{\mathcal{D}}^{+}$, $\underline{\mathcal{D}}^{-}$ and $\underline{\mathcal{N}}$ by Procedure TP(T_r,$(x(a):a\varepsilon A(T_r))$).

(1.ii): Find $\mathcal{D}^{+}$, $\mathcal{D}^{-}$ and $\mathcal{N}$ by the corresponding JOIN-operations.

Step 2: (Preparation for Pivot Operations)

(2.i): If $\mathcal{D}=\mathcal{D}^{+}\cup\mathcal{D}^{-}$ is empty, then the algorithm terminates and x is an optimal solution.

(2.ii): If there exist no leaf branches Q of T_r such that $Q\varepsilon\mathcal{D}$, then go to Step 5.

Step 3: (Pivot Operation)

(3.i): Find a leaf branch $B_{a^*} \varepsilon \mathcal{D}$ of T_r.

(3.ii): For $B_{a^*} \varepsilon \mathcal{D}^{\pm}$, find a_ρ and ρ by Operation 2 (a^*, $S^{\mp}_{a^*}$, $T^{\pm}_{a^*}$).

(3.iii): Renew potential p by Operation 1 (p,T_r,ρ,a^*). Set $T_r=T_r-a^*+a_\rho$ and renew circulation x by Procedure FLO (p,T_r).

Step 4: (Renewal of Tree Partitioning)

Find $\mathcal{D}^+$, $\mathcal{D}^-$ and $\mathcal{N}$ for the new tree T_r in the same way as in Step 1. Then go to Step 2.

Step 5: (Trimming Operation)

(5.i): Choose an $\mathcal{N}$-surrounded graph $Q \varepsilon \mathcal{D}$ and its surrounding $\mathcal{N}$-graph Q_i such that leaf branch $B_{a''}=Q_i$ for some a''.

(5.ii): Find the arc $a^* \varepsilon A(T_r)$ which connects Q to the component of $T_r-V(Q)$ containing root r.

(5.iii): For $Q \varepsilon \mathcal{D}^{\pm}$, find a_ρ and ρ by Operation 2 (a'', $S^{\pm}_{a''}$, $T^{\mp}_{a''}$). Renew potential p by replacing a^* with a'' in (3.iii). (The circulation x is unchanged.)

(5.iv): We repeat Tree Partitioning for the new tree $T_r=T_r-a''+a_\rho$ and decide $\mathcal{D}^+$, $\mathcal{D}^-$ and $\mathcal{N}$ as in Step 1.

(5.v): If arc a^* is good, then go to (3.ii). Otherwise, for the new tree T_r find both the $\mathcal{N}$-surrounded graph Q incident to a^* and its surrounding $\mathcal{N}$-graph Q_i such that leaf branch $B_{a''}=Q_i$ for some a''. Then go to (5.iii).

4. Estimation of the Computational Complexity of the Algorithm

4.1. Change of the Deficiency and the Classes of Tree Partitioning

After a pivot operation, if we repeat Tree Partitioning for the new tree $\tilde{T}_r$, then the classes and the deficiency $d(\mathcal{D})$ will be different from those of T_r in general. Our aim in this section is to show that $d(\mathcal{D})$ does not increase in course of the algorithm and to examine the change of the classes by pivot operations. We achieve this aim in the following two subsections (4.1.1) and (4.1.2) separately.

4.1.1. Pivot Operations in Trimming Operations

It is easy to see that $d(\mathcal{D})$ is unchanged by Trimming operations. Hence, the problem to be considered here is to estimate the number of consecutive Trimming operations. Let $\underline{\tilde{\mathcal{D}}}^+$, $\underline{\tilde{\mathcal{D}}}^-$ and $\underline{\tilde{\mathcal{N}}}$ be the classes of Tree Partitioning for $\tilde{T}_r$, and $\tilde{\mathcal{D}}^+$, $\tilde{\mathcal{D}}^-$ and $\tilde{\mathcal{N}}$ those obtained by the corresponding JOIN-operations. For the two arcs a'' and a_ρ chosen in (5.i) and (5.iii) of Step 5, define vertices $u,y \varepsilon V-V(B_{a''})$ and $v,w \varepsilon V(B_{a''})$ by $\{u,v\}=\{\partial^+a'', \partial^-a''\}$,

$\{w,y\}=\{\partial^+a_\rho,\ \partial^-a_\rho\}$. (See Fig.7.) Here, assume that $Q\varepsilon\mathcal{D}$ is an $\mathcal{N}$-surrounded graph and $B_{a''}$ is its surrounding $\mathcal{N}$-graph. The possibility of consecutive Trimming operations consists of the next two cases:

Case C1: $y\varepsilon V(B_{a'})$ for some surrounding $\mathcal{N}$-graph $B_{a'}$ of Q. ($B_{a''}$ is attached to $B_{a'}$ by arc a_ρ in $\tilde{T}_r$.)

Case C2: $y\varepsilon V(Q)$. ($B_{a''}$ is attached to Q by arc a_ρ in $\tilde{T}_r$.)

In other cases, any surrounding $\mathcal{N}$-graph is to be removed from Q in $\tilde{T}_r$. (See Fig.8.) The following proposition, which is easily proved, shows the result for Case C1.

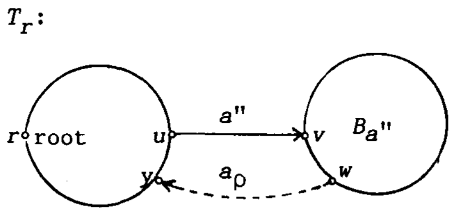

Fig.7

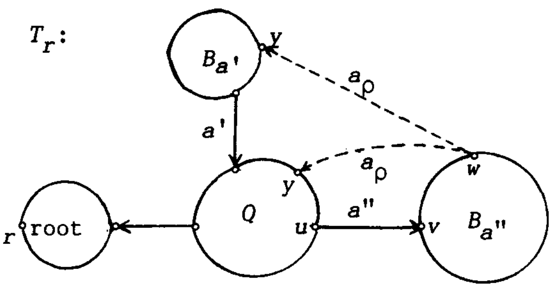

Fig.8

Proposition 4.1. In Case C1, we have

$H_0\varepsilon\tilde{\mathcal{N}}$, $\mathcal{D}=\tilde{\mathcal{D}}$ and $\mathcal{N}-\{B_{a''},\ B_{a'}\}=\tilde{\mathcal{N}}-\{H_0\}$, where $H_0=B_{a''}\cup B_{a'}+a_\rho$. □

Before proving Case C2, first note the following fact:

$\{Q'\varepsilon\underline{\mathcal{N}}:V(Q')\cap V(P_{vw})=\phi\}=\{\tilde{Q}\varepsilon\underline{\tilde{\mathcal{N}}}:V(\tilde{Q})\cap V(\tilde{P}_{vw})=\phi\}$,

where P_{vw} ($\tilde{P}_{vw}$) is the path of T_r ($\tilde{T}_r$), respectively.

Let $T'_r=T_r-\cup\{V(Q'):Q'\varepsilon\underline{\mathcal{N}},V(Q')\cap V(P_{vw})=\phi\}$ and consider T'_r in place of T_r for Case C2. Let $\tilde{T}'_r$ be the new rooted tree given by the pivot operation of T'_r, and $\mathcal{D}^+,\mathcal{D}^-,\mathcal{N}$($\tilde{\mathcal{D}}^+,\tilde{\mathcal{D}}^-,\tilde{\mathcal{N}}$) the partition classes obtained from T'_r ($\tilde{T}'_r$), respectively. Then for an $\mathcal{N}$-surrounded graph Q of T'_r and its surrounding $\mathcal{N}$-graph which equals leaf branch $B'_{a''}$ of T'_r for some a'', we have the following proposition.

Proposition 4.2. For $Q\varepsilon\mathcal{D}^\pm$ in Case C2, we have either (1) or (2).

(1) $H_1\varepsilon\tilde{\mathcal{D}}^\pm$, $\mathcal{D}^\pm-\{Q\}=\tilde{\mathcal{D}}^\pm-\{H_1\}$ and $\mathcal{N}-\{B'_{a''}\}=\tilde{\mathcal{N}}$,

(2) For some leaf branch $\tilde{B}'_e\subsetneq B'_{a''}$ of $\tilde{T}'_r$,

$H_2\varepsilon\tilde{\mathcal{D}}^\pm$, $\mathcal{D}^\pm-\{Q\}=\tilde{\mathcal{D}}^\pm-\{H_2\}$ and $\mathcal{N}-\{B'_{a''}\}=\tilde{\mathcal{N}}-\{\tilde{B}'_e\}$,

where $H_1=B'_{a''}\cup Q+a_\rho$ and $H_2=(B'_{a''}-V(\tilde{B}'_e))\cup Q+a_\rho$.

Proof: We only consider the case when an $\mathcal{N}$-surrounded graph $Q\varepsilon\mathcal{D}^+$,

$A(T_r')=A^-(T_r')$ and $\partial^+a_\rho=y$. Let $B_{a''}'$ be the leaf branch as to some a'' and T_r' such that $B_{a''}'$ equals a surrounding $\mathcal{N}$-graph of Q, and define X by $X=\{a\varepsilon A^+(\tilde{T}_r')\cap A(\tilde{P}_{vy}):x(a)=b(a)\}$ where $x(a)$ is the value given in Step (5.iii). By JOIN-operations, if $X=\phi$, then Case (1) is obtained. Otherwise we have Case (2). Note that $B_{a''}'\neq\tilde{B}_e'$ since $a_\rho\notin X$, where the arc $e\varepsilon X$ is the nearest to vertex w. □

We prove the next proposition, based on Propositions 4.1 and 4.2.

Proposition 4.3. Steps (5.iii)~(5.v) in Trimming operation are repeated at most $2|V|-4$ times.

Proof: Define p, $\tilde{p}$, q and $\tilde{q}$ as follows.

$p=\Sigma\{|V(Q)|:Q\varepsilon\mathcal{N}\}$, $q=|\mathcal{N}|$, $\tilde{p}=\Sigma\{|V(\tilde{Q})|:\tilde{Q}\varepsilon\tilde{\mathcal{N}}\}$ and $\tilde{q}=|\tilde{\mathcal{N}}|$.

We only prove the case where an $\mathcal{N}$-surrounded graph Q is in $\mathcal{D}^+$. After one pivot operation, from Propositions 4.1 and 4.2 we have $p=\tilde{p}$, $q-1=\tilde{q}$ for Case C1, while Case C2 shows $p>\tilde{p}$, $q\geqq\tilde{q}$. It is easy to see $|V|-2\geqq p\geqq q$, and that p or q does not increase after each pivot operation. If the value $p+q$ vanishes, then Trimming operation is over. So Steps (5.iii)~(5.v) are repeated at most $2|V|-4$ pivot operations. □

4.1.2. Pivot Operations except Trimming Operations

The next proposition can be proved similarly as Proposition 3.1.

Proposition 4.4. For the leaf branch $B_{a^*}\varepsilon\mathcal{D}^\pm$ of T_r chosen in Step 3, we have an arc a_ρ and the circulation x such that

$$x(a_\rho)=\begin{cases} 0 & (\ \partial^\pm a_\rho\varepsilon V(B_{a^*})\ ), \\ b(a_\rho) & (\ \partial^\mp a_\rho\varepsilon V(B_{a^*})\ ). \end{cases}$$ □

After doing a pivot operation except Trimming operations, the circulation changes on the cycle formed by T_r and a_ρ. For $v',v''\varepsilon V$, let $P_{v'v''}$ ($\tilde{P}_{v'v''}$) be the path of T_r ($\tilde{T}_r$), respectively. Assume that vertex v^* ($v^*\varepsilon V(P_{ru})\cap V(P_{ry})$) is the farthest from root r on P_{ry}. (See Fig.9.) Then the following proposition is easily shown.

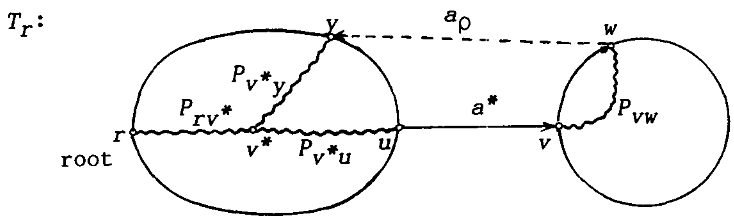

Fig.9

Proposition 4.5. $\{Q'\varepsilon\,\underline{\mathcal{D}}^{\pm}\cup\underline{\mathcal{N}}:V(Q')\cap V^{*}=\phi\}=\{\tilde{Q}\varepsilon\,\tilde{\underline{\mathcal{D}}}^{\pm}\cup\tilde{\underline{\mathcal{N}}}:V(\tilde{Q})\cap\tilde{V}^{*}=\phi\}$, where $V^{*}=V(P_{rw})\cup V(P_{v^{*}y})$ and $\tilde{V}^{*}=V(\tilde{P}_{rv})\cup V(\tilde{P}_{v^{*}u})$. □

From Proposition 4.5, consider the graph obtained by removing $Q'\varepsilon\mathcal{S}$ such that $V(Q')\cap V^{*}=\phi$, where $\mathcal{S}=\underline{\mathcal{D}}^{+}\cup\underline{\mathcal{D}}^{-}\cup\underline{\mathcal{N}}$. That is, let us execute the next operation CUT $(T_r,a^{*},a_{\rho},V^{*})$ (or simply CUT-operation) by using the circulation x obtained in Section 2.1 and $z_r=0$.

CUT $(T_r,a^{*},a_{\rho},V^{*})$:

While there is a leaf branch $B_{a'}\varepsilon\mathcal{S}$ of T_r such that $V(B_{a'})\cap V^{*}=\phi$ for some a', then carry out ADJUST$(T_r,a',(x(a):a\varepsilon A(T_r)),z_r)$ and set $\mathcal{S}=\mathcal{S}-\{B_{a'}\}$. Otherwise, we stop this operation.

When we apply CUT-operation to both T_r and $\tilde{T}_r$, we finally have two some rooted subtrees and renewed values, i.e., let

(4.1) T_r^0, $(x^0(a):a\varepsilon A(T_r^0))$ and z_r^0 be the rooted subtree and the values determined after doing CUT$(T_r,a^{*},a_{\rho},V^{*})$,

(4.2) $\tilde{T}_r^0$, $(\tilde{x}^0(a):a\varepsilon A(\tilde{T}_r^0))$ and $\tilde{z}_r^0$ be the rooted subtree and the values determined after doing CUT$(\tilde{T}_r,a^{*},a_{\rho},\tilde{V}^{*})$.

Now, we are ready to observe the change of the classes and the deficiencies $d(\mathcal{D})$, $d(\tilde{\mathcal{D}})$ between the new trees T_r^0 and $\tilde{T}_r^0$, in place of T_r and $\tilde{T}_r$. For the leaf branch $B_{a*}^0\varepsilon\,\mathcal{D}^{\pm}$ of T_r^0, let $B_i^0\subset B_{a*}^0$ $(1\leq i\leq k)$ be the members of $\underline{\mathcal{D}}^{\pm}$ such that these B_i^0 are ordered in such a way that $w\varepsilon V(B_1^0)$ and that B_{i+1}^0 appears after B_i^0 toward root r. (a^{*} is the arc chosen in (3.i) of Step 3 in Section 3.3.2.) Let $a_i\varepsilon A(T_r^0)$ be the arc joined by B_i^0 and B_{i+1}^0, where we define $a_k=a^{*}$. For each a_i in B_{a*}^0 of $\mathcal{D}^{\pm}$ and the value $x^0(a_i)$ defined in (4.1), let

$$\Delta_i=\begin{cases}-x^0(a_i) & \text{if } a_i\varepsilon A^{\pm}(T_r^0),\\ x^0(a_i)-b(a_i) & \text{if } a_i\varepsilon A^{\mp}(T_r^0),\end{cases}\tag{4.3}$$

and $\Delta_0=0$. Then the following proposition, which is easily proved, characterizes each B_i^0.

Proposition 4.6. We have $B_i^0\varepsilon\,\underline{\mathcal{D}}^{\pm}$ $(1\leq i\leq k)$ for the leaf branch $B_{a*}^0\varepsilon\,\mathcal{D}^{\pm}$ of T_r^0 if and only if the next conditions are satisfied.

$$\begin{array}{ll}
x^0(a_i)+\Delta_{i-1}<0 & \text{if } a_i\varepsilon A^{\pm}(T_r^0),\\
x^0(a_i)-\Delta_{i-1}>b(a_i) & \text{if } a_i\varepsilon A^{\mp}(T_r^0),\\
0\leq x^0(a)+\Delta_{i-1}<b(a) & \text{if } a\varepsilon A^{\pm}(T_r^0)\cap A(B_i^0)\cap A(P_{vw}^0),\\
0<x^0(a)-\Delta_{i-1}\leq b(a) & \text{if } a\varepsilon A^{\mp}(T_r^0)\cap A(B_i^0)\cap A(P_{vw}^0),\\
0\leq x^0(a)<b(a) & \text{if } a\varepsilon A^{\pm}(T_r^0)\cap A(B_i^0)-A(P_{vw}^0),\\
0<x^0(a)\leq b(a) & \text{if } a\varepsilon A^{\mp}(T_r^0)\cap A(B_i^0)-A(P_{vw}^0),
\end{array}$$

where P^0_{vw} is the path of T^0_r and $x^0(a)$ $(a\varepsilon A(T^0_r))$ appearing above are the values defined in (4.1). Moreover we have $\tilde{d}(B^0_i)=\Delta_i-\Delta_{i-1}>0$ and $d(B^0_{a^*})=\Delta_k$. □

The following proposition, which is easily proved, shows the relation between $(x^0(a):a\varepsilon A(B^0_{a^*}))$ and $(\tilde{x}^0(a):a\varepsilon A(\tilde{B}^0_{a_\rho}))$ defined in (4.1) and (4.2), where $B^0_{a^*}$ ($\tilde{B}^0_{a_\rho}$) is the leaf branch as to T^0_r and arc a^* ($\tilde{T}^0_r$ and arc a_ρ).

Proposition 4.7. For the leaf branch $B^0_{a^*}\varepsilon\mathcal{B}^\pm$ of T^0_r, we have

(1) $\tilde{x}^0(a)=x^0(a)-\Delta_k$ if $a\varepsilon A^\pm(\tilde{T}^0_r)\cap A(\tilde{B}^0_{a_\rho})\cap A(\tilde{P}^0_{vy})$,

$\tilde{x}^0(a)=x^0(a)+\Delta_k$ if $a\varepsilon A^\mp(\tilde{T}^0_r)\cap A(\tilde{B}^0_{a_\rho})\cap A(\tilde{P}^0_{vy})$,

(2) $\tilde{x}^0(a)<b(a)$ if $a\varepsilon A^\pm(\tilde{T}^0_r)\cap A(\tilde{B}^0_{a_\rho})$,

$\tilde{x}^0(a)>0$ if $a\varepsilon A^\mp(\tilde{T}^0_r)\cap A(\tilde{B}^0_{a_\rho})$,

where $\tilde{P}^0_{vy}$ denotes the path of $\tilde{T}^0_r$. □

First, we consider how the leaf branch $\tilde{B}^0_{a_\rho}$ of $\tilde{T}^0_r$ is partitioned by Tree partitioning algorithm. For leaf branch $B^0_{a^*}$ in $\mathcal{B}^\pm$ of T^0_r, let

(4.4) $U^\pm=\{a\varepsilon A^\mp(\tilde{T}^0_r)\cap A(\tilde{P}^0_{vy}):\tilde{x}^0(a)>b(a)\}\cup\{a\varepsilon A^\pm(\tilde{T}^0_r)\cap A(\tilde{P}^0_{vy}):\tilde{x}^0(a)<0\}$,

where $\tilde{x}^0(a)$ $(a\varepsilon A(\tilde{T}^0_r))$ is the value defined in (4.2). We assume that $U^\pm$ is not empty in the future discussion. Let $a^1\varepsilon U^\pm$ be the arc which is the nearest to vertex v on the path $\tilde{P}^0_{vy}$ of $\tilde{T}^0_r$, and

$$(4.5)\qquad \Delta_{a^1}=\begin{cases} b(a^1)-x^0(a^1) & \text{if } a^1\varepsilon A^\mp(\tilde{T}^0_r),\\ x^0(a^1) & \text{if } a^1\varepsilon A^\pm(\tilde{T}^0_r),\end{cases}$$

where the value $x^0(a^1)$ is defined in (4.1). Then we see that the leaf branch $\tilde{B}^0_{a^1}$ of $\tilde{T}^0_r$ is in $\tilde{\underline{\mathcal{B}}}^\pm$ at (P2). After doing ADJUST-operation in (P4), we have the values $\tilde{x}^1(a)$ $(a\varepsilon A(\tilde{T}^1_r))$ for $\tilde{T}^1_r=\tilde{T}^0_r-V(\tilde{B}^0_{a^1})$ such that the following proposition shows.

Proposition 4.8. For the leaf branch $B^0_{a^*}\varepsilon\mathcal{B}^\pm$ of T^0_r and nonempty set $U^\pm$, we have the following (1), (2) and (3).

(1) $\Delta_k>\Delta_{a^1}>0$.

(2) $\tilde{x}^1(a)=x^0(a)-\Delta_{a^1}$ if $a\varepsilon A^\pm(\tilde{T}^1_r)\cap A(\tilde{B}^0_{a_\rho})\cap A(\tilde{P}^1_{vy})$,

$\tilde{x}^1(a)=x^0(a)+\Delta_{a^1}$ if $a\varepsilon A^\mp(\tilde{T}^1_r)\cap A(\tilde{B}^0_{a_\rho})\cap A(\tilde{P}^1_{vy})$,

(3) $\tilde{x}^1(a)<b(a)$ if $a\varepsilon A^\pm(\tilde{T}^1_r)\cap A(\tilde{B}^0_{a_\rho})$,

$\tilde{x}^1(a)>0$ if $a\varepsilon A^\mp(\tilde{T}^1_r)\cap A(\tilde{B}^0_{a_\rho})$,

where $\tilde{P}^1_{vy}$ is the path of $\tilde{T}^1_r$ and $x^0(a)$ $(a\varepsilon A(\tilde{T}^1_r))$ appearing above is the value defined in (4.1).

Proof: We only consider the case when $B^0_{a^*}\varepsilon\mathcal{B}^+$ and $A(T^0_r)=A^+(T^0_r)$. We check Case (2) first. From $a^1\varepsilon U^+$, Propositions 4.6 and 4.7, we have $\Delta_k>\Delta_{a^1}$. Let $\tilde{x}^1(a)$ $(a\varepsilon A(\tilde{T}^1_r))$ be the values obtained by ADJUST$(\tilde{T}^0_r,a^1,(\tilde{x}^0(a):a\varepsilon A(\tilde{T}^0_r))$,

$\tilde{z}_r^0$) where $\tilde{T}_r^1=\tilde{T}_r^0-V(\tilde{B}_a^0 1)$ for the leaf branch $\tilde{B}_a^0 1$ of $\tilde{T}_r^0$. Then from Proposition 4.7 we have

$$\tilde{x}^1(a)=\tilde{x}^0(a)-(\Delta_k-\Delta_a 1)=x^0(a)+\Delta_a 1 \quad (a\varepsilon A(\tilde{P}_{vy}^1)\cap A(\tilde{B}_{a_\rho}^0)),$$

where $\tilde{x}^0(a)$ (resp. $x^0(a)$) is the value defined in (4.2) (resp. (4.1)) and $\tilde{P}_{vy}^1$ is the path of $\tilde{T}_r^1$. When we deal with (3), it suffices to see the arcs of $\tilde{P}_{vy}^1$. The following three cases can be considered with respect to a^1:

Case A: $a^1=a_\rho$,

Case B: $a^1=a_j$ for some a_j joining B_j^0 and B_{j+1}^0,

Case C: $a^1\varepsilon A(B_j^0)\cap A(\tilde{P}_{vw}^0)$ for some j and the path $\tilde{P}_{vw}^0$ of $\tilde{T}_r^0$.

We examine Case B only. From Proposition 4.6 and $\Delta_a 1<\Delta_k$, we have Case (1) of this proposition, and for each arc $a\varepsilon A(B_i^0)\cap A(\tilde{P}_{vw}^1)$ $(i\leqq j)$,

$$\tilde{x}^1(a)=x^0(a)+\Delta_a 1\geqq x^0(a)+\Delta_i>x^0(a)+\Delta_{i-1}\geqq 0,$$

where $x^0(a)$, Δ_i are defined in (4.1), (4.3), respectively. Similarly, we have $\tilde{x}^1(a_i)>0$ for arc a_i $(i\leqq j-1)$ joining B_i^0 and B_{i+1}^0. □

The following proposition is the result of Tree Partitioning for the leaf branch $\tilde{B}_{a_\rho}^0$ of $\tilde{T}_r^0$. Let a^i be the arc for which Tree Partitioning is implemented at the i-th repetition.

Proposition 4.9. If Tree Partitioning is executed for $\tilde{B}_{a_\rho}^0$, then we have the following (1) or (2) for some integer n and the leaf branch $B_a^0{*}\varepsilon\mathcal{D}^\pm$ of T_r^0.

(1) $\quad \tilde{B}_a^0 1\varepsilon\, \underline{\tilde{\mathcal{D}}}^\pm$,

(2) $\quad \tilde{B}_a^0 i{+}1-V(\tilde{B}_a^0 i)\varepsilon\, \underline{\tilde{\mathcal{D}}}^\pm \quad (1\leqq i\leqq n-1)$,

where $\tilde{B}_a^0 i$ $(1\leqq i\leqq n)$ are leaf branches of $\tilde{T}_r^0$.

Proof: We have already checked Case (1). Concerning Case (2), we apply Propositions 4.7 and 4.8 repeatedly. □

After (P4) of Tree partitioning algorithm has been done by the arc $a^n\varepsilon A(\tilde{B}_{a_\rho}^0)$ chosen finally in (P2) with respect to $\tilde{B}_{a_\rho}^0$, do the algorithm on the path between w and v^* (u and v^*) of T_r^0 (the current rooted subtree of $\tilde{T}_r^0$), respectively. That is, let

(4.6) $\quad \tilde{T}_r^n=\tilde{T}_r^0-V(\tilde{B}_a^0 n)-\cup\{V(\tilde{Q}):\tilde{Q}\varepsilon\,\underline{\tilde{\mathcal{D}}}^+\cup\underline{\tilde{\mathcal{D}}}^-\cup\underline{\tilde{\mathcal{N}}},\ V(\tilde{P}_{uv}^0{*})\cap V(\tilde{Q})=\phi\}$,

(4.7) $\quad T_r^0=T_r^0-\cup\{V(Q):Q\varepsilon\,\underline{\mathcal{D}}^+\cup\underline{\mathcal{D}}^-\cup\underline{\mathcal{N}},\ V(P_{wv}^0{*})\cap V(Q)=\phi\}$,

where $\tilde{P}_{uv}^0{*}$ ($P_{wv}^0{*}$) is the path of $\tilde{T}_r^0$ (the old T_r^0), respectively and $\tilde{B}_a^0 n$ is the leaf branch of $\tilde{T}_r^0$.

For the leaf branch $B_a^0{*}\varepsilon\mathcal{D}^\pm$ of the old T_r^0, let

$$(4.8)\qquad \Delta_a n=\begin{cases} b(a^n)-x^0(a^n) & (a^n\varepsilon A^\mp(\tilde{T}_r^0)),\\ x^0(a^n) & (a^n\varepsilon A^\pm(\tilde{T}_r^0)),\end{cases}$$

where $x^0(a^n)$ is the value defined in (4.1). We assume that

(4.9) $x^0(a)$ $(a\varepsilon A(T_r^0))$ and z_r^0 are the values determined by the repetition of (P4) until the new T_r^0 is obtained,

(4.10) $\tilde{x}^n(a)$ $(a\varepsilon A(\tilde{T}_r^n))$ and $\tilde{z}_r^n$ are the values determined by the repetition of (P4) until $\tilde{T}_r^n$ is obtained.

Note that we compare the new T_r^0 with $\tilde{T}_r^n$ from now on and that for the new T_r^0, the values $(x^0(a):a\varepsilon A(T_r^0))$ in (4.1) differ from those in (4.9) on the path of the new T_r^0 between v^* and r.

We continue Tree Partitioning for $Q'\varepsilon\mathcal{D}\cup\mathcal{N}$ such that $y\varepsilon V(Q')$. If Q' is in $\mathcal{D}^+$ (resp. $\mathcal{D}^-$), then let $B_i'\subset Q'$ $(1\leq i\leq m)$ be the members of $\underline{\mathcal{D}}^+$ (resp. $\underline{\mathcal{D}}^-$) such that they are ordered so that B_{i+1}' appears after B_i' toward root r (we see that $y\varepsilon V(B_1')$.). We denote by a_i' the arc connecting B_{i+1}' to B_i', where $a_m'=a'$ for some leaf branch $B_{a'}^0=Q'$ of T_r^0. Similarly, we also define B_i' and a_i' for $Q'\varepsilon\mathcal{N}$. In the example of Fig.10, we have $m=3$ and $Q'=B_{a'}^0$. In this figure, Q' is composed of subtrees B_i' $(1\leq i\leq 3)$ of Q' and two tree arcs a_1', a_2'.

$\tilde{T}_r^n$:

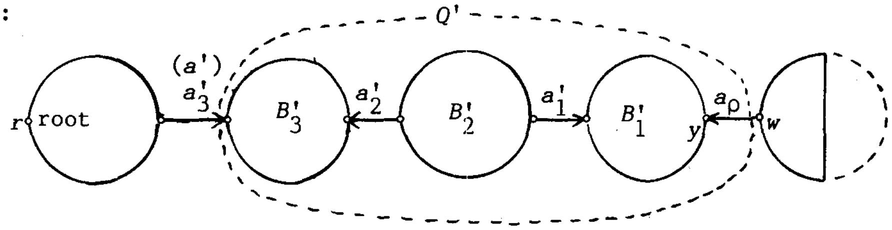

Fig.10

First assuming $Q'\varepsilon\mathcal{D}^\pm$ as to the leaf branch $B_{a^*}^0\varepsilon\mathcal{D}^\pm$ of the rooted subtree in (4.1), we have the following proposition.

Proposition 4.10. If $B_{a^*}^0\varepsilon\mathcal{D}^\pm$ and $Q'\varepsilon\mathcal{D}^\pm$, then we have

(1) $B_{a^*}^0\cup Q'+a_\rho\varepsilon\tilde{\mathcal{D}}^\pm$,

(2) $\mathcal{D}^\pm-\{B_{a^*}^0, Q'\}=\tilde{\mathcal{D}}^\pm-\{B_{a^*}^0\cup Q'+a_\rho\}$, $\quad\mathcal{D}^\mp=\tilde{\mathcal{D}}^\mp$.

Proof: We only check the case where $B_{a^*}^0\varepsilon\mathcal{D}^+$ and $A(T_r^0)=A^-(T_r^0)$. Let $(x^0(a):a\varepsilon A(T_r^0))$, z_r^0 (resp. $(\tilde{x}^n(a):a\varepsilon A(\tilde{T}_r^n))$, $\tilde{z}_r^n$) be the values defined in (4.9) (resp. (4.10)). If B_1' $(B_1'\subset Q')$ equals some rooted subtree T_r', then from $B_1'\varepsilon\underline{\mathcal{D}}^+$ we have

(4.11) $\tilde{z}_r^n=z_r^0-\Delta_a n<0.$

Note that we have $z_r^0<0$ in (4.11). Otherwise, from $x^0(a_1')>b(a_1')$ and Proposition 4.6, it follows that

(4.12) $\tilde{x}^n(a_1')=x^0(a_1')+\Delta_a n>b(a_1').$

Note that we have the arc a_1' and $\tilde{x}^n(a)=x^0(a)+\Delta_a n>0$ $(a\varepsilon A(B_1')\cap A(\tilde{P}^n_{yr}))$ for the path $\tilde{P}^n_{yr}$ of $\tilde{T}^n_r$. Let $X=\{a\varepsilon A(B_1')\cap A(\tilde{P}^n_{yr}):\tilde{x}^n(a)>b(a)\}$. If X is empty, then we easily have (1) and (2) of this proposition. Let $e^1\varepsilon X$ be the arc which is the nearest to vertex y. In (P2) of Tree partitioning algorithm, we have

(4.13) $\quad \tilde{B}^n_{e^1}\varepsilon\ \tilde{\mathcal{D}}^+$,

where $\tilde{B}^n_{e^1}$ is the leaf branch of $\tilde{T}^n_r$. Define Δ_{e^1} by replacing a^n with e^1 in (4.8), and we have $\Delta_{e^1}\geq 0$ from Proposition 4.6. If $\Delta_{e^1}=0$, then we have $B_1'-V(\tilde{B}^n_{e^1})\varepsilon\ \tilde{\mathcal{D}}^+$ and that other members in the classes of $\tilde{T}^n_r$ are the same as those of T^0_r. If $\Delta_{e^1}>0$, then we see that every time (P2) is executed, a member D of $\tilde{\mathcal{D}}^+$ is derived and that there exist no members of $\tilde{\mathcal{N}}$ obtained from D. Finally, we have this proposition by JOIN-operations. □

Second, assume that Q' is in $\mathcal{N}$ with respect to the leaf branch $B^0_{a^*}\varepsilon\,\mathcal{D}^{\pm}$ of (the old) T^0_r. We see that Q' equals either the rooted new tree T^0_r or some leaf branch $B^0_{a'}$ of (the new) T^0_r. If Q' equals T^0_r, then we have the following proposition.

Proposition 4.11. For $B^0_{a^*}\varepsilon\,\mathcal{D}^{\pm}$ and $Q'=T^0_r\varepsilon\,\mathcal{N}$, we have either (1) or (2):

(1) For some rooted subtree T^3_r of T^0_r such that $P^0_{yr}\subset T^3_r$, we have

$$B^0_{a^*}\cup T^3_r+a_\rho\varepsilon\,\tilde{\mathcal{D}}^{\pm},\quad \mathcal{D}^{\pm}-\{B^0_{a^*}\}=\tilde{\mathcal{D}}^{\pm}-\{B^0_{a^*}\cup T^3_r+a_\rho\},\quad \mathcal{D}^{\mp}=\tilde{\mathcal{D}}^{\mp}.$$

(2) For some leaf branch $B^0_{e^*}$ and some subtree $Q^0\subset B^0_{e^*}$ of T^0_r such that $V(P^0_{yr})\cap V(B^0_{e^*})\subset V(Q^0)$, we have

$$B^0_{a^*}\cup Q^0+a_\rho\varepsilon\,\tilde{\mathcal{D}}^{\pm},\quad \mathcal{D}^{\pm}-\{B^0_{a^*}\}=\tilde{\mathcal{D}}^{\pm}-\{B^0_{a^*}\cup Q^0+a_\rho\},\quad \mathcal{D}^{\mp}=\tilde{\mathcal{D}}^{\mp},$$

where P^0_{yr} is the path of T^0_r.

Proof: We only prove the case where $B^0_{a^*}\varepsilon\,\mathcal{D}^+$ and $A(T^0_r)=A^-(T^0_r)$. Let $(x^0(a):a\varepsilon A(T^0_r))$ and z^0_r, $(\tilde{x}^n(a):a\varepsilon A(\tilde{T}^n_r))$ and $\tilde{z}^n_r$ be the values defined in (4.9), (4.10), respectively. From $Q'=T^0_r\varepsilon\,\mathcal{N}$, it follows that

(4.14) $\quad \tilde{x}^n(a)=x^0(a)+\Delta_a n>0 \qquad (\ a\varepsilon A(\tilde{P}^n_{yr})\ )$,

(4.15) $\quad \tilde{z}^n_r=z^0_r-\Delta_a n<0$,

where $\tilde{P}^n_{yr}$ is the path of $\tilde{T}^n_r$. Note here that we have $0\leq x^0(a)\leq b(a)$ $(a\varepsilon A^-(T^0_r))$ and $z^0_r=0$. Define $X=\{a\varepsilon A(\tilde{P}^n_{yr}):\tilde{x}^n(a)>b(a)\}$. For $X=\phi$, we have (1) of this proposition. If $X\neq\phi$, then employ the same argument as in Proposition 4.10. □

Assume that $Q'\varepsilon\,\mathcal{N}$ equals some leaf branch $B^0_{a'}$ of (the new) T^0_r, then we have a member $Q''\varepsilon\,\mathcal{D}^+\cup\mathcal{D}^-$ such that Q' and Q'' are joined by the arc a'. Here, if Q'' belongs to $\mathcal{D}^{\mp}$ for the leaf branch $B^0_{a^*}\varepsilon\,\mathcal{D}^{\pm}$ of (the old) T^0_r, then we have the following proposition which can be shown in the same way as Proposition 4.11.

Proposition 4.12. If $Q'=B^0_{a'}\varepsilon\,\mathcal{N}$ and $Q''\varepsilon\,\mathcal{D}^{\mp}$ for $B^0_{a^*}\varepsilon\,\mathcal{D}^{\pm}$, then we have:

For some subtree $Q^3 \subset B_1'$ of Q' such that $V(\tilde{P}^n_{yr}) \cap V(B_1') \subset V(Q^3)$,

$$B^0_{a^*} \cup Q^3 + a_\rho \in \tilde{\mathcal{D}}^{\pm}, \quad \mathcal{D}^{\pm} - \{B^0_{a^*}\} = \tilde{\mathcal{D}}^{\pm} - \{B^0_{a^*} \cup Q^3 + a_\rho\}, \quad \mathcal{D}^{\mp} = \tilde{\mathcal{D}}^{\mp},$$

where $\tilde{P}^n_{yr}$ is the path of $\tilde{T}^n_r$. □

On the other hand, if $Q' = B^0_{a'} \in \mathcal{N}$ and Q'' joined by the arc a' is in $\mathcal{D}^{\pm}$ as to the leaf branch $B^0_{a^*} \in \mathcal{D}^{\pm}$, then examine the graph Q'' in detail. For $Q'' \in \mathcal{D}^{\pm}$, let $B_i'' \subset Q''$ $(1 \le i \le k'')$ be the members of $\mathcal{D}^{\pm}$ such that B''_{i+1} comes after B_i'' toward root r. In the same way as a_i' is defined, denote by a_i'' the arc joining B_i'' and B''_{i+1}. Then we have:

Proposition 4.13. If $Q' \in \mathcal{N}$ equals the leaf branch $B^0_{a'}$ and $Q'' \in \mathcal{D}^{\pm}$ for the leaf branch $B^0_{a^*} \in \mathcal{D}^{\pm}$, then we have either (1) or (2).

(1) $H_3 \in \tilde{\mathcal{D}}^{\pm}, \quad \mathcal{D}^{\pm} - \{B^0_{a^*}, Q''\} = \tilde{\mathcal{D}}^{\pm} - \{H_3\}, \quad \mathcal{D}^{\mp} = \tilde{\mathcal{D}}^{\mp},$

(2) For some leaf branch $B^4_{e'}$ of $B^0_{a'}$,

$$H_4 \in \tilde{\mathcal{D}}^{\pm}, \quad \mathcal{D}^{\pm} - \{B^0_{a^*}, Q''\} = \tilde{\mathcal{D}}^{\pm} - \{H_4\}, \quad \mathcal{D}^{\mp} = \tilde{\mathcal{D}}^{\mp},$$

where $H_3 = (B^0_{a^*} + a_\rho) \cup (B^0_{a'} + a') \cup Q''$ and $H_4 = B^0_{a^*} \cup B^4_{e'} + a_\rho$.

Proof: We only prove the case where $B^0_{a^*} \in \mathcal{D}^+$ and $A(T^0_r) = A^-(T^0_r)$. Let z^0_r, $(x^0(a) : a \in A(T^0_r))$ (resp. $\tilde{z}^n_r$, $(\tilde{x}^n(a) : a \in A(\tilde{T}^n_r))$) be the values defined in (4.9) (resp. (4.10)). From $B^0_{a'} \in \mathcal{N}$ and $B_1'' \subset Q'' \in \mathcal{D}^+$, we have

(4.16) $\quad \tilde{z}^n_r < 0 \qquad (B_1'' = T^0_r - V(Q')),$

(4.17) $\quad \tilde{x}^n(a_1'') > b(a_1'')$ (otherwise).

Note that we have the arc a_1'' if B_1'' is not equal to $T^0_r - V(Q')$. For the path $\tilde{P}^n_{yr}$ of $\tilde{T}^n_r$, let $X = \{a \in A(B^0_{a'} \cup B_1'' + a') \cap A(\tilde{P}^n_{yr}) : \tilde{x}^n(a) > b(a)\}$. We have (1) of this proposition for $X = \phi$, while we apply the same argument as in Proposition 4.10 for nonempty set X. □

Finally suppose that Q' is in $\mathcal{D}^{\mp}$ for the leaf branch $B^0_{a^*} \in \mathcal{D}^{\pm}$ of (the old) T^0_r where it will be only shown that the deficiency $d(\mathcal{D})$ of the network N decreases in this case. For this purpose, define S_1, S_2 and Δ_i' $(1 \le i \le m)$ with respect to $B^0_{a^*} \in \mathcal{D}^{\pm}$ by

(4.18) $\quad S_1 = \{a \in A^{\mp}(\tilde{T}^n_r) \cap X'' : \tilde{x}^n(a) > b(a)\} \cup \{a \in A^{\pm}(\tilde{T}^n_r) \cap X'' : \tilde{x}^n(a) < 0\},$

(4.19) $\quad S_2 = \{a \in A^{\mp}(\tilde{T}^n_r) \cap X'' : \tilde{x}^n(a) < 0\} \cup \{a \in A^{\pm}(\tilde{T}^n_r) \cap X'' : \tilde{x}^n(a) > b(a)\},$

(4.20)
$$\Delta_i' = \begin{cases} -x^0(a_i') & (a_i' \in A^{\mp}(\tilde{T}^n_r)), \\ x^0(a_i') - b(a_i') & (a_i' \in A^{\pm}(\tilde{T}^n_r)), \end{cases}$$

where $X'' = A(Q') \cap A(\tilde{P}^n_{yr})$ for the path $\tilde{P}^n_{yr}$ of $\tilde{T}^n_r$ and the values $x^0(a_i')$ on the arc a_i' connecting B_i' to B'_{i+1}, $\tilde{x}^n(a)$ appearing above are defined in (4.9), (4.10), respectively. Assume that $e'' \in S_1 \cup S_2$ is the arc nearest to vertex y and that

(4.21) $(\tilde{x}^{n+1}(a):a\varepsilon A(\tilde{T}_r^{n+1}))$, $\tilde{z}_r^{n+1}$ are the values given after doing ADJUST$(\tilde{T}_r^n, e'', (\tilde{x}^n(a):a\varepsilon A(\tilde{T}_r^n)), \tilde{z}_r^n)$,

where $\tilde{T}_r^{n+1}=\tilde{T}_r^n-V(\tilde{B}_{e''}^n)$ for leaf branch $\tilde{B}_{e''}^n$ of $\tilde{T}_r^n$ and $(\tilde{x}^n(a):a\varepsilon A(\tilde{T}_r^n))$, $\tilde{z}_r^n$ are the values defined in (4.10). Then we have:

Proposition 4.14. If $B_{a*}^0 \varepsilon \mathcal{D}^{\pm}$ and $Q' \varepsilon \mathcal{D}^{\mp}$, then we have either (1) or (2).

(1) (1-1) If there is some j such that $e''=a_j' \varepsilon S_1$, then we have

$$\tilde{x}^{n+1}(a)>0 \qquad (a\varepsilon A^{\mp}(\tilde{T}_r^{n+1}) \cap X_1),$$
$$\tilde{x}^{n+1}(a)<b(a) \qquad (a\varepsilon A^{\pm}(\tilde{T}_r^{n+1}) \cap X_1),$$

(1-2) If there is some j such that $e''=a_j' \varepsilon S_2$, then we have

$$d(\mathcal{D})>d(\tilde{\mathcal{D}}) \text{ for } \mathcal{D}=\mathcal{D}^+ \cup \mathcal{D}^- \text{ and } \tilde{\mathcal{D}}=\tilde{\mathcal{D}}^+ \cup \tilde{\mathcal{D}}^-,$$

(2) If there is some j such that $e'' \varepsilon A(B_j')$, then we have

$$\tilde{x}^{n+1}(a)>0 \qquad (a\varepsilon A^{\mp}(\tilde{T}_r^{n+1}) \cap X_2),$$
$$\tilde{x}^{n+1}(a)<b(a) \qquad (a\varepsilon A^{\pm}(\tilde{T}_r^{n+1}) \cap X_2),$$

where $X_1=A(B_{j+1}') \cap A(\tilde{P}_{yr}^n)$ and $X_2=A(B_j'-V(\tilde{B}_{e''}^n)) \cap A(\tilde{P}_{yr}^n)$ for the path $\tilde{P}_{yr}^n$ of $\tilde{T}_r^n$ and the leaf branch $\tilde{B}_{e''}^n$ of $\tilde{T}_r^n$.

Proof: We only prove the case (1) where $A(T_r^0)=A^-(T_r^0)$ and $B_{a*}^0 \varepsilon \mathcal{D}^+$. Let $(x^0(a):a\varepsilon A(T_r^0))$ be the values defined in (4.9).

(1-1): If $e''=a_j' \varepsilon S_1$, then we have $\tilde{B}_{e''}^n \varepsilon \underline{\tilde{\mathcal{D}}}^+$. Let $\Delta_{e''}=b(a_j')-x^0(a_j')$, then from Proposition 4.6 and (4.20), it follows that

(4.22) $\tilde{x}^{n+1}(a)=x^0(a)+\Delta_{e''}>x^0(a)+\Delta_j' \geq 0 \qquad (a\varepsilon A(B_{j+1}') \cap A(\tilde{P}_{yr}^n)).$

(1-2): Since $e''=a_j' \varepsilon S_2$, we have $G' \varepsilon \underline{\tilde{\mathcal{D}}}^-$ and $d(G')=\Delta_j'-\Delta_{a^n}$ for some subgraph G' of $\tilde{B}_{e''}^n$. Hence the deficiency change is given as follows.

(4.23) $d(\mathcal{D})-d(\tilde{\mathcal{D}})=\Delta_k+\Delta_j'-((\Delta_k-\Delta_{a^n})+(\Delta_j'-\Delta_{a^n}))>0.$ □

Proposition 4.14 means what follows. Case (1) (i.e., the arc e'' equals some arc a_j' joining B_j' and B_{j+1}') says that only if the leaf branch $\tilde{B}_{e''}^n$ is in $\underline{\tilde{\mathcal{D}}}^{\mp}$ for the leaf branch $B_{a*}^0 \varepsilon \mathcal{D}^{\pm}$ of (the old) T_r^0, then the deficiency reduces. Conversely, for Case (2) (i.e., the arc e'' is on some B_j'), we always have $\tilde{B}_{e''}^n \varepsilon \underline{\tilde{\mathcal{D}}}^{\pm}$ for $B_{a*}^0 \varepsilon \mathcal{D}^{\pm}$. The following proposition shows that the deficiency decreases for two special cases.

Proposition 4.15. If the leaf branch B_{a*}^0 is in $\mathcal{D}^{\pm}$ and $Q' \varepsilon \mathcal{D}^{\mp}$, then the deficiency decreases in the following two cases (1) and (2).

Case (1): Q' equals (the new) T_r^0.

Case (2): Q' equals some leaf branch $B_{a'}^0$ of (the new) T_r^0 and that we have at least one member of $\underline{\tilde{\mathcal{D}}}^{\mp}$ from Q'.

Proof: We only prove Case (1) where $B_{a*}^0 \varepsilon \mathcal{D}^+$, $A(T_r^0)=A^-(T_r^0)$ and $S_1 \cup S_2$ (defined in (4.18), (4.19)) is empty. Let z_r^0, $\tilde{z}_r^n$ be the values defined in

(4.9), (4.10), respectively. If $z_r^0>\Delta_a n$ for $\Delta_a n$ defined in (4.8), then we have $\tilde{T}_r^4\varepsilon\,\tilde{\mathcal{D}}^-$, $d(\tilde{B}_a^0 n)=\Delta_k-\Delta_a n$ and $d(\tilde{T}_r^4)=z_r^0-\Delta_a n$ where $\tilde{B}_a^0 n$ is the leaf branch of $\tilde{T}_r^0$ and $\tilde{T}_r^4\supset Q'$ is the rooted subtree of $\tilde{T}_r^n$. Therefore, the deficiency change is computed as

(4.24) $\quad d(\mathcal{D})-d(\tilde{\mathcal{D}})=\Delta_k+z_r^0-((\Delta_k-\Delta_a n)+(z_r^0-\Delta_a n))>0.$

Note here that we have $\Delta_a n>0$. If $z_r^0=\Delta_a n$, then put $z_r^0=\Delta_a n$ in (4.24). On the other hand, if $z_r^0<\Delta_a n$, then from $\tilde{z}_r^n=z_r^0-\Delta_a n$ and $\tilde{T}_r^5\varepsilon\,\underline{\mathcal{D}}^+$ for some rooted subtree $\tilde{T}_r^5$ of $\tilde{T}_r^n$, it follows that

(4.25) $\quad d(\mathcal{D})-d(\tilde{\mathcal{D}})=\Delta_k+z_r^0-((\Delta_k-\Delta_a n)+(\Delta_a n-z_r^0))>0.$

Note here that we have $z_r^0>0$. □

Now, the only case remained to be checked is the one where for the leaf branch $B_{a*}^0\varepsilon\mathcal{D}^\pm$ of (the old) T_r^0, $Q'\varepsilon\mathcal{D}^\mp$ equals the leaf branch $B_{a'}^0$ of (the new) T_r^0 and that we have no member $M\subset Q'$ of $\underline{\tilde{\mathcal{D}}}^\mp$. In this case we can similarly prove $d(\mathcal{D})>d(\tilde{\mathcal{D}})$. Hence, we have the following proposition.

Proposition 4.16. We have $d(\mathcal{D})>d(\tilde{\mathcal{D}})$ for $B_{a*}^0\varepsilon\mathcal{D}^\pm$ and $Q'\varepsilon\mathcal{D}^\mp$. □

We have proved the previous propositions 4.8~4.16 by assuming that the set $U^\pm$ defined in (4.4) is not empty, but it is also easy to see that we have the propositions 4.10~4.16 for empty set $U^\pm$.

Let $p=|\mathcal{D}|$, $q=\Sigma\{|V(Q)|:Q\varepsilon\mathcal{D}\}$, $\tilde{p}=|\tilde{\mathcal{D}}|$ and $\tilde{q}=\Sigma\{|V(\tilde{Q})|:\tilde{Q}\varepsilon\tilde{\mathcal{D}}\}$. (p is the number of members of $\mathcal{D}$, while q is the number of vertices contained in those of $\mathcal{D}$.) From Propositions 4.10~4.13, it follows that if the leaf branch $B_{a*}^0\varepsilon\mathcal{D}^\pm$ and $Q'\varepsilon\mathcal{D}^\pm\cup\mathcal{N}$, then the deficiency $d(\mathcal{D})$ does not change but that p and q are monotone functions of the number of pivot operations where the value $p+|V|-q$ is reduced at least one by a pivot operation. Hence we have:

Proposition 4.17. For $B_{a*}^0\varepsilon\mathcal{D}^\pm$ and $Q'\varepsilon\mathcal{D}^\pm\cup\mathcal{N}$, we have $d(\mathcal{D})=d(\tilde{\mathcal{D}})$, $p\geq\tilde{p}$, $|V|-q\geq|V|-\tilde{q}$, where $(p,\ |V|-q)\neq(\tilde{p},\ |V|-\tilde{q})$. □

We are now ready to estimate the computational complexity of the algorithm.

Theorem 4.18. The algorithm requires at most $(2|V|-4)(|V|-1)d(\mathcal{D})$ pivot operations and its running time is $O(|V|^2|A|d(\mathcal{D}))$.

Proof: The deficiency function d is nonnegative by definition. Since $b(a)$ and $x(a)$ ($a\varepsilon A$) are integral, $d(\mathcal{D})$ is also integral. We have seen that d is non-increasing and, if $d(\mathcal{D})=d(\tilde{\mathcal{D}})$, then $p+|V|-q$ is reduced at least 1 from Proposition 4.17. Consider the number of pivot operations except Trimming operations, to decrease the value $d(\mathcal{D})$. Now let $p^{(k)}$ (resp. $q^{(k)}$) be the number of the members of $\mathcal{D}$ (resp. that of vertices contained in those of $\mathcal{D}$) after k pivot operations. Assume that we initially have $p^{(0)}=p$ and $q^{(0)}=q$. Then from $p\leq q\leq |V|$, we have $(p-1)+(|V|-q)\leq |V|-1$. This means that at most $|V|-1$ pivot operations are required except Trimming operations. From Proposition 4.3 we have at most $(2|V|-4)(|V|-1)$ pivot operations to reduce $d(\mathcal{D})$. If $d(\mathcal{D})=0$, then the algorithm is over. So, the total number of pivot operations is $(2|V|-4)(|V|-1)d(\mathcal{D})$. As to the running time, we require $O(|A|)$ time in Steps 0 and 3, $O(|V|)$ time in Steps 1, 2 and 4 and $O(|A||V|)$ time in Step 5. Hence the total running time is $O(|V|^2|A|d(\mathcal{D}))$. □

5. Application of Scaling Method

We express upper capacities $b(a)$ $(a\in A)$ in binary form. Then we consider a sequence of problems each of which has approximated capacities. Scaling method is the new way to get an optimal solution for the original problem (P) by solving a sequence of these approximated problems repeatedly. Let

(5.1) $\quad t=\max\{i+1:2^i\leq \max\{b(a):a\in A\}<2^{i+1}\}$.

Define, for each $a\in A$, $b^{(k)}(a)=\lceil b(a)/2^{t-k}\rceil$ $(0\leq k\leq t)$, where for a number η, $\lceil\eta\rceil$ is the minimum integer greater than or equal to η. Now, we consider the following minimum cost flow problem P(k):

$$\begin{aligned} \mathrm{P}(k):\quad & \min\Sigma\{c(a)x(a):a\in A\}, \\ & \Sigma\{x(a):a\in\delta^+v\}-\Sigma\{x(a):a\in\delta^-v\}=0 \qquad (\ v\in V\ ), \\ & 0\leq x(a)\leq b^{(k)}(a) \qquad (\ a\in A\ ). \end{aligned}$$

We also define the deficiency functions $\tilde{d}_k$ and d_k with respect to Problem P(k) similarly as $\tilde{d}$ and d. Let $\bar{x}^{(k)}$ be an optimal solution of the problem P(k). Our aim here is to estimate the deficiency $d_k(\mathcal{D})$. Since $b^{(0)}(a)=1$ for each $a\in A$, $d_0(\mathcal{D})$ is bounded as follows.

Proposition 5.1. For Problem P(0), we have $d_0(\mathcal{D})\leq 2(|V|-1)|A|$.

Proof: From $b^{(0)}(a)=1$ $(a\in A)$ and Tree partitioning algorithm of T_r, we have $d_0(Q)\leq |A|$ for each $Q\in\mathcal{D}$ such that Q is either a leaf branch or $\mathcal{N}$-surrounded graph of T_r. Assume that Q is in $\mathcal{D}^+$ and $Q'\in\mathcal{D}^-$ is the nearest to Q. Denote $X(Q')$ by

(5.2) $\quad X(Q')=\{\tilde{Q}\in\mathcal{D}^+:\tilde{Q}$ is contained in some component of $T_r-V(Q')$ without root $r\}$.

Then we have $d_0(Q')\leq\Sigma\{d_0(\tilde{Q}):\tilde{Q}\in X(Q')\}+|A|$. Note that the members of $\mathcal{D}^+$ and those of $\mathcal{D}^-$ appear from root r to each leaf of T_r, alternatively, if we ignore members of $\mathcal{N}$. Let x be the circulation in Section 2.1 obtained from T_r for the problem P(0). Then $d_0(\mathcal{D})$ is not more than the sum of two times the difference between the value $x(a'')$ and its upper or lower bound, where a'' is the tree arc joining directly or indirectly between the member of $\mathcal{D}^+$

and that of $\mathcal{D}^-$. (i.e., a'' may be joining a member of $\mathcal{D}$ and that of $\mathcal{N}$.) The number of such a'' is at most $|V|-1$. □

Next, we make some preparations for finding an upper bound of $d_k(\mathcal{D})$ ($k\geq 1$). We have the following relation:

(5.3) $\quad 2b^{(k-1)}(a)-1\leq b^{(k)}(a)\leq 2b^{(k-1)}(a) \quad (a\in A).$

Let $T_r^{(k-1)}$ be a rooted spanning tree from which an optimal solution for Problem P(k-1) is obtained. Such a tree is called an optimal tree. Let $p^{(k-1)}$ be the potential determined by $T_r^{(k-1)}$. Then for Problem P(k), the potential $p^{(k-1)}$ satisfies the condition (CS)*. Now, we also define Problem $\tilde{\mathrm{P}}(k)$ similar to Problem P(k), i.e., $b^{(k)}$ given in the problem P(k) is replaced by $2b^{(k)}$ for the problem $\tilde{\mathrm{P}}(k)$. Once we find an optimal solution $\bar{x}^{(k-1)}$, an optimal one $\tilde{x}^{(k-1)}$ for the problem $\tilde{\mathrm{P}}(k-1)$ is easily obtained because we may choose $T_r^{(k-1)}$ as an optimal tree. For the problem P(k), we pick up $T_r^{(k-1)}$ as an initial tree. Let $x^{(k)}$ be the circulation determined from $T_r^{(k-1)}$ in Section 2.1. For an arc a ($a\in A$) and the leaf branch $B_a^{(k-1)}$ of $T_r^{(k-1)}$, define A_i ($1\leq i\leq 4$) by

(5.4) $\quad A_1=\{e\in A-A(T_r^{(k-1)}):\tau(p^{(k-1)},e)\geq 0,\partial^+e\in V(B_a^{(k-1)}),\partial^-e\in V-V(B_a^{(k-1)})\},$

(5.5) $\quad A_2=\{e\in A-A(T_r^{(k-1)}):\tau(p^{(k-1)},e)<0,\partial^+e\in V(B_a^{(k-1)}),\partial^-e\in V-V(B_a^{(k-1)})\},$

(5.6) $\quad A_3=\{e\in A-A(T_r^{(k-1)}):\tau(p^{(k-1)},e)\geq 0,\partial^-e\in V(B_a^{(k-1)}),\partial^+e\in V-V(B_a^{(k-1)})\},$

(5.7) $\quad A_4=\{e\in A-A(T_r^{(k-1)}):\tau(p^{(k-1)},e)<0,\partial^-e\in V(B_a^{(k-1)}),\partial^+e\in V-V(B_a^{(k-1)})\}.$

A_i ($1\leq i\leq 4$) is illustrated in Fig.11.

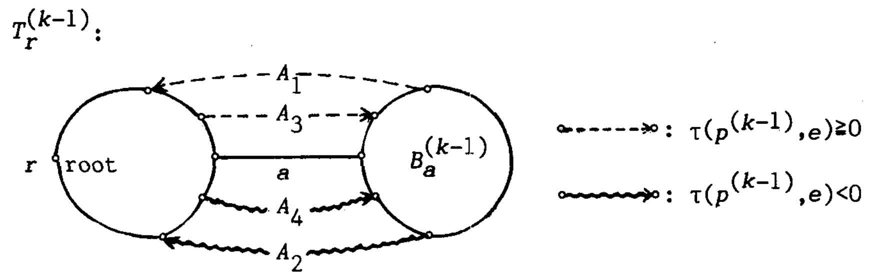

Fig.11

Then the following proposition estimates the difference between the value $x^{(k)}(a)$ and its upper or lower bound for $a\in A(T_r^{(k-1)})$.

Proposition 5.2. For each $a\in A(T_r^{(k-1)})$ ($k\geq 1$), we have

$\max\{-x^{(k)}(a),\ x^{(k)}(a)-b^{(k)}(a)\}\leq\max\{|A_1|+|A_2|,\ |A_3|+|A_4|\}+1.$

Proof: Assume that $A(T_r^{(k-1)})=A^+(T_r^{(k-1)})$. It suffices to show that for the arc $a\varepsilon A(T_r^{(k-1)})$ not satisfying the capacity condition, we have

(1) $x^{(k)}(a)-b^{(k)}(a)\leq|A_3|+|A_4|+1$,

(2) $-x^{(k)}(a)\leq|A_1|+|A_2|+1$.

We check (1) only. Let $A_i'=\{e\varepsilon A_i:\tilde{x}^{(k-1)}(e)=2b^{(k-1)}(e)\}$ $(i=1,3)$. Then from an optimal solution $\tilde{x}^{(k-1)}$, we have

(5.8) $$\Sigma\{2b^{(k-1)}(e):e\varepsilon A_2\}+\Sigma\{\tilde{x}^{(k-1)}(e):e\varepsilon A_1'\}$$
$$=\tilde{x}^{(k-1)}(a)+\Sigma\{2b^{(k-1)}(e):e\varepsilon A_4\}+\Sigma\{\tilde{x}^{(k-1)}(e):e\varepsilon A_3'\},$$

(5.9) $$0\leq\tilde{x}^{(k-1)}(a)\leq 2b^{(k-1)}(a).$$

As $x^{(k)}$ is a circulation in the problem $P(k)$, we have

(5.10) $$\Sigma\{b^{(k)}(e):e\varepsilon A_2\}+\Sigma\{b^{(k)}(e):e\varepsilon A_1'\}$$
$$=x^{(k)}(a)+\Sigma\{b^{(k)}(e):e\varepsilon A_4\}+\Sigma\{b^{(k)}(e):e\varepsilon A_3'\}.$$

From (5.3) and (5.8)~(5.10), it follows that

$$\begin{aligned} x^{(k)}(a)-b^{(k)}(a) &\leq \Sigma\{2b^{(k-1)}(e):e\varepsilon A_2\}-\Sigma\{(2b^{(k-1)}(e)-1):e\varepsilon A_4\} \\ &\quad +\Sigma\{2b^{(k-1)}(e):e\varepsilon A_1'\}-\Sigma\{(2b^{(k-1)}(e)-1):e\varepsilon A_3'\} \\ &\quad -(2b^{(k-1)}(a)-1) \\ &\leq \tilde{x}^{(k-1)}(a)+|A_4|+1-2b^{(k-1)}(a)+|A_3| \\ &\leq |A_3|+|A_4|+1. \square \end{aligned}$$

The following proposition, which gives an upper bound of $d_k(\mathcal{D})$, can be shown by using Propositions 5.1 and 5.2.

Proposition 5.3. For Problem $P(k)$ $(k\geq 1)$, the deficiency is bounded as $d_k(\mathcal{D})\leq 2(|V|-1)|A|$. □

From Theorem 4.18, Propositions 5.1 and 5.3, a polynomial-time algorithm for Problem (P) is given by solving a sequence of $(t+1)$ sub-problems $P(k)$ $(0\leq k\leq t)$. Hence we have the following theorem.

Theorem 5.4. We can obtain an optimal solution for the minimum cost flow problem (P) by the repetition of at most $4(|V|-2)(|V|-1)^2|A|(t+1)$ pivot operations. □

References

[1] Balinski, M. L. and Gomory, R. E.: A Primal Method for the Assignment and Transportation Problems. *Management Science*, Vol.10, No.3 (1964), 578-593.

[2] Cunningham, W. H.: A Network Simplex Method. *Mathematical Programming*, Vol.11 (1976), 105-116.

[3] Dantzig, G. B.: *Linear Programming and Extensions*. Princeton University Press. Princeton, N.J., 1963.

[4] Edmonds, J. and Karp, R. : Theoretical Improvements in Algorithmic Efficiency for Network Flow Problems. *Journal of the ACM*, Vol.19 (1972), 248-264.

[5] Ford, L. R. Jr. and Fulkerson, D. R.: A Primal Dual Algorithm for the Capacitated Hitchcock Problem. *Naval Research Logistics Quarterly*, Vol.4 (1957), 47-54.

[6] Ford, L. R. Jr. and Fulkerson, D. R.: *Flows in Networks*. Princeton University Press. Princeton, N.J., 1962.

[7] Fujishige, S.: A Capacity-rounding Algorithm for the Minimum-cost Circulation Problem ---- A Dual Framework of the Tardos Algorithm. *Mathematical Programming*, Vol.35, No.3 (1986), 298-308.

[8] Ikura, Y. and Nemhauser, G. L.: A Polynomial-time Dual Simplex Algorithm for the Transportation Problem. School of Operations Research and Industrial Engineering, Cornell University, 1984.

[9] Ikura, Y. and Nemhauser, G. L.: Computational Experience with a Polynomial-time Dual Simplex Algorithm for the Transportation Problem. *Discrete Applied Mathematics*, Vol.13 (1986), 239-248.

[10] Iri, M.: A New Method of Solving Transportation Network Problems. *Journal of the Operations Research Society of Japan*, Vol.3 (1960), 27-87.

[11] Klein, M.: A Primal Method for Minimal Cost Flows with Applications to the Assignment and Transportation Problems. *Management Science*, Vol.14 (1967), 205-220.

[12] Lawler, E. L.: *Combinatorial Optimization - Networks and Matroids*. Holt, Rinehart and Winston, New York, 1976.

[13] Orlin, J. B.: Genuinely Polynomial Simplex and Non-simplex Algorithms for the Minimum Cost Flow Problem. Working Paper No.1615-84, Alfred P. Sloan School of Management, MIT, December, 1984.

[14] Takahashi, I. and Fujishige, S.: *Discrete Mathematics* (in Japanese). Information Science, Vol.17, Iwanami Shoten, 1981.

[15] Tardos, E.: A Strongly Polynomial Minimum Cost Circulation Algorithm. *Combinatorica*, Vol.5, No.3 (1985), 247-255.

Akira NAKAYAMA: Department of Management Sciences, Faculty of Commerce,
Otaru University of Commerce,
Midori, Otaru, Hokkaido, 047, Japan.