Journal of the Operations Research
Society of Japan
Vol. 23, No. 2, June 1980

FLOW IN A NETWORK WITH A CHECK NODE

Tetsuo Ichimori,
Hiroaki Ishii
and
Toshio Nishida
Osaka University

(Received February 19, 1979; Final December 21, 1979)

Abstract This paper considers a new variant of maximum network flow problem, which has a constraint that each unit of network flow has to visit a specified node called a check node on the way from source to sink. For this problem an efficient algorithm is given which is an improved version of Hu's two-commodity flow algorithm. Moreover this paper discusses the applicability of this type of algorithm to the problem where the check node only has a positive gain.

Introduction

We often come across the problem in which each unit of a network flow has to go through a certain fixed node called a check node. For Example, in the transporation of milk from a dairy farm, milk is processed at a milk company, which corresponds to the check node, on the way from the dairy to consumers. Then the volume of milk may increase or decrease at the milk company which means that the check node has a positive gain denoted by γ. Our objective is to find a flow whose value is the maximum under the above constraint.

The problem is equivalent to the special case of the problem of Hu [1]. (See also [3].) However solving our problem efficiently needs some developments and improvements of Hu's two-commodity flow algorithm.

In the follwoing statement the case in which the gain of the check node equals one is discussed. Moreover it is shown that the case in which $\gamma\neq 1$ can be easily extended from the case $\gamma=1$.

Let $G=[N,A]$ be an n-node m-arc network with node set N and directed arc set A. We associate with each arc $(x,y)\in A$ a positive capacity $c(x,y)$. Let $s\in N$ be the source node, $t\in N$ be the sink node, and $u\in N$ be the check node. A function $f(x,y)$ defined on A is called a flow of value v in G if

(i) for every arc $(x,y)\in A$, $|f(x,y)|\le c(x,y)$

(ii) for every node $x\in N$,

$$\sum_y f(x,y)-\sum_y f(y,x)=\begin{cases} v & x=s\\ 0 & x\neq s,t\\ -v & x=t\end{cases}$$

where (and throughout this paper) each sum is taken over every y for which the summand is defined.

Problem Formulation

Let $f_b(x,y)$ and $f_a(x,y)$, respectively, denote the flows before and after visiting the check node u. Then the following must hold.

(iii) $|f_b(x,y)|+|f_a(x,y)|\le c(x,y)$

(iv) $$\sum_y f_b(x,y)-\sum_y f_b(y,x)=\begin{cases} v & x=s\\ 0 & x\neq s,u\\ -v & x=u\end{cases}$$

$$\sum_y f_a(x,y)-\sum_y f_a(y,x)=\begin{cases} v & x=u\\ 0 & x\neq u,t\\ -v & x=t.\end{cases}$$

We consider the following problem CN:

CN: maximize v

under the condition that f_b and f_a are flows in G, satisfying (iii) and (iv).

Let v^* denote the maximum value of v. Here we change variables as follows.

$$g_1(x,y)=f_b(x,y)+f_a(x,y)$$
$$g_2(x,y)=f_b(x,y)-f_a(x,y).$$

Then we have the following conditions (v) and (vi) in place of (iii) and (iv);

(v) $|g_1(x,y)|\le c(x,y)$

$|g_2(x,y)|\le c(x,y)$

(vi) $$\sum_y g_1(x,y)-\sum_y g_1(y,x)=\begin{cases} v & x=s\\ 0 & x\neq s,t\\ -v & x=t\end{cases}$$

$$\sum_y g_2(x,y)-\sum_y g_2(y,x)=\begin{cases} v & x=s\\ -2v & x=u\\ v & x=t\\ 0 & x\neq s,u,t.\end{cases}$$

From the max-biflow min-cut theorem of Hu [1] the relation between the values of f_b and f_a becomes as curve ABCD shown in Fig.1. If $\gamma\neq 1$, the value of f_b multiplied by γ should be equal to that of f_a, which condition is indicated by a line like line OE shown in Fig.1. In Fig.1 point A corresponds to the maximum value of flow f_b when $f_a(x,y)=0$ for every arc (x,y), point D to the maxixum value of flow f_a when $f_b(x,y)=0$ for every arc (x,y) and segment BC to the maximum sum of the values of two flows f_b and f_a. The coordinates of A, B, C, and D are $(v_{su1},0)$, (v_{su1},v_{tu1}), (v_{su2},v_{tu2}) and $(0,v_{tu2})$, respectively, and these v... are given by the next algorithm. Here we have to distinguish among the following three cases: (1) point E is on segment DC; (2) E is on segment CB; (3) E is on segment BA, which is determined by the value of γ and the relation between f_b and f_a. When $\gamma=1$, we note that $v^*=(v_{su1}v_{su2}-v_{su2}v_{tu1})/(v_{su1}+v_{tu2}-v_{su2}-v_{tu1})$ in case (2), and that $v^*=v_{tu2}$ in case (1), $v^*=v_{tu1}$ in case (3), which shows that v^* determined in the next algorithm is optimum.

Algorithm for CN

(Determination of v^*)

Step 1 Set $j\leftarrow 1$, $S\leftarrow s$, $T\leftarrow u$, and $f_i\leftarrow 0$. Go to Step 6.

Step 2 Set $v_{su1}\leftarrow v$, $S\leftarrow s,u$, $T\leftarrow t$, and $f_i\leftarrow f$. Go to Step 6.

Step 3 Set $v'_{tu1}\leftarrow v$, $S\leftarrow s,t$, $T\leftarrow u$. Go to Step 6.

Step 4 Set $v''_{tu1}\leftarrow v-v_{su1}$, $v_{tu1}\leftarrow \min(v'_{tu1},v''_{tu1})$, $S\leftarrow u$, $T\leftarrow t$, and $f_i\leftarrow 0$. Go to Step 6.

Step 5 Set $v_{tu2}\leftarrow v$, $v_{su2}\leftarrow v_{su1}+v_{tu1}-v_{tu2}$. Go to Step 7.

Step 6 Find the maximum flow f and its value v from S to T in G with initial flow f_i. Set $j\leftarrow j+1$. Go to Step j.

Step 7 Set $w\leftarrow (v_{su1}-v_{tu1})(v_{su2}-v_{tu2})$.
If $w<0$, set $v^*\leftarrow (v_{su1}v_{tu2}-v_{su2}v_{tu1})/(v_{su1}+v_{tu2}-v_{su2}-v_{tu1})$.
If $w\geq 0$, set $v^*\leftarrow \min(v_{su2},v_{tu2})$.

(Determination of f_b and f_a)

Step 8 Set $j\leftarrow 8$, $S\leftarrow s$, $T\leftarrow u$, $f_i\leftarrow 0$, and $v\leftarrow v^*$. Go to Step 12.

Step 9 Set $f_i\leftarrow f$, $S\leftarrow s,u$, $T\leftarrow t$, $v\leftarrow v^*$. Go to Step 12.

Step 10 Set $g_1\leftarrow f$, $S\leftarrow s,t$, $T\leftarrow u$, $v\leftarrow 2v^*$. Go to Step 12.

Step 11 Set $g_2\leftarrow f$. Go to Step 13.

Step 12 Find the flow f of value v from S to T in G under the constraint $\sum_y f(s,y)=v^*$, taking f_i as the initial flow. Set $j\leftarrow j+1$. Go to Step j.

Step 13 Set $f_b\leftarrow (g_1+g_2)/2$, and $f_a\leftarrow (g_1-g_2)/2$. Stop.

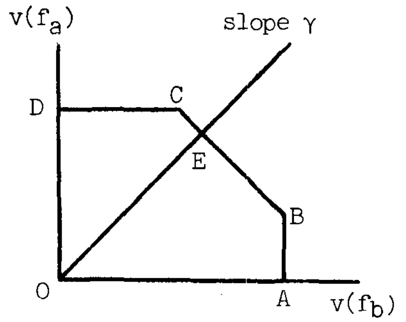

Fig. 1. v(f.) is the value of f..

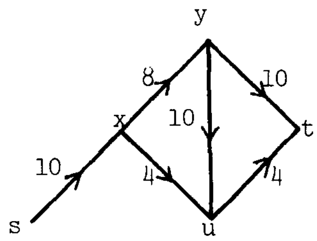

Fig. 2.

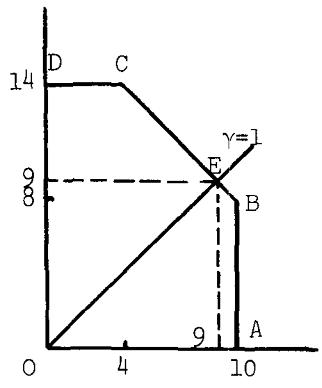

Fig. 3.

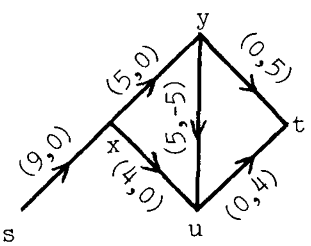

Fig. 4. (f_b, f_a).

Number of Operations

In Step 6 and Step 12 Karzanov's preflow method [2] can be used whish solves the maximum flow problem in $O(n^3)$ operations. The number of operations in the other steps is negligible as compared with that required in Step 6 and Step 12. Therefore the network flow problem with a check node can be solved in $O(n^3)$ operations.

Example

Consider the network of Fig.2. where the number on each arc is the capacity. We illustrate the algorithm by solving the example.

Step 1 — Step 6 :

J	S	T	f(s,x)	f(x,y)	f(x,u)	f(y,u)	f(y,t)	f(u,t)	v	
1	s	u	10	6	4	6	0	0	10	$v_{su1}=10$
2	s,u	t	10	6	4	-4	10	4	14	$v'_{tu1}=14$
3	s,t	u	10	6	4	10	-4	-4	18	$v''_{tu1}=8$
										$v_{tu1}=8$
4	u	t	0	0	0	-10	10	4	14	$v_{tu2}=14$
										$v_{su2}=4$

Step 7 : $w=(10-8)(4-14)<0$. $v^*=(10\times14-4\times8)/(10+14-4-8)=9$. See Fig.3.

Step 8 — Step 12 :

j	S	T	f(s,x)	f(x,y)	f(x,u)	f(y,u)	f(y,t)	f(u,t)	
8	s	u	9	5	4	5	0	0	
9	s,u	t	9	5	4	0	5	4	$\leftarrow g_1$
10	s,t	u	9	5	4	10	-5	-4	$\leftarrow g_2$
Step 13 :			9	5	4	5	0	0	$\leftarrow f_b$
			0	0	0	-5	5	4	$\leftarrow f_a$

Optimum flows (f_b, f_a) are shown in Fig.4.

Remark

The value v^* determined in Algorithm is only for the case $\gamma=1$. For the general case $\gamma\geq0$ it is easily obtained from Fig.1.

Acknowledgements

The authors wish to thank the referees for making valuable comments.

References

[1] Hu, T.C. : "Multi-Commodity Network Flows" *Operations Research* 11, 344-360 (1963).

[2] Karzanov, A.U. : "Determining the Maximum Flow in a Network by the Method of Preflows" *Soviet Mathematical Doklady* 15, 434-437 (1974).

[3] Megiddo, N. : "Optimal Flows in Networks with Multiple Sources and Sinks" *Mathematical Programming* 7, 97-107 (1974).

Tetsuo ICHIMORI: Department of
Applied Physics,
Faculty of Engineering,
Osaka University,
Suita, Osaka 565,
Japan