韓國經營科學會誌
第25卷 第2號
2000年 6月

최대유통문제의 단수치명호 결정 방법

정호연* · 안재근** · 박순달***

Determining the Single Most Vital Arc in the Maximum Flow Problem

Hoyeon Chung* · Jaegeun Ahn** · Soondal Park***

■ Abstract ■

The most vital arc in the maximum flow problem is that arc whose removal results in the greatest reduction in the value of the maximal flow between a source node and a sink node. This paper develops an algorithm to determine such a most vital arc in the maximum flow problem.

We first define the transformed network corresponding to a given network in order to compute the minimal capacity for each candidate arc. The set of candidate arcs for single most vital arc consists of the arcs whose flow is at least as great as the flow over every arc in a minimal cut. As a result, we present a method in which the most vital arc is determined more easily by computing the minimal capacity in the transformed network. The proposed method is demonstrated by numerical example and computational experiment.

1. 서 론

최대유통문제(Maximum Flow Problem)는 대표적인 네트워크문제 중의 하나로써 시점에서 종점까지 네트워크를 통해 보낼 수 있는 최대의 유통량과 경로를 구하는 문제이다[1, 3, 6, 8, 10].

이러한 최대유통문제에서 치명호(Most Vital Arc)란 최대유통량에 가장 치명적인 영향을 미치는 호(arc)를 가리킨다[2, 4, 9, 11]. 즉, 호와 마디(node)로 구성된 최대유통문제의 네트워크에서 우발적이거나 고의적인 여러 요인에 의해 호가 고장나거나 파괴될 수가 있는데, 이 때 어떤 호의

* 전주대학교 산업공학과
** 한경대학교 컴퓨터공학과
*** 서울대학교 산업공학과

고장이나 파괴가 시점과 종점간의 최대유통량을 가장 크게 감소시키는가를 알아보는 문제를 말한다. 이러한 연구는 적의 공격 하에 처해 있는 이해상충의 상황이나 물류(Logistics) 또는 통신네트워크에서 어느 호가 치명호인지 알아내어 적의 공격으로부터 경계를 강화하거나, 또는 어떤 호를 파괴시켜야 적의 시스템의 효율성을 가장 크게 저하시킬 수 있는지 알고자 하는 문제에 잘 적용될 수 있다. 이에 대한 연구는 Wollmer[12, 13]에 의해 가장 먼저 연구되었고, Durbin[7]이 Wollmer의 알고리듬을 적용하여 고속도로 시스템에서 치명호를 결정하였으며, Lubore[9] 등은 치명호의 필요조건을 제시하여 Wollmer 알고리듬의 단점을 개선하였다. Wollmer는 최대유통문제에서 최대유통량을 확보하기 위해 각 호에서 반드시 수송해야 되는 호의 값을 정의하여 이 호의 값이 가장 큰 호가 치명호가 됨을 보였다. 그러나 모든 호에 대하여 각 호에서 반드시 수송해야 되는 호의 값을 구해야 하기 때문에 효율성이 떨어지는 단점을 갖고 있다. 이러한 단점은 Lubore 등[9]에 의하여 개선되었다. Lubore 등은 최소절단(Minimal cut)에 속하는 호의 용량보다 적어도 더 큰 호만이 치명호가 될 수 있다는 필요조건을 제시하여 치명호를 결정하는 방법의 효율성을 개선시켰다.

치명호와 관련된 유사연구로는 Chang[5]의 최대유통문제에서 치명마디(Most Vital Node)를 결정하는 문제가 있는데, Chang은 치명마디를 결정하는 문제가 변환네트워크에서 치명호를 찾는 문제와 같음을 보이고, Lubore와 Ratliff 등의 방법을 변형시켜 1개의 치명마디(1-Most Vital Node)와 k개의 치명마디(k-Most Vital Node)를 결정할 수 있는 방법을 제시하였다.

본 연구에서는 최대유통문제에서 치명호를 구하는 이들의 방법을 변형시켜 좀 더 효율적으로 치명호를 구하는 방법을 제시하고자 한다.

2. 이론적 배경

본 연구에서 다루는 네트워크에는 시점(source) s와 종점(sink) t가 지정되고, 각 호 (i,j)에는 비음수 파라미터인 용량상한치(upper bound) u_{ij}와 용량하한치(lower bound) l_{ij}가 주어진다. u_{ij}는 호 (i,j)를 통과하는 유통량 f_{ij}의 최대허용치를 나타내며, l_{ij}는 f_{ij}의 최소허용치를 나타낸다(본 연구에서는 편의상 l_{ij}를 0으로 놓았음). 이 때 각 교점에서 흘러나가는 유통량(outflow)은 유통량보존법칙(flow conservation law)에 의해 그 교점으로 흘러 들어오는 유통량(inflow)과 같아야 한다(단, 시점 s와 종점 t에서는 예외). 시점에서 나간 모든 유통량은 종점에 도착해야 하기 때문에, 이 유통량의 총량을 v라고 하면 v는 다음과 같이 표현된다.

$$\sum_k f_{sk} = v = \sum_j f_{jt}$$

이러한 용량제약이 있는 네트워크에서 최대유통문제는 시점 s에서 종점 t까지 네트워크를 통해 보낼 수 있는 최대의 유통량을 구하는 문제이기 때문에, 최대유통문제 $G=(N,A)$는 다음과 같이 모형화 된다[1, 3, 8, 10].

$$\begin{aligned} &Max \quad v \\ &s.t. \quad \sum_{j:(i,j)\in A} f_{ij} - \sum_{j:(j,i)\in A} f_{ji} = \begin{cases} v, & i=\text{시 점 } s \\ 0, & i=\text{중간점} \\ -v, & i=\text{종 점 } t \end{cases} \\ &\qquad 0 \le f_{ij} \le u_{ij}, \quad \forall (i,j)\in A \end{aligned}$$

여기서 $G=(N,A)$는 마디 수가 $|N|=n$이고, 호의 수가 $|A|=m$인 유방향 네트워크라고 가정한다. 위의 네트워크 $G=(N,A)$에서, $G=(N,A)$의 교점집합 N을 시점 s와 종점 t를 중

심으로 상호배반(mutually exclusive)인 부분집합 X와 $\overline{X}$로 분리했을 때 이 두 부분을 연결하는 호의 집합을 절단집합 $(X, \overline{X})$라고 한다. 즉,

$$(X, \overline{X}) = \{ (x, y) | (x, y) \in A, x \in X, y \in \overline{X} \},$$
단, $s \in X$, $t \in \overline{X}$, $X \cap \overline{X} = \emptyset$, $X \cup \overline{X} = N$

여기서 절단용량(cut capacity) $C(X, \overline{X})$는 X에서 $\overline{X}$로 향하는 모든 호의 용량상한치의 합에서 $\overline{X}$에서 X로 향하는 모든 호의 용량하한치의 합을 뺀 것으로 정의된다[3,8,10]. 이 때 시점에서 종점까지의 최대유통량은 시점과 종점을 분리하는 $(X, \overline{X})$절단의 최소용량과 같다는 최대유통 최소절단정리(maximum flow minimum cut theorem)가 알려져 있다[1, 3, 8, 10].

위의 절단집합을 이용한 Lubore[9]의 정리는 다음과 같다.

[정리 1] 네트워크 $G = (N, A)$에 대한 최적해 $V(F)$가 주어졌을 때, 호 (a, b)가 치명호가 되기 위한 필요조건(necessary condition)은 (a, b)상의 유통량이 최소절단에 속한 모든 호들의 각각의 유통량보다 적어도 더 크거나 같아야 한다. □

위의 [정리 1]은 네트워크 $G = (N, A)$의 최소절단에 속한 호 중에서 가장 큰 유통량 보다 적은 유통량이 흐르는 호는 치명호의 후보로서 고려할 필요가 없다는 사실을 말해 준다.

지금 네트워크 $G = (N, A)$에 대한 최대유통량을 $V(F)$라 하고, 이 때의 각 호 $(i, j) \in A$에 대한 최적유통량을 f_{ij}라 하자.

일단 $G = (N, A)$에 대한 최적해가 주어진 상태에서 호 용량이 변하게 되면 주어진 최적해가 비가능(infeasible)일 수 있다. 따라서 이 때에는 주어진 각 호의 유통량을 재최적화시켜 주어야 하는데, 이 때의 원활한 계산을 위해 $G = (N, A)$에 대한 변환네트워크를 다음과 같이 정의해 보자

[정의 1] 변환네트워크 : $G_F = (N, A')$

$G = (N, A)$에 대한 최적해가 주어졌을 때 변환네트워크 $G_F = (N, A')$는 다음과 같이 정의된다.

$G = (N, A)$의 어떤 호 (x, y)에 대하여 만일 $f_{xy} > 0$, $(x, y) \in A$ 이면 $G_F = (N, A')$의 A'에는 새로운 유통용량 $u'_{xy} = u_{xy} - f_{xy}$ 를 갖는 호 (x, y)와 $u'_{yx} = f_{xy}$를 갖는 호 (y, x)를 추가하고, 만일 $f_{xy} = 0$, $(x, y) \in A$ 이면 $G = (N, A)$의 유통용량과 같게 즉, $u'_{xy} = u_{xy}$ 로 놓는다. □

최대유통문제는 네트워크 상에서 두 점 사이에 최대로 수송할 수 있는 양과 경로를 구하는 문제이기 때문에 $V(F)$를 산출하는 해의 형태(flow pattern)가 여러가지 일 수 있다[9,10]. 이처럼 $G = (N, A)$에 대한 대안 최적해가 다수개가 존재하더라도 각 호에는 $V(F)$를 얻기 위해서 반드시 수송해 주어야 하는 최소한의 유통량이 있는데[9], 이를 최소유통용량 (minimum capacity)이라고 정의하자. 이를 최소유통용량이라고 부르는 이유는 이 최소유통용량 이하의 유통량을 보낼 경우에는 최대유통량을 얻을 수 없기 때문이다.

[정의 2] 최소유통용량(minimum capacity) c_{ij}

$G = (N, A)$에 대한 최적해가 주어졌을 때 시점과 종점간의 $V(F)$를 얻기 위해 각 호에서 반드시 수송해 주어야 하는 최소한의 유통량을 호의 최소유통용량(minimum capacity)이라고 한다. □

최소유통용량은 따라서 항상 비음의 유통량 값을 갖는다. 여기에서는 $G = (N, A)$에 대한 최적해가 주어졌을 때, 호 (i, j)의 최소유통용량을 쉽게 구하기 위해서, 원 네트워크를 변환시킨 $G_F = (N, A')$를 사용하여 최소유통용량을 구할 수 있는 방법을 제시하고자 한다.

먼저 $G_F = (N, A')$에서 각 호 (i, j)의 시작마디 i와 도착마디 j를 시점과 종점으로 놓고, 최대유통문제를 풀었을 경우 구할 수 있는 최대유통량을

h_{ij}라고 정의하자. 이 때 $G_F=(N, A')$에서의 최대유통량 h_{ij}는 유통량보존법칙(flow conservation law)에 의해 주어진 대안해의 형태와 상관없이 항상 동일하게 나타난다.

[특성 1] $G_F=(N, A')$에서 마디 i에서 마디 j까지의 최대유통량을 h_{ij}라고 하자. 이 때 h_{ij}는 $G=(N, A)$ 의 모든 대안 최적해에 대해 동일한 값을 가진다. □

위의 [특성 1]은 대안 최적해가 존재할 때 임의의 한 최적해만 주어지더라도 그것으로부터 호(i, j)의 최소유통용량을 구할 수 있다는 사실을 말해준다.

위의 특성에 따라 $G_F=(N, A')$에서 구한 최대유통량 h_{ij}를 이용하여 최소유통용량 c_{ij}를 계산하면 다음과 같다.

[정리 2] $G=(N, A)$에 대한 최적해가 주어졌을 때 호 (i, j)의 최소유통용량 c_{ij}는 $u_{ij}>h_{ij}$ 이면 $c_{ij}=u_{ij}-h_{ij}$ 이고, $u_{ij} \le h_{ij}$ 이면 $c_{ij}=0$ 이다.

(증명) 먼저 $G_F=(N, A')$에서 마디 i에서 마디 j까지 갈 수 있는 모든 유통경로의 집합을 P_{ij}로 정의하자. 그러면 마디 i에서 마디 j까지의 최대유통량 h_{ij}는 호 (i, j)를 통해 직접 보내는 유통량 ($u_{ij}-f_{ij}$) 와 호 (i, j)를 우회하여 보내는 유통량의 합 ($\sum_{p\in P_{ij}\setminus(i,j)} f_{ij}^p$)으로 나타낼 수 있다. 즉, $h_{ij}=(u_{ij}-f_{ij})+\sum_{p\in P_{ij}\setminus(i,j)} f_{ij}^p$ 이다.

그런데 호 (i, j)의 최소유통용량 c_{ij}는 f_{ij}에서 마디 i에서 마디 j까지 우회해서 보내는 유통량을 뺀 값, 즉, $c_{ij}=f_{ij}-\sum_{p\in P_{ij}\setminus(i,j)} f_{ij}^p$ 이기 때문에 $f_{ij} > \sum_{p\in P_{ij}\setminus(i,j)} f_{ij}^p$ 인 경우 $c_{ij}=u_{ij}-h_{ij}$이 되고, $f_{ij} \le \sum_{p\in P_{ij}\setminus(i,j)} f_{ij}^p$ 인 경우 최소유통용량의 정의에 따라 c_{ij}는 0의 값을 가진다.

이를 간단히 나타내면 $u_{ij}-h_{ij}$가 양이면 $c_{ij}=u_{ij}-h_{ij}$이고, $u_{ij}-h_{ij}$가 비양이면 $c_{ij}=0$이 된다. □

3. 최대유통문제의 단수치명호 해법

네트워크 $G=(N, A)$에 대한 최적해가 주어진 상황에서 호 (i, j)가 제거되었다고 가정해 보자. 그러면 $V(F)$는 호의 최소유통용량 c_{ij} 만큼 감소되게 된다.

[정리 3] $G=(N, A)$에 대한 최적해가 주어진 상황에서 만일 호 (i, j)가 제거된다면 최대유통량은 $V(F)-c_{ij}$가 된다.

(증명)

$G=(N, A)$에 대한 최적해가 주어져 있으므로 각 호 (i, j)에는 f_{ij} 의 유통량이 흐르고 있다. 호 (i, j)를 통해서 보내는 이 f_{ij} 의 양은 대안 최적해가 존재할 경우 마디 i에서 마디 j까지의 우회경로를 통해 유통량을 우회시킬 수도 있으나, $V(F)$를 얻기 위해서는 적어도 최소유통용량 c_{ij} 만큼의 유통량은 반드시 호 (i, j)를 통해서 보내주어야 한다. 그런데 호 (i, j)가 제거된다면 호의 최소유통용량 c_{ij} 만큼의 유통량을 시점에서 종점까지 전달해 줄 수 없기 때문에 최대유통량은 c_{ij} 만큼 감소되게 된다. 따라서 호 (i, j)가 제거되게 되면 최대유통량은 $V(F)-c_{ij}$가 된다. □

위의 [정리 3]에 따라 호 (i, j)가 제거될 경우 최대유통량을 가장 크게 감소시키는 호, 즉, 치명호는 호의 최소유통용량이 가장 큰 호가 된다.

[정리 4] $G=(N, A)$에 대한 최적해가 주어진 상황에서, 만일 호 (i, j)가 제거된다면 호의 최소

유통용량이 최대인 호가 치명호가 된다.

(증명)

최대유통문제에서 치명호란 한 호의 제거로 최대유통량을 가장 크게 감소시키게 되는 호를 말한다. 이 때 호의 최소유통용량 c_{ij}는 시점과 종점간의 최대유통량을 얻기 위해서 각 호에서 반드시 수송해 주어야 하는 최소한의 유통량이라 정의하였으므로, [정리 2]에 의해 호의 최소유통용량이 가장 큰 호가 최대유통량을 가장 크게 줄이게 된다. 따라서 호의 최소유통용량이 최대인 호가 치명호가 된다. □

주어진 네트워크에서 치명호를 구하기 위해서는 먼저 $G=(N,A)$에서 최소절단을 구한 뒤, $G=(N,A)$에 대한 변환네트워크 $G_F=(N,A')$를 구성한다. 그 다음, 최소절단에 속하는 호의 용량보다 적어도 더 큰 호에 대하여 $G_F=(N,A')$에서 h_{ij}를 구한 뒤 c_{ij}를 계산하면 치명호를 구할 수 있게 된다. 자세한 해법은 다음과 같다.

계산법 (최대유통문제의 단수치명호 결정문제)

[단계 0] 최적해 산출

(0-1) $G=(N,A)$에 대한 최적해를 구한다. 이 때의 최소절단을 $(X,\overline{X})$라 하자.

(0-2) $u^*=MAX_{(i,j)\in(X,\overline{X})} f_{ij}$라 놓는다.

(0-3) 치명호의 후보집합 S를 다음과 같이 놓는다.
$S=\{(i,j)\in A|\ f_{ij}\geq u^*\}$

[단계 1] $G_F=(N,A')$에서의 유통량(h_{ij}) 및 호의 최소유통용량 c_{ij} 계산

(1-1) 주어진 최대유통문제에 대한 $G_F=(N,A')$를 구성한다.
이 때 $G_F=(N,A')$에서의 수정된 용량상한 $\overline{U}$는 다음과 같이 정의된다.
만일 $f_{xy}>0,\ (x,y)\in A$ 이면
$\overline{u}_{xy}=u_{xy}-f_{xy},\ \overline{u}_{yx}=f_{xy},$
만일 $f_{xy}=0,\ (x,y)\in A$ 이면 $\overline{u}_{xy}=u_{xy}$

(1-2) $G_F=(N,A')$에서의 후보집합 S에 속한 호들의 유통량(h_{ij}) 계산
최소절단에 해당하는 호 $(i,j)\in S$에 대하여 $c_{ij}=u_{ij}$ 로 놓는다.
최소절단에 속하지 않는 호 $(i,j)\in S$에 대하여 $c_{ij}=Max\{0,u_{ij}-h_{ij}\}$를 계산

[단계 2] 치명호 결정

최소유통용량 c_{ij}가 가장 큰 호가 치명호가 된다.
즉, MVA = $Max_{(i,j)\in S}\ c_{ij}$

4. 단수치명호 결정을 위한 꼬리표법의 효율화 방안

최대유통문제의 해법중의 하나인 꼬리표법(Labeling Method)을 사용하여 위의 최대유통문제의 단수치명호를 효율적으로 결정하기 위해서는 [단계 1]에서 모든 호에 대해 각 호 (i,j)의 최소유통용량 c_{ij}의 값을 계산하면 된다. 그런데 실제로 [단계 2]에서 필요한 호는 $Max_{(i,j)\in S}\ c_{ij}$이다. 그러므로 [단계 1]과 [단계 2]를 통합할 경우에 중간단계의 $Max_{(i,j)\in S}\ c_{ij}$ 를 기억하는 변수가 필요한데 이를 $Temp_c_{ij}$라고 하자. 치명호 결정과정에서, 지금까지 고려한 호들에 대하여 가장 큰 최소유통용량을 구하는 변수의 값인 $Temp_c_{ij}$의 값은 계산하는 호를 거듭할 수록 증가하는 성질이 있다. 그런데 한 호의 최소유통용량인 c_{ij}는 $u_{ij}-h_{ij}$로 표현되기 때문에 치명호를 구하는 중간단계에서 구한 $Temp_c_{ij}$보다 작은 호의 용량 u_{ij}를 갖는 호 (i,j)는 치명호가 될 수 없다. 그러므로 이 호에 대해서는 최대유통량을 구하는 과정을 생략

해도 무방하다. 이를 정리하면 다음과 같다.

[관찰 1] 치명호를 구하는 과정에서, 지금까지 고려한 호들 중 $Temp\_c_{ij}$ 보다 호의 용량상한 u_{ij}가 작은 호 (i, j)에 대해서는 호 (i, j)의 시작마디인 i에서 도착마디인 j로 향하는 변환네트워크에서의 최대유통량 h_{ij}를 구하는 과정을 생략해도 무방하다. □

또한, 한 호 (i, j)에 대한 c_{ij}를 구하는 것은 결과적으로 $u_{ij} - h_{ij}$를 구하는 것이다. 그러나 h_{ij}를 구하기 위하여 꼬리표법을 사용하면 시작마디 i에서 도착마디 j로 향하는 유통확장경로(FAP)를 매번 하나씩 구해서 임시로 구한 최대유통량 ($Temp_h_{ij}$라고 하자)에 추가하여야 한다. 그런데 $Temp_h_{ij}$는 각 호별로 FAP이 추가되는 순간마다 증가하는 성질을 가지고 있다. 그런데 우리가 구하고자 하는 중간단계에서 구한 c_{ij}값들 중의 최대값 $Temp_c_{ij}$은 $u_{ij} - h_{ij}$ 를 통해 구할 수 있다. 그런데 $Temp_h_{ij}$는 증가하는 성질이 있기 때문에 $u_{ij} - Temp_h_{ij}$의 값은 감소하게 된다. 그러므로 호 (i, j)가 치명호의 후보가 될 수 있는지 없는지는 h_{ij}를 모두 구하고서 알 수 있는 것은 아니다. 이를 정리하면 다음의 [관찰 2]와 같다.

[관찰 2] 치명호를 구하는 과정에서, 한 호 (i, j)의 시작마디 i에서 도착마디 j로 향하는 최대유통경로의 유통량의 합인 $Temp_h_{ij}$가 $u_{ij} - Temp\_c_{ij}$ 보다 크게 되면, 이 호 (i, j)는 치명호가 될 수 없으므로 더 이상 $Temp_h_{ij}$에 FAP을 추가하여 최대유통량을 구할 필요는 없다. □

[관찰 1]과 [관찰 2]를 통해, 한 호의 최대유통량 h_{ij}와 최소유통용량 c_{ij}대신에 다음의 두 가지 표현을 사용하여 계산법을 수정하고자 한다. 첫째, h_{ij}는 한 호 (i, j)의 시작마디 i에서 도착마디 j로 향하는 최대유통경로의 유통량의 중간 합을 의미하는 변수인 $Temp_h_{ij}$로 수정하고, c_{ij}는 치명호를 구하는 중간단계에서 지금까지 구해진 가장 큰 최소유통용량을 의미하는 변수인 $Temp_c_{ij}$로 수정한다. 이 두 변수를 이용하면 불필요한 계산을 줄일 수 있다.

이를 이용하여 단수치명호를 결정하는 방법을 효율화시킨 계산법은 다음과 같다.

계산법 (최대유통문제의 단수치명호 결정 문제의 효율적 구현)

[단계 0] 최적해 산출

(0-1) $G = (N, A)$에 대한 최적해를 구한다. 이 때의 최소절단을 $(X, \overline{X})$라 하자.

(0-2) $u^* = MAX_{(i,j) \in (X, \overline{X})} f_{ij}$라 놓는다.

(0-3) 치명호의 후보집합 S를 다음과 같이 놓는다.

$$S = \{(i, j) \in A | \ f_{ij} \geq u^*\}$$

[단계 1] 주어진 최대유통문제에 대한 $G_F = (N, A')$의 구성.

이 때 $G_F = (N, A')$에서의 수정된 용량상한 $\overline{U}$는 다음과 같이 정의된다.

만일 $f_{xy} > 0, (x, y) \in A$ 이면

$$\overline{u}_{xy} = u_{xy} - f_{xy}\ , \quad \overline{u}_{yx} = f_{xy}$$

만일 $f_{xy} = 0, (x, y) \in A$ 이면 $\overline{u}_{xy} = u_{xy}$

[단계 2] $G_F = (N, A')$에서의 유통량(h_{ij}) 및 호의 최소유통용량 c_{ij} 계산

최소절단에 해당하는 호 $(i, j) \in S$에 대하여 $Temp\_c_{ij} = u_{ij}$ 로 놓는다.

최소절단에 속하지 않는 호 $(i, j) \in S$에 대하여 c_{ij}를 다음과 같이 계산한다.

$$Temp\_c_{ij} = Max_{(i,j) \in S}\ Temp\_c_{ij}$$

(2-1) 선택되지 않은 호 (i, j)를 하나 선택한다.

선택할 호가 없으면 [단계 2]로 간다.

(2-2) 만일 ($Temp\_c_{ij} > u_{ij}$) 이면, 이 호는 치명호가 될 수 없다. (2-1)로 간다.

(2-3) 각 호 (i,j)의 시작마디 i와 도착마디 j를 시점과 종점으로 놓는다.

(2-4) $Temp\_h_{ij} = 0$

(2-5) 시점과 종점사이에 FAP을 하나 찾는다. 만일 FAP이 있을 경우, FAP의 최소용량을 $Temp_h_{ij}$에 더한다. (2-6)으로 간다. 아니면, 만일 ($Temp\_c_{ij} < u_{ij} - Temp\_h_{ij}$)이면, $Temp\_c_{ij} = u_{ij} - Temp\_h_{ij}$ 호 (i,j)를 치명호의 후보로 놓는다. (2-1)로 간다.

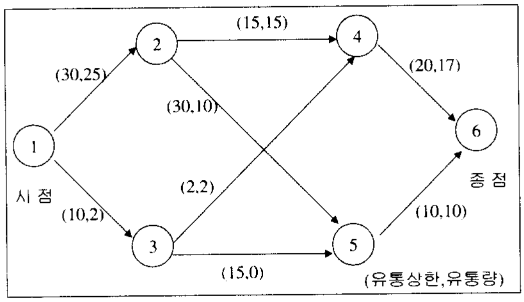
(2-6) 만일 ($u_{ij} < Temp\_h_{ij}$) 이면, 이 호는 치명호가 될 수 없다. (2-1)로 간다. 아니면, (2-5)로 간다.

위의 계산법에서 (2-2) 조건인 ($Temp\_c_{ij} > u_{ij}$) 이면, [관찰 1]에 해당한다. 그리고 최소유통용량 계산과정에서 치명호에 속하지 않음이 확실한 호의 FAP을 찾는 꼬리표 수행을 중간에 종료하는 방법은 효율적인 계산법의 (2-6)의 조건에 부합할 경우에 해당한다. 이는 [관찰 2]를 통해 얻을 수 있는 결과이다.

5. 예 제

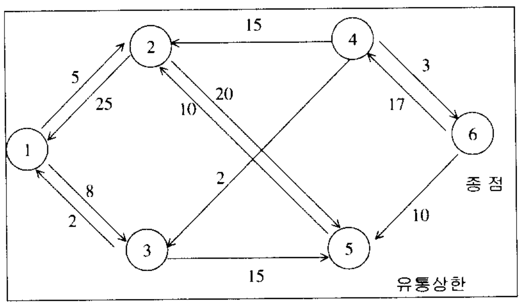
다음과 같이 마디수가 6이고 호의 수가 8인 최대유통문제를 고려해 보자. 이 문제의 최대유통량은 27이며, 이 때의 각 호에 대한 최적유통량은 괄호의 왼편에 나타나 있다[9].

[단계 0]에서 최소절단은 ({① ② ③ ⑤},{④ ⑥}) 이며, 최소절단에 해당하는 호는 {(2,4),(3,4), (5,6)} 이다. 단계 (0-2)에 의해 $U^* = 15$가 되고, 단계 (0-3)에 의해 치명호 후보집합 $S = \{(1,2),\ (2,4),\ (2,5),\ (3,5),\ (4,6)\}$ 이 된다.



[그림 1] 최대유통문제 $G=(N,A')$의 $V(F)=27$

그리고 [단계 1]에서 [단계 0]에서 구한 최적해를 통해 주어진 네트워크에 대한 변환네트워크 $G_F=(N,A')$를 구성하면 [그림 2]와 같다.



[그림 2] $G=(N,A)$에 대한 $G_F=(N,A')$

각 호 (i,j)의 최소유통용량을 구하기 위해 [그림 2]의 $G_F=(N,A')$ 상에서 마디 i에서 마디 j까지의 최대유통량을 구해 보면 다음과 같다.

〈표 1〉 변환네트워크에서 구한 각 마디간의 최대유통량

H =

From \ To	①	②	③	④	⑤	⑥
①	∞	**13**	8	0	13	0
②	-	∞	-	0	**28**	0
③	-	-	∞	0	**17**	0
④	-	-	-	∞	-	**3**
⑤	-	-	-	-	∞	0
⑥	-	-	-	-	-	∞

(단, 음영부분의 계산만이 필요)

그리고 [정리 2]에 의해 각 호 (i, j)의 최소유통 용량은 다음과 같다.

〈표 2〉 $G=(N, A)$에서의 각 호(i, j)에 대한 최소유통용량

C =

From \ To	①	②	③	④	⑤	⑥
①	∞	17	2	-	-	-
②	-	∞	-	15	2	-
③	-	-	∞	2	0	-
④	-	-	-	∞	-	17
⑤	-	-	-	-	∞	10
⑥	-	-	-	-	-	∞

(단, 음영부분의 계산만이 필요)

그러므로 치명호 후보집합중에서 호의 최소유통 용량이 가장 큰 호는 호 (1,2)와 호 (4,6)이기 때문에 [정리 4]에 의해 이들 호가 치명호가 된다.

6. 실험 및 분석

위의 해법에 대해 기존의 Wollmer 알고리듬의 단점을 개선한 Lubore의 알고리듬과의 비교실험을 위해 DIMACS에서 표준문제생성기인 Washington 문제의 문제형태 1, 2, 3, 6, 7, 8, 9, 10번문제(1. Mesh Graph, 2. Random Leveled Graph, 3. Random 2 Leveled Graph, 6. Basic Line Mesh, 7. Exponential Line, 8. Double Exponential Line, 9. Line Graph - Bad case for Dinics, 10. Bad case for Goldberg)와 Goldberg가 제시한 문제 생성기 Ak를 이용하여 40개의 문제에 대해 Lubore의 방법과 제안한 방법간의 비교실험을 수행하였다. 실험결과는 다음 <표 3>과 같다.

본 연구에서 제안한 방법은 실험문제에 대해 40배에서 2000배 정도의 수행횟수에서의 감소를 보였고, (전체 수행시간/전체 수행횟수)의 비는 Lubore의 방법에 비해 본 논문에서 제안한 방법이 2배 정도의 증가를 보였다. 이는 변환네트워크를 구성하기 때문에 시점에서 종점으로의 한번 꼬리표지하고 유통을 갱신하는 계산량이 변환네트워크를 구성하지 않는 경우보다 약 2배 정도를 더 소모하는 것으로 보인다. 그러나 Lubore 등의 방법이 해를 구할 수 있었던 문제들만을 고려하여 비교해 보면 평균적으로 Lubore등의 방법보다 수행시간에 있어서는 660배의 감소, 수행횟수에 있어서는 420배의 수행횟수의 감소를 보였다. Line Graph인 washington문제의 9번 문제중의 소형문제를 제외하고는 모든 문제에 있어서 본 논문에서 제안한 방법이 수행도의 면에서 탁월한 것으로 보인다. 이는 꼬리표법을 사용한 방법이 기존의 Lubore 등의 방법에서는 최적해를 이용하지 않기 때문에 한 호씩을 제거한 네트워크에서 중복적으로 꼬리표지를 하는 수행횟수가 많음에 기인한 것으로 보인다.

7. 결 론

본 연구는 최대유통문제의 최적해가 주어진 상태에서, 이를 이용하여 최대유통문제의 단수치명호를 선정할 수 있는 방법을 제시한 것이다.

이를 위해 먼저 주어진 네트워크에 대한 변환네트워크를 정의한 후, 이 변환네트워크에서 각 호의 시작마디로부터 종착마디까지의 최대유통량을 구하고, 이 값을 이용하여 시점과 종점간의 최대유통량을 얻기 위해 각 호에서 반드시 수송해야 하는 최소유통용량을 계산하였다. 이 때 단수치명호는 최소유통용량이 가장 큰 호가 됨을 보였다. 본 연구에서는 이에 대한 계산법과 꼬리표법의 수행을 효율화할 수 있는 방안을 적용하여 구한 효율적인 계산법도 함께 제시하였다. 실험결과 꼬리표법을 사용할 경우 기존의 방법에 비해 실용적으로 매우 효율적인 방법임을 확인할 수 있었다.

추후연구로는 최대유통문제의 해법으로 꼬리표법만을 비교실험의 대상으로 사용하였으나, 다른 해법에 대한 비교실험을 통해 모든 최대유통문제의 해법에 대한 효율성을 비교하는 것이 추후 필요할 것으로 생각된다.

〈표 3〉 최대유통문제의 단수치명호 수행결과

문제 (seed값 포함)	마디수	호수	최대 유통량	한 호의 제거로 가장 많이 줄어드는 량	수행횟수		수행시간		전체수행시간/전체 수행횟수	
					Lubore 등	제시한 해법	Lubore 등	제시한 해법	Lubore 등	제시한 방법
ak10	46	67	23	12	1141	**27**	0.03	**0.00**	0.00003	**0.0**
ak20	86	127	43	22	4176	**47**	0.19	**0.01**	0.00005	**0.00021**
ak30	126	187	63	32	9111	**67**	0.56	**0.02**	0.00006	**0.00029**
ak40	166	247	83	42	15946	**87**	1.32	**0.00**	0.00008	**0.0**
ak50	206	307	103	52	24681	**107**	2.53	**0.02**	0.00010	**0.00019**
ak100	406	607	203	102	96856	**207**	20.92	**0.07**	0.00022	**0.00033**
ak200	806	1207	403	202	383706	**407**	174.88	**0.28**	0.00046	**0.00069**
ak300	1206	1807	603	302	860556	**607**	613.74	**0.61**	0.00071	**0.00100**
ak400	1606	2407	803	402	1527406	**807**	1462.02	**1.09**	0.00096	**0.00135**
ak500	2006	3007	1003	502	-	**1007**	-	**1.88**		
ak1000	4006	6007	2003	1002	-	**2007**	-	**7.19**		
ak2000	8006	12007	4003	2002	-	**4007**	-	**29.19**		
ak3000	12006	18007	6003	3002	-	**6007**	-	**71.48**		
ak4000	16006	24007	8003	4002	-	**8007**	-	**126.36**		
ak5000	20006	30007	10003	5002	-	**10007**	-	**206.12**		
ak10000	40006	60007	20003	10002	-	**20007**	-	**872.48**		
1_10_20_10	202	590	100	10	51120	**518**	9.57	**0.07**	0.00019	**0.00014**
1_20_50_50	1002	2980	918	50	-	**2160**	-	**1.44**		
1_50_100_100	5002	14950	4391	97	-	**22226**	-	**60.59**		
2_10_20_10	202	590	67	10	35883	**471**	6.71	**0.04**	0.00019	**0.00008**
2_20_50_50	1002	2980	746	47	-	**3781**	-	**1.69**		
2_50_100_100	5002	14950	3689	98	-	**30343**	-	**49.82**		
3_10_20_10	202	590	98	13	50640	**602**	9.68	**0.07**	0.00019	**0.00011**
3_20_50_50	1002	2980	687	48	-	**2825**	-	**1.37**		
3_50_100_100	5002	14950	3546	100		**21049**	-	**34.56**		
6_10_20_10	202	1052	64956007	5172182	270846	**670**	81.05	**0.17**	0.00030	**0.00025**
7_10_20_10	202	1030	9878999	784658	398511	**2047**	112.14	**0.57**	0.00028	**0.00028**
8_10_20_10	202	1053	4514195	643302	224418	**1839**	65.66	**0.59**	0.00029	**0.00032**
9_10_20_10	10	17	11	10	108	**17**	0.00	**0.01**	0.0	**0.00059**
9_20_50_50	20	37	21	20	513	**37**	0.00	**0.01**	0.0	**0.00027**
9_50_100_100	50	97	51	50	3528	**97**	0.06	**0.01**	0.00002	**0.00010**
9_100_200_200	100	197	101	100	14553	**197**	0.50	**0.00**	0.00003	**0.0**
9_500_1000_1000	500	997	501	500	372753	**997**	68.13	**0.25**	0.00018	**0.00025**
9_1000_2000_2000	1000	1997	1001	1000	-	**1997**	-	**1.03**		
10_10_20_10	33	41	10	10	80	**30**	0.00	**0.00**	0.0	**0.0**
10_20_50_50	63	81	20	20	1160	**60**	0.05	**0.00**	0.00004	**0.0**
10_50_100_100	153	201	50	50	7400	**150**	0.63	**0.02**	0.00009	**0.00013**
10_100_200_200	303	401	100	100	29800	**300**	5.01	**0.08**	0.00017	**0.00027**
10_500_1000_1000	1503	2001	500	500	-	**1500**	-	**1.55**		
10_1000_2000_2000	3003	4001	1000	1000	-	**3000**	-	**6.53**		
합계	Lubore방법도 해를 구한 문제만 고려				4385092	**10395**	2635.38	**3.99**	0.00464	**0.00685**
평균					182712	**433**	109	**0.166**	0.0001	**0.0002**

참 고 문 헌

[1] 박순달, 「OR(경영과학)」, 삼정판, 민영사, 1991.

[2] 안재근, "네트워크에서의 치명호에 대한 연구", 서울대학교 박사학위논문, 1997.

[3] Ahuja, R.K., T.L. Magnanti, and J.B. Orlin, *Network Flows-Theory, Algorithms and Applications*, Prentice-Hall, 1993.

[4] Ball, M.O., R.L. Golden, and R.V. Vohra, "Finding the Most Vital Arcs in a Networks", *Operations Research Letters*, Vol.8 (1989), pp.73-76.

[5] Corley H.W., H. Chang, "Finding the n Most Vital Nodes in a Flow Network", *Management Science*, Vol.21, No.3(1975), pp.362-364.

[6] Dinic, E.A., "Algorithm for Solution of a Problem of Maximum Flow in Networks with Power Estimation", *Soviet Mathematics Doklady*, Vol.11(1970), pp.1277-1280.

[7] Durbin, E.P., "An Interdiction Model of Highway Transportation", *Memorandum RM-4945-PR*, May (1966).

[8] Ford, L.R., and D.R. Fulkerson, *Flows in Networks*, Princeton University Press, Princeton, NL, 1962.

[9] Lubore, S.H., H.D. Ratliff, G.T. Sicillia "Determining the Most Vital Link in a Flow Network", *Naval Research Logistics Quarterly*, Vol.18, No.4(1971), pp.711-713.

[10] Murty, K.G., *Network Programming*, Prentice-Hall, 1992.

[11] Ratliff, H.D., G.T. Sicilia, and S.H. Lubore, "Finding the n Most Vital Links in Flow Networks", *Management Science*. Vol.21, No.5(1975), pp.531-539.

[12] Wollmer Richard, Some Methods for Determining the Most Vital Link in a Railway Network, *Memorandum RM-3321-ISA*, 1963.

[13] Wollmer R.D., M.J. Ondrasek, "A Model for Targeting Strikes in an Loc Network", *Memorandum RM-5940-PR*, September(1969).