Journal of the Korean
OR/MS Society
Vol. 16, No. 1, June., 1991

수송 네트워크에서 최대물동량경로 문제의 최적해법

성기석* 박순달**

(An Optimal Algorithm for Maximum Origin-Destination Flow Path in the Transportation Network)

Seong Ki Suk and Park Soon Dal

Abstract

This paper studies an optimal algorithm for the Maximum Origin-Destinaton Flow Path (MODFP) in an acyclic transportation network.

We define a Pseudo-Flow for each arc so that it can give an upper bound to the total flow of a given path. And using the K-th Shortest Path algorithm we obtain upper bound of MODF which is decreasing as the number of searched path grows. Computational Complexity of optimal algorithm is $O((K+m)n^2)$, K being the total number of searched path.

We proved that the problem complexity of finding MODFP in an acyclic network is NP-hard, showing that the 3-satisfiability problem can be polynomialy reduced to this problem. And we estimated the average of the number K as being $(m/n)^{1.103}\text{Exp}(0.00689_{g}m)$ from the computational experiments.

* 강릉대학교 산업공학과

** 서울대학교 산업공학과

1. 서 론

도시간의 공공운송체계는 주로 화물의 운송과 관련하여, 철도를 이용하는 열차, 고속도로 및 일반도로를 이용하는 정기화물트럭, 정해진 항로와 일정에 따라 운항하는 콘테이너 선박, 화물 수송기 등을 들 수 있다. 반면 도시내의 공공운송체계는 주로 지역주민의 출퇴근 및 일상생활에 다른 이동과 관련하여, 도시 가로망에서 정해진 경로를 따라 운행하는 노선버스와 지하철을 들 수 있다. 그런데 이들 운송체계는 정해진 경로를 따라 운행하면서 그 경로상에서 경유하는 도시나 지점들 사이에 발생하는 운송수요를 충족시켜 준다는 점에서 일반적인 공통점을 갖고 있다.

철도나 고속도로의 노선, 열차의 운행노선, 정기화물 운송노선, 콘테이너선박 및 항공기의 운항노선 등을 정할 때, 각 지역간의 연계의 중요성과 화물 물동량등을 조사하여 가능한한 많은 화물을 운송하고 지역간의 연계를 이룰 수 있도록 해야 한다. 또, 도시내의 버스나 지하철의 운행노선을 정할 때에도 각 지역간의 승객수 등을 조사하여 가능한 한 많은 승객을 운송할 수 있도록 해야 한다.

일반적인 수송 네트워크에서 경로의 가치는, 그 경로가 경유함으로써 연결 가능하게 되는 모든 지역간의 증대되는 물동량의 합이라 할 수 있다. 즉, 일정한 출발지로 부터 도착지에 이르는 운송노선이 중도에 경유하면서 충족시켜주는 모든 지역들 사이의 물동량의 합이다.

이 논문에서는 무환인 수송네트워크에서 각 지역간의 물동량의 합이 최대인 경로를 찾는 문제의 최적해법을 제시한다. 이러한 연구는 다음과 같은 의의가 있다. 지금까지의 경로에 관한 문제는 대부분 네트워크상의 각 호(arc)나, 호에 의해 인접된 마디쌍들에 주어진 값을 다루었으나, 이 논문에서 다루는 문제는 네트워크상에서 곧바로 인접해있지 않드라도, 경로에 의해서 연결이 가능한 마디쌍들이 주어진 값을 다룬다.

2. 연구 배경

최단경로 문제는 여러가지의 경로 문제들 중 기본형태이다. Deo, Pang(1984) 등은 최단경로 문제의 연구현황을 잘 분류 정리하였다. 그들 외에 Dreyfus (1969), Pierce(1975), Golden, Magnanti(1977) 등도 경로 문제에 관한 연구현황을 검토하였다. 현재까지 연구된 바 있는 최단경로 문제의 해법의 계산 복잡도를 정리하면 [표1]과 같다.

또한 최단경로 문제에 여러가지 실제적인 제약들을 첨가한 문제들에 관해서도 많은 연구가 진행되고 있다. 최단경로 문제로부터 일반화된 유사한 경로문제들로서는, 모든 마디쌍 사이의 최단경로를 구하는 문제(All Pairs Shortest Path Problem), 다수의 최단경로를 구하는 문제(K-th Shortest Path Problem), 경로상의 총 부하량이 제약된 최단경로를 구하는 문제(Shortest Weight Constraind Path Problem), 각 마디의 도착 시간구간이 제약된 최단경로를 구하는 문제(Shortest Path with Time Window Problem) 등이 있다.

[표 1] 최단경로 문제의 계산 복잡도

네트워크형태	음의환	호의 길이	계산 복잡도	비고
무 환	없음	정수	$O(n^2)$	
유방향	없음	비음인 정수	$O(n^2)$	Dijkstra
	없음	정수	$O(n^3)$	Bellman-Ford
	있음	정수	NP-hard	
무방향	없음	정수	$O(n^3)$	Non-Bipartit Matching
	없음	정수	NP-hard	

다수최단경로 문제는 난이도가 NP-complete이나, 계산 복잡도가 구하고자하는 경로의 수 K의 다항식, 즉 입력크기(Imput Size)의 지수항으로 나타나므로 의사다항시간문제(Pseudo-Polynomial)로 취급되고 있다. Dreyfus(1969), Fox(1978), Lawler(1972), Shier(1979), Minieka(1974), Wongseelashote(1976), Katoh(1982), Yen (1971), Martins(1984) 등이 다수최단경로 문제의 해법을 개발하는데 많은 공헌을 하였다. 현재까지 연구된 바 있는 다수최단경로 문제의 해법의 계산복잡도를 정리하면 [표2]와 같다.

[표 2] 다수최단경로 문제의 계산 복잡도

재방문	네트워크형태	호의 길이	계산 복잡도	비고
허용함	유방향, 무방향	정수	$O(Kn \log n)$	Dreyfus
			$O(rn+Kn \log n)$	Fox
허용안함	무방향	비음인 정수	$O(Kn \log n)$	Katoh
	유방향, 무방향	정수	$O(K n^3)$	Yen

Current(1986) 등은 다목적 수송네트워크에 관련된 연구현황을 소개하였다. 그들은 수송 네트워크를 계획하는데 있어서 고려해야할 목표들을 비용, 가용성, 이윤, 지역의 균형, 환경적 요건 등으로 구분하고, 이들 목표들 간의 상충을 고려하는 수송계획모형과 해법에 관한 연구들을 분류한 후, 각각의 내용과 적용 가능한 실제를 요약 소개하였다.

Martins(1984)는 다수의 비교기준을 가지는 최단경로문제를 제안하고, 다수꼬리표(Multi-Labelling) 방법을 이용하여 유효최단경로의 집합을 구하는 해법을 제시하였다.

Current(1985) 등은 경로에 의해서 만족되는 수요를 최대화하고, 또한 그 경로의 길이를 최소화하는 두가지 목표를 고려한 다목적 경로문제(Maximum Covering/Shortest Path Problem)를 모형화하였다. 이 모형은 지방의 이동진료센타나 이동수리센타 등의 이동경로를 정하는데 적용할 수 있으며, 새로운 고속도로나 철도를 건설할 때, 그 경유지를 각 지역이나 도시의 인구밀도와 중요도를 고려하여 정하는 경우에도 적용할 수 있다.

또 그들(1987)은 수송네트워크에서, 운용자의 비용이라 할 수 있는 경로의 길이를 최소화하고 사용자의 비용이라 할 수 있는 경로에의 접근 거리를 최소화 하는 두가지 목표의 상충을 고려한 다목적 수

송네트워크 모형을 제안하고 최적해법을 제시하였다.

Current(1988)은 환적을 위한 장비를 사용해야하는 계층적 수송네트워크에서, 전체 비용을 최소화하도록 일차적 수송경로와 이차적 수송경로 및 환적장비의 위치를 정하는 수송네트워크 모형을 제안하고 다수최단경로를 이용한 발견적 해법을 제시하였다.

3. 최대물동량경로 모형

먼저 다음과 같이 기호를 정의하자.

$G=(V, A)$: 유방향 네트워크

V, A : G상의 모든 마디와 호의 집합.

$n=|V|$, $m=|A|$: G상의 마디와 호의 수.

$C=\{(u, v) \mid (u, v)\in V\times V, u<v\}$: G상에서 연결가능한 마디쌍의 집합

f_{uv} : 마디쌍 $(u, v)\in C$ 사이의 물동량

s, t : G상에서 정해진 출발마디와 도착마디.

Π : G상에서 s에서 t까지의 경로들의 집합.

Π_i : G상에서 s에서 마디 i까지의 경로들의 집합.

$P=(V_P, A_P)$: G상의 임의의 경로.

V_P, A_P : P상의 모든 마디와 호의 집합

$C_P=\{(u, v) \mid (u, v)\in V_P\times V_P, u<v\}$: P상에서 연결가능한 마디쌍의 집합

그리고, 유방향 네트워크에서 각 마디쌍에 대해서 물동량이 주어져 있고, 정해진 두 마디를 잇는 임의의 경로가 있다 하자. 이 때 그 경로상에서 연결 가능한 모든 마디쌍의 물동량의 합을 그 '경로의 물동량'이라 하자. 그리고 정해진 출발마디로부터 도착마디까지의 경로들 중 경로의 물동량이 최대인 경로를 최대물동량경로라 하고 그 경로의 물동량을 '최대물동량'이라 하자. 이러한 경로의 물동량과 최대물동량은 각각,

$W_P=\sum_{(u, v)\in C_P} f_{uv}$,

$W_P^*=\mathrm{Max}\{W_P \mid P\in\Pi\}$,

와 같이 나타내어진다. 또 이때 경로 P^*, $P^*\in\Pi$가 최대물동량경로이다. 여기서, 무환인 네트워크에서 최대물동량경로를 찾는 문제를 최대물동량경로 문제라 하고 그 최적해법을 만들어보자.

4. 문제의 난이도.

우선 최대물동량경로를 찾는 문제의 난이도를 알아보자. 그러기 위해서 먼저 3-Satisfiability(3-SAT) 문제를 보자. 3-SAT 문제는 다음과 같이 정의된다. x_1, x_2…를 부울변수라 하고, $\bar{x}_i$는 x_i의 여변수라고 하자. 그리고 하나의 부울변수 또는 여변수를 '문자'라고 하고, 임의의 3개의 문자들의 OR 연산으로 이루어진 식을 하나의 '절'이라하자. 그리고 2개 이상의 절들의 AND 연산으로 이루어진 식을 주어진 부울식이라 하자. 이 때 각 부울변수 x_1, x_2…들에 각각 참값 또는 거짓값을 정하여 줌으로써 주어진 부울식이 참값이 되도록 할 수 있는가를 결정하는 문제를 3-SAT 문제라 한다.

이러한 3-SAT 문제는 그 난이도가 NP-Complete임이 이미 알려져 있다. [Garey(1979)]. 따라서 이 3-SAT 문제가 어떠한 문제로 다항적으로 변환(Polynomially Reduce)되면, 그 문제의 난이도가 NP-hard임을 증명할 수 있다. 이것을 이용하여 다

음의 정리를 증명하기로 한다.

정리 1 : 무환 네트워크에서 최대물동량경로를 찾는 문제의 난이도는 NP-hard이다.

(증명) 3-SAT 문제가 최대물동량경로를 찾는 문제로 다항적으로 변환됨을 보인다.

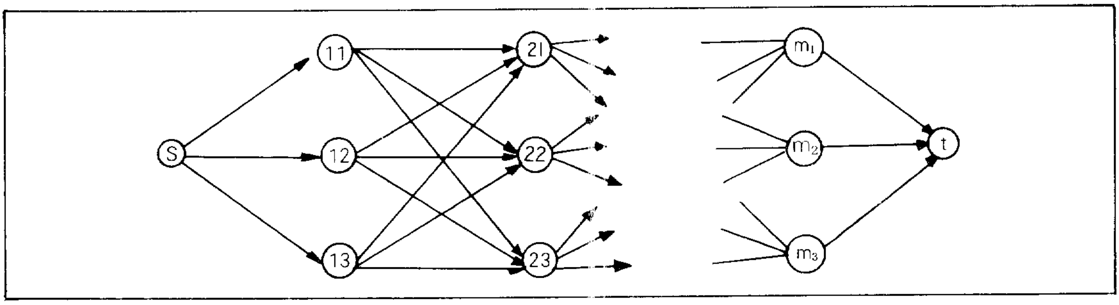
부울변수 x_1, x_2…들 또는 그의 여변수 $\sim x_1$, $\sim x_2$…들 중에 임의의 3개가 OR 연산으로 이루어진 m개의 절이 AND 연산으로 연결된 부울식이 다음과 같이 주어져있다 하자.

$$(P_{11} \vee P_{12} \vee P_{13}) \wedge (P_{21} \vee P_{22} \vee P_{23}) \wedge \cdots \wedge (P_{m1} \vee P_{m2} \vee P_{m3}).$$

여기서 P_{ij}는 주어진 부울식에서 i번째 절의 j번째에 주어진 부울변수 x_1, x_2…들 또는 그의 여변수 $\sim x_1$, $\sim x_2$ …들을 가리킨다. 이 P_{ij}를 주어진 부울식에서 i번째 절의 j번째에 주어진 '문자'라고 부르자.

이 주어진 부울식에 대해서 무환인 네트워크를 다음과 같이 생성시킨다. 먼저 부울식에 나타난 각 문자 P_{ij}에 대응하여 하나의 마디 V_{ij}를 만든다. 그리고 하나의 출발마디 s와 하나의 도착마디 t를 첨가한다. 그런 다음 부울식에서 순차적으로 연결된 절에 속한 문자에 대응하는 마디들 간에 다음과 같이 호를 연결한다. 즉, 부울식에서 서로 인접한 두 절에 대해서, 두 절에 포함된 문자들에 대응하는 마디들을 잇는 호를 만든다. 이렇게 만들어진 네트워크를 그림으로 나타내면 다음과 같다.

호의 집합 $A = \{(s, V_{1j}) \mid 1 \le j \le 3\}$
$\cup \{(V_{ij}, V_{i+1,k}) \mid 1 \le i \le m, 1 \le j, k \le 3\}$
$\cup \{(V_{mj}, t) \mid 1 \le j \le 3\}$.



위의 네트워크에서 출발마디 s로부터 도착마디 t에 이르는 모든 경로는, s와 t를 제외하고 m개의 마디를 가지는데, 이 마디들은 각각 m개의 절에 있는 3개의 문자들 중 하나에 대응한다. 그리고 이 마디에 대응하는 문자들이 모두 참값을 가지도록 하면, m개의 절이 모두 참값을 가지므로 주어진 부울식은 참값이 될 수 있다.

위와 같이 만들어진 무환 네트워크에서 각 마디쌍 (V_{ij}, V_{kl}), $1 \le i \le k \le m$, $1 \le j$, $l \le 3$들의 물동량을 다음과 같이 놓는다.

$$f(V_{ij}, V_{kl}) = \begin{cases} 1, & P_{ij}\text{가 } P_{kl}\text{의 여변수가 아닌 경우} \\ 0, & P_{ij}\text{가 } P_{kl}\text{의 여변수인 경우.} \end{cases}$$

그리고 여기에서 최대물동량경로를 찾으면 그 경로의 물동량이 $m(m-1)/2$인 경우와 그것보다 작은 경우가 발생한다.

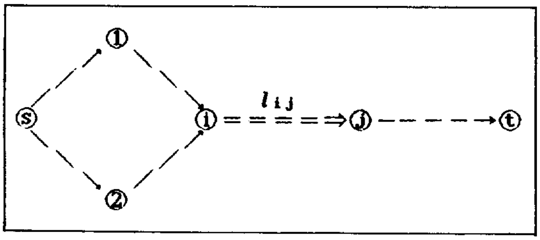
여기서 구한 최대물동량경로의 물동량이 $m(m-1)/2$인 경우에는, 그 경로상에 있는 마디에 대응하는 m개의 문자들 중에 서로 여변수인 문자들이 존재하지 않는다. 따라서 그 경로상에 있는 마디에 대응하는 m개의 문자들이 모두 참값이 되도록 각 부

울변수 x_1, x_2…들의 값을 정할 수 있다. 즉, 주어진 부울식이 참이 될 수 있음을 알 수 있다.

따라서 3-SAT 문제는 무환 네트워크에서 최대물동량경로를 찾는 문제로 다항적으로 변환된다. 그러므로 이 문제는 그 난이도가 NP-hard이다.

5. 최적해법.

먼저 각 호의 의사물동량을 정의해보자. 즉, 출발마디로부터 호(i, j)를 지나서 마디 j에 이르는 경로에 대해서, 그 경로상에 있는 마디들과 마디 j가 이루는 마디쌍의 물동량의 합의 최대값을 '호(i, j)의 의사물동량'이라 하자. 이러한 호(i, j)의 의사물동량는 $l_{ij}=Max\{\sum_{u\in V_P} f_{uj} \mid P\in\prod_i\}$와 같다. 이것을 예를 들어 그림으로 보이면 다음과 같다.



$$l_{ij}=Max\{f_{sj}+f_{1j}+f_{ij}, f_{sj}+f_{2j}+f_{ij}\}$$

그리고 임의의 경로 P에 대해서 그 경로상의 모든 호의 의사물동량의 합을 '경로 P의 의사물동량'이라 하자. 이러한 경로의 의사물동량은 $L_P=\sum_{(i,j)\in A_P} l_{ij}$와 같고 이것은 그 경로의 물동량의 상한이 된다.

정리 2 : 임의의 경로 P에 대해서, $L_P\ge W_P$이다.

증명) 경로 P의 물동량은 $W_P=\sum_{(u,v)\in C_P} f_{uv}$

$$W_P=\sum_{(u,v)\in C_P} f_{uv}$$
$$=\sum_{v\in V_P}\sum_{u\in V_P,\ u<v} f_{uv}$$
$$=\sum_{v\in V_P}\{\sum_{u\in V_P'} f_{uv}\}$$

단, P'은 경로 P에서 마디 v의 선행마디 r까지 부분. 즉 $P'\in\prod r$, $(r, v)\in A_P$이다. (단 π_i는 출발마디로부터 마디 i까지의 경로들의 집합) 따라서

$$\le\sum_{v\in V_P} Max\{\sum_{u\in V_P'} f_{uv} \mid P'\in\prod r,\ (r, v)\in A_P\}$$이다.

한편 호(i, j)의 의사물동량은 $l_{ij}=Max\{\sum_{u\in V_P} f_{uj} \mid P\in\prod i\}$이므로, 위식은 계속해서

$$=\sum_{v\in V_P} l_{rv},\ \text{단 } (r, v)\in A_P$$
$$=\sum_{(r,v)\in A_P} l_{rv}$$
$$=L_P$$

와 같이 나타내어진다. 즉, $L_P\ge W_P$임이 증명되었다.

그러면 여기서 각 호의 의사물동량을 구하는 방법을 보자. 우선 다음의 정리를 보자.

정리 3 : 주어진 네트워크에서 마디 k에서 출발하는 모든 호(k, r)들의 길이를 마디쌍(k, j)의 물동량 f_{kj}로 놓고, 정해진 출발마디로부터 마디 i까지 구한 최장거리에 마디쌍 (i, j)의 의사물동량l_{ij}이다.

(증명) 정리에서 서술한 값을 식으로 나타내면 $Max\{\sum_{(k,r)\in A_P} f_{kj} \mid P\in\prod i\}+f_{ij}$이다. 그런데, 마디 k에서 출발하는 호(k, r)들 중에서 경로 P에 속하는 것은 최대한 하나이므로 위의 식은

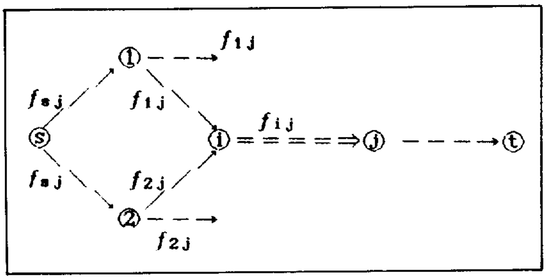
$$Max\{\sum_{k\in(V_P\setminus i)} f_{kj} \mid P\in\prod i\}+f_{ij}$$

와 같이 되고 이것은 다시

$$Max\{\sum_{k\in V_P} f_{kj} \mid P\in\prod i\}=l_{ij}$$

와 같이 된다. 따라서 정리가 성립한다.

위의 정리를 예를 들어 그림으로 보이면 다음과 같다.



$$l_{ij}=\mathrm{Max}\{f_{sj}+f_{1j}+f_{ij},\ f_{sj}+f_{2j}+f_{ij}\}$$
$$=\mathrm{Max}\{f_{sj}+f_{1j},\ f_{sj}+f_{2j}\}+f_{ij}$$
$$=\mathrm{Max}\{\textstyle\sum_{k\in(VP/i)} f_{kj} \mid P\in\prod i\}+f_{ij}$$

따라서 주어진 네트워크에서 각 마디 k에서 출발하는 모든 호의 길이를 마디쌍(k, j)의 물동량 f_{ij}를 더하면 호(i, j)의 의사물동량이다.

한편 일반적인 유방향 네트워크에서 최장경로 문제는 복잡도가 NP-hard이다. 그러나 주어진 네트워크가 무환인 경우에는 음의 환(Negative Cycle)이 존재하지 않으므로 각 호의 길이의 부호를 반대로 놓고 최단경로 문제로 풀면 된다. 따라서 최장거리는 $O(n^2)$만에 구할 수 있으며, 모든 호에 대한 의사물동량은 $O(mn^2)$만에 구할 수 있다.

이와 같이 각 호의 의사물동량을 구한 후, 주어진 네트워크에서 각 호의 길이를 그 호의 의사물동량의 음수값으로 놓고 다수최단경로 (K-th Shortest Path)를 이용하여 길이가 짧은 경로들부터 차례로 찾아 나간다. 그러면 찾아지는 경로의 길이의 음수값이 그 경로의 의사물동량이고 또 그 값이 높은 차례로 경로들을 찾을 수 있다.

여기서 다수최단경로를 이용하여 경로들을 차례로 구할 때, k번째에 찾아진 경로를 P(k), 그 경로의 물동량을 $W_{P(k)}$, 그 경로의 의사물동량을 $L_{P(k)}$라고 표현하자.

임의의 차례에 구해진 경로의 의사물동량보다 그에 앞서서 구해진 어떠한 경로의 물동량이 크면, 그 경로의 물동량은 그 이후에 구해지는 모든 경로의 물동량보다 항상 크다. 또한 임의의 차례에 구해진 경로의 의사를 동량보다 그에 앞서서 구해진 경로의 물동량 중에서 가장 큰 값이 더 크면, 그 값이 최대물동량이며, 그 값을 주는 경로가 최대물동량경로이다. 다음의 정리를 보자.

정리 4 : $W_{P(h)}=\mathrm{Max}\{W_{P(i)} \mid 1\leq i\leq k\}$이고, $W_{P(h)}\geq L_{P(k)}$이면, 모든 t에 대해서 $W_{P(h)}\geq W_{P(t)}$이다.

(증명) 다수최단경로를 이용하여 그 값이 높은 순서대로 찾은 것이므로, t>k이면, $L_{P(k)}\geq L_{P(t)}$이다. 한편 정리2에 의해서 $L_{P(t)}\geq W_{P(t)}$이므로 모든 t>k에 대해서 $L_{P(k)}\geq W_{P(t)}$이다.

따라서 $W_{P(h)}\geq L_{P(k)}$이면, 모든 t>k에 대해서 $W_{P(h)}\geq W_{P(t)}$이다.

한편 $W_{P(h)}=\mathrm{Max}\{W_{P(i)} \mid 1\leq i\leq k\}$이므로, 모든 t, $1\leq t\leq k$에 대해서도 $W_{P(h)}\geq W_{P(t)}$이다.

따라서 모든 t에 대해서 $W_{P(h)}\geq W_{P(t)}$이다.

따라서 최대물동량경로를 찾는 알고리즘은 다음과 같다.

＊최대물동량경로를 구하는 알고리즘.

단계 0 : 각 호 (i, j)에 대한 의사물동량 l_{ij}를 구한다.

단계 1 : 각 호 (i, j)의 길이를 의사물동량의 음수값인 $-l_{ij}$로 둔다.

반복횟수 k=0, 후보 경로 P*={ }라고 두고 최대물동량의 상한을 L=+∞, 최대물동량의 하한을 W=−∞라고 둔다.

단계 2 : k=k+1로 둔다.

k번째 최단경로 P(k)와 그 길이의 음수값을 $L_{P(k)}$로 둔다.

구한 경로 P(k)의 물동량을 다음과 같이 구한다.

$W_{P(k)}=\sum_{(u,\ v)\in C_P} f_{uv}$.

단계 3 : 만약 $W<W_{P(k)}$이면 $W=W_{P(k)}$, P*=P(k)라고 둔다.

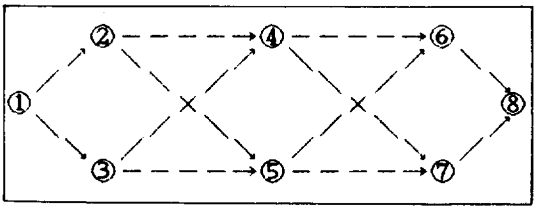
만약 $L>L_{P(k)}$이면 $L=W_{P(k)}$라고 둔다.

만약 L≤W이면 P*를 최적해로 하고 끝낸다.

아니면 단계2로 간다.

6. 수치예제.

다음의 예를 보자. 주어진 네트워크는 [그림1]과 같고 각 마디쌍에 대한 물동량은 [표3]과 같다하자.

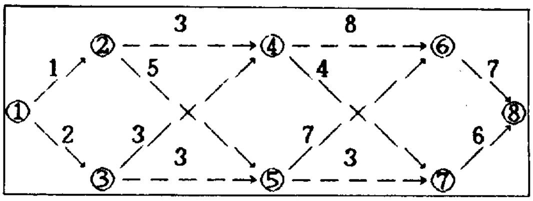


[그림 1] 네트워크

[표 3] 마디쌍의 물동량 f_{uv}

	2	3	4	5	6	7	8
1	1	2	2	1	3	1	1
2			1	4	1	1	2
3			1	2	2	1	1
4					3	2	1
5					2	1	2
6							2
7							1

우선 각 호의 의사물동량을 구하면 다음의 [그림 2]와 같다.



[그림2] 각 호의 의사물동량.

단계 0 : K=0, P*={ },

L=+∞, W=−∞

단계 1 : K=1, P(1)=1−3−4−6−8,

$L_{P(1)}=20$,

$W_{P(1)}=(2+2+3+1)+(1+2+1)+(3+1)+2=18$

단계 2 : $L=20$

단계 3 : $W=18$, $P^*=P(1)$

단계 1 : $K=2$, $P(2)=1-2-5-6-8$,

$L_{P(2)}=20$,

$W_{P(2)}=(1+1+3+1)+(4+1+2)+(2+2)+2=19$

단계 2 : $L=20$

단계 3 : $W=19$, $P^*=P(2)$

단계 1 : $K=3$, $P(3)=1-3-5-6-8$,

$L_{P(3)}=19$,

$W_{P(3)}=(2+1+3+1)+(2+2+1)+(2+2)+2=18$

단계 2 : $L=19$

단계 3 : $W=19$, $P^*=P(2)$, Stop.

따라서 최대물동량경로는 1－2－5－6－8이고, 최대물동량은 19이다.

7. 계산복잡도.

앞에서 제시한 계산방법의 계산복잡도는 크게 세 부분으로 나누어진다. 즉, 각 호의 의사물동량을 구하는 단계의 복잡도, 각 k번째 최단경로를 구하는 단계의 복잡도, 최적해가 될 때까지 구해야하는 k번째 최단경로의 수등이다.

주어진 네트워크의 마디의 수를 n, 호의 수를 m이라 하고 계산복잡도를 보자. 각호의 의사물동량을 구하는 단계의 계산복잡도는 최단경로를 구하는 계산복잡도와 같으므로 n^2이고, 이 계산단계를 각 호에 대해서 적용하여야 하므로 mn^2이 된다. 또한 구해진 각 호의 의사물동량을 이용하여 k번째 최단경로를 구하는 계산단계도 역시 최단경로를 구하는 계산복잡도와 같아서 n^2이다. 그리고 최적해를 구할 때까지 반복적으로 구해야 하는 k번째 최단경로의 수는 문제의 특성에 따라 달라 지는데, 이 수를 K라 하면 위의 계산방법의 총 계산복잡도는 $O((K+m)n^2)$이 된다.

여기서 최적해를 구할 때까지 반복적으로 구해야 하는 최장경로의 수 K의 상한은 마디수에 대해 지수식으로 늘어난다. 따라서 위의 계산복잡도는 지수식시간복잡도(指數式時間複雜度)를 가진다.

한편 전산실험을 통하여 주어지 문제의 마디와 호의 수가 증가함에 따라 반복회수가 어떻게 증가하는가를 알아 보았다.

문제를 풀기 위한 프로그램은 Pascal Language로 작성하였으며 프로그램의 수행은 Micro-VAX Ⅱ를 사용하였다. 전산실험에 사용한 문제는 마디의수 호의수 및 마디쌍의 물동량 분포에 따라 총 600개의 문제를 생성하여 사용하였다. [표4]에 실험 결과의 일부를 나타내었다. 그리고 마디수와 호의 수의 증가에 따른 반복회수의 증가를 회귀식으로 추정해본 결과, $K=(m/n)^{1.103}\ \mathrm{Exp}(0.00689m)$와 같이 구해졌다.

8. 결 론

이 논문에서는 무환인 수송 네트워크에서 정해진 출발지로부터 도착지에 이르는 경로들 중에서, 경로상에서 연결 가능한 지점들 사이의 물동량의 합이 최대인 경로를 찾는 문제의 최적해법을 연구하였다.

최대물동량경로를 찾는 최적해법은 다음과 같다.

우선' 네트워크상의 마디쌍의 물동량을 호의 의사물동량으로 전환한다. 이 호의 의사물동량은 그 호를 지나는 경로의 물동량 상한을 구할 수 있도록 주어지며, 그것을 구하는 계산복잡도는 $O(n^2)$이다. 이러한 각 호의 물동량을 이용하여 구한 경로의 물동량 상한을 경로의 의사물동량이라 하고, 이 경로의 의사물동량이 높은 경로를 차례로 찾아서 이것을 이용하여 원문제의 상한을 구한다. 동시에 차례로 찾은 경로의 물동량을 구함으로써 원문제의 하한을 구한다. 경로를 계속 찾아감에 따라 원문제의 상한과 하한이 좁혀지다가 상한과 하한값이 일치하게 될 때 최적해를 구하게 된다. 한편 경로의 의사물동량이 높은 순서로 경로들을 찾는 것은 다수최단경로의 해법을 이용한다.

한편 위 문제의 난이도가 NP-hard임을 보였고, 제시한 최적해법의 반복회수가 주어진 문제의 마디와 호의 수가 증가함에 따라 어떻게 증가하는지 600개의 예제에 대하여 전산 실험하고 마디와 호의 수에 대한 회귀식으로 추정해본 결과, $K=(m/n)^{1.103}$ $\mathrm{Exp}(0.00689m)$와 같이 구해졌다.

[표 4] 최적해법의 반복회수(K)

마디 수	호의 수	반복 회수	계산시간 (초)
60	384	296	104.53
60	326	9	21.44
60	219	3	13.85
60	134	1	8.66
50	335	31	22.62
50	249	10	13.62
50	186	4	9.20
50	117	2	5.72
40	263	71	19.33
40	215	4	7.67
40	147	4	5.09
40	91	3	3.36
30	203	4	5.38
30	171	7	4.61
30	108	2	2.52
30	72	1	1.59
20	133	1	2.07
20	92	2	1.35
20	64	2	0.97
20	38	2	0.66

—參考文獻—

1. 성기석, 「수송네트워크에서 최대물동량경로에 관한 연구」, 서울대학교 대학원 공학박사 학위논문, 1990
2. Ceder, A., N.H.M.Wilson, "Bus Network Design", Transportation Research-B, Vol.20B, No.4, 1986, 331-344.
3. Current, J.R., H.Min, "Multi-Objective Design of Transportation Nertworks : Taxonomy and Annotation", European J. of Operational Research, Vol. 26, 1986, 187-201.
4. Current, J.R., C.S.Revelle, J.L.Cohon, "The Median Shortest Path Problem : A Multiobjective Ap-

proach to Analyze Cost vs. Accessibility in the Design of Transportation Networks", Transportation Science, Vol.21, No.3, 1987, 188–197.

5. Current, J.R., "The Design of a Hierarchical Transportation Network with Transshipment Facilities", Transportation Science, Vol. 22, No.4, 1988, 270–277.
6. Deo, N., C. Pang, "Shortest Path Algorithms : Taxonomy Annotation", Networks, Vol.14, 1984, 275 ~323
7. Dreyfus, S.E., "An Apprisal of Some Shortest-Path Algorithms", Operations Research, Vol.17, 1969, 395–412.
8. Fox, B.L., "Data Structures and Computer Science Techniques in Operations Research", Operations Reseanch, Vol.26, 1978, 686~717.
9. Garey, M.R., D.S.Johnson, Computers and Intractability—A guide to the theory of NP-Completeness, W.H.Freeman and Company, 1979.
10. Golden, B.L., T.L. Magnanti, "Deterministic Network Optimization—A Bibliography", Networks, Vol.7, 1977, 149–183.
11. Katoh, N., T.Ibaraki, H.Mine, "An Efficient algorithm for K Shortest Simple Path", Networks, Vol. 12, 1982, 411–427.
12. Lawler, E.L., "A Procedure for Computing the K Best Solutions to Discrete Optimization Problems and Its Application to the Shortest Path Problem", Management Science, Vol.18, No.7, 1972
13. Mandl, C.E., "Evaluation and Optimization of Urban Public Transportation Networks", European J. of Operational Research, Vol. 5, 1980, 396–404.
14. Magnanti, T.L., R.T.Wong, "Network Design and Transportation Planning : Modeis and Algorithms", Transportation Science, Vol.18, 1984, 1–55.
15. Martins, E.Q.V., "An Algorithm for Ranking Paths that may contain Cycles", European J. of Operational Research, Vol.18, 1984, 123–130.
16. Pierce, A.R., "Bibliography on Algorithms for Shortest Path, Shortest Spanning Tree, and Related Circuit Routing Problems(1956~1974)", Networks, Vol.5, 1975, 129–149
17. Perko, A., "Implementation of Algorithms for K Shortest Loopless Paths", Networks, Vol.16, 1986, 149–160.
18. Minieka, E., "On Computing Sets of Shortest Paths in a Graph", Comm. ACM, Vol.17, 1974, 351–353.

19. Shier, D.R., "On Algorithms for Finding the K-Shortest Paths in a Network", Networks, Vol.9, 1979, 195-214.

20. Wongseelashote, A., "An Algebra for Determining All Path-Values in a Network with Application to K-Shortest-Paths Problems", Networks, Vol.6, 1976, 307-334.

21. Yen, J.Y., "Finding the K Shortest Loopless Paths in a Network", Management Science, Vol.17, 1971, 712-716.