Journal of the Korean Institute
of Industrial Engineers
Vol.23, N0.3, September, 1997

일반화된 네트워크에서 최단흐름생성경로문제

The Shortest Flow-generating Path Problem in the Generalized Network

정성진* · 정의석**

S. J. Chung* · E. S. Chung**

Abstract

In this paper, we introduce the shortest flow-generating path problem in the generalized network. As the simplest generalized network model, this problem captures many of the most salient core ingredients of the generalized network flows and so it provides both a benchmark and a point of departure for studying more complex generalized network models.

We show that the generalized label-correcting algorithm for the shortest flow-generating path problem has O(mn) time complexity if it starts with a good point and also propose an $O(n^3m^2)$ algorithm for finding a good starting point. Hence, the shortest flow-generating path problem is solved in $O(n^3m^2)$ time.

1. 서론

일반화된 네트워크(generalized network)는 흐름의 보존·유통이라는 단순 네트워크흐름의 개념을 확장하기 위해 단순네트워크(ordinary network)의 각 링크에 흐름이득계수(flow gain factor)를 부가설정한 네트워크이다.

즉, 일반화된 네트워크는 각 링크흐름의 선형변환-링크e에 유입된 흐름량 f_e는 흐름이득계수 r_e(>0)배로 변환되어 흐름 r_ef_e가 유출되는-을 전제로 한다. 따라서 이러한 일반화된 네트워크의 개념은 링크흐름의 보존(r_e=1)을 전제로 하는 단순네트워크모형으로 다룰 수 없었던 많은 문제의 모형화를 가능하게 하였

* 서울대학교 산업공학과
** 한국통신 통신망연구소

다[2].

그런데 단순네트워크흐름은 그 이론적 체계가 확고하게 정립되어 있지만, 일반화된 네트워크흐름은 그렇지 못한 상태로 각 연구가들의 연구가 개별적인 입장에서 이뤄져 왔다. 따라서 통합된 시각에서의 이론적 정립이 요구된다.

단순네트워크흐름문제의 확고한 이론적 체계는 최단경로문제(shortest path problem)와 최대흐름문제(max flow problem)라는 두 가지 기본문제를 양축으로 구성된다[2]. 즉, 단순네트워크흐름문제는 노드에서의 흐름의 공급과 수요, 링크에서의 흐름용량, 링크의 흐름비용을 바탕으로 하는 데, 이는 결국 상보적인 두 요소-흐름(flow)과 잠재가(potential)-를 통해 분석된다. 최대흐름문제는 흐름만을 극단적으로 대표하는 기본문제이고, 최단경로문제는 잠재가만을 극단적으로 대표하는 기본문제이다. 이 두 문제의 강성다항식 해법은 네트워크흐름을 대표하는 문제, 최소비용흐름문제의 여러 강성다항식 해법[2,5,6,20,21]에 씨줄과 날줄처럼 얽혀져 있다. 특히, 네트워크단체법에서 기저순환방지를 위한 강성가능기저(strongly feasible basis[19])개념은 최단경로문제의 최적기저개념과 동일하고, 또한 최단경로문제의 강성다항식 라벨수정법[7,9]은 네트워크단체법에서 기저정체현상의 방지와 밀접한 관련이 있다[3,10,19].

위에서 언급된 단순네트워크흐름문제의 이론적 체계는 일반화된 네트워크흐름문제에 시사하는 바가 많다. 다시 말하자면, 일반화된 네트워크흐름의 통합된 시각에서의 이론 정립을 위해서는 흐름과 잠재가 양극단을 대표하는 기본문제의 설정하고, 이의 강성다항식 혹은 다항식복잡도의 좋은 알고리듬을 개발하는 것이 요구된다. 현재, 일반화된 네트워크에서 흐름만을 극단적으로 대표하는 일반화된 최대흐름문제(generalized max flow)는 많이 연구되어 있고[4,11,12,15,16,19], 1991년에 Goldberg등[8]에 의해 이 문제의 다항식해법이 발표되었다. 그러나, 잠재가만을 극단적으로 대표하는 기본문제-일반화된 최단경로문제-는 공식적으로 언급된 바는 없다.

따라서, 본 연구에서는 II장에서 잠재가를 대표하는 가장 간단한 형태의 기본문제로 최단흐름생성경로문제를 제안하고, III장에서 이것이 적합하게 정의된(well-defined) 문제임을 문제의 원 · 쌍대구조를 통해 알아본다. 또한, 잠재가와 관련된 몇 가지 사항을 정리하기로 한다. IV장에서는 최단경로문제의 해법인 라벨수정법을 일반화한 일반화된 라벨수정법을 제시하고, 이 알고리듬이 '특별한 초기잠재가'에서 출발하면 O(mn)에 종료될 수 있음을 보이고, '특별한 초기잠재가'를 찾는 문제 역시 강성다항식복잡도로 해결될 수 있음을 보인다.

2. 최단 흐름생성경로문제

최단경로문제는 네트워크의 이산수학적 구조(combinatorial structure)만으로 취급이 가능하지만, 최소비용흐름문제의 범주에서 가장 단순한 흐름구조를 갖는 문제로 이해할 수 있다.

(P)

$$\min\{\, \Sigma_{ij} c_{ij} f_{ij} \mid \Sigma_i f_{ik} - \Sigma_j f_{kj} = b_k \quad \forall k \in V,\ f_{ij} \geq 0\ \forall (i,j) \in E \}$$

즉, 두 노드 s, t를 연결하는 최단경로는 노드s와 t에서 단위흐름의 유입・방출되는 최소비용의 흐름으로 환원된다(b_t=-b_s=1, b_i=0, $\forall i \neq s,t$).

또한, 노드 s와 다른 모든 노드간의 최단경로문제는 부분최적의 원리("최단경로의 부분경로는 부분경로의 두 노드를 연결하는 경로 중에서 최단경로이다")에 의해 하나의 문제(b_s=1-n, b_i=1, $\forall i \neq s$)로 취급이 가능하다.

이제, "경로 ≡ 가장 단순한 흐름궤적" 이라는 관계에서 출발하여, 일반화된 네트워크에서 최소비용흐름문제로서 최단경로문제를 정의하자.

(GP)

$$min\{ \Sigma_{ij} c_{ij} f_{ij} \mid \Sigma_i r_{ik} f_{ik} - \Sigma_j f_{kj} = b_k \quad \forall k \in V, f_{ij} \geq 0 \; \forall (i,j) \in E\}$$

문제 (GP)에서 흐름이득계수는 양의 실수이다. 즉, r_{ij}>0 이다.

단순네트워크흐름에서는 흐름의 보존을 특징으로 하지만($\Sigma_k b_k = 0$), 일반화된 네트워크는 그 흐름의 선형변환이라는 특징 때문에 흐름의 내부생성과 내부흡수가 가능하다($\Sigma_k b_k \neq 0$). 링크 (i,j)에 유입된 흐름 f_{ij}는 링크를 통과하면서 $r_{ij}f_{ij}$의 흐름이 유출되는데, 만약 링크의 유출흐름을 링크의 처음 노드로 환류시킨다면 (i=j), $(r_{ij}$-1$)f_{ij}$의 잉여흐름이 발생한다. 즉, 흐름이득계수가 1보다 크다면 흐름의 생성, 반대의 경우는 흐름의 흡수가 이루어지는 것이다. 이러한 흐름의 생성・흡수원리는 닫힌 경로($i \rightarrow i$), 즉 환에서도 적용가능하고 이는 환을 구성하는 모든 링크의 흐름이득계수를 곱한 환의 흐름이득계수로 판단할 수 있다.

따라서, 일반화된 네트워크의 흐름특성을 반영하는 가장 단순한 형태의 흐름궤적은 네트워크 내부에서 생성된 흐름을 단일노드에서 단위흐름을 방출하는 경우로 상정할 수 있다(b_t=1, b_i=0, $\forall i \neq t$). 이때의 흐름궤적은 흐름을 생성하는 환(cycle)과 환에서 생성된 흐름을 노드 t로 보내는 경로(path)로 구성된다. 이를 *t-흐름생성경로*(*t-flow generating path*)라 정의한다. 이때 노드 t가 환에 속하는 경우는 환 자체만으로도 t-흐름생성경로이다.

일반화된 네트워크에서도 부분최적의 원리([보조정리 2])가 성립하여 각각의 노드 t에서 최단 t-흐름생성경로를 찾기 위하여 n개의 개별적인 문제를 다루지 않고, 하나의 문제로 취급이 가능해진다(b_i=1, $\forall i \in V$). 그리고, 이때의 문제를 *최단흐름생성경로문제*라 정의한다.

3. 최단흐름생성경로문제의 구조

3.1 원구조(primal structure)

일반화된 네트워크흐름문제의 기저는 단순네트워크흐름문제의 기저인 트리구조를 포괄하는 유사포리스트(quasi forest)구조를 갖는다. 유사포리스트는 유사트리(quasi tree)의 집합이며, 유사트리는 트리에 링크가 하나 더 첨가된 구조로 환에 몇 개의 트리가 부착된 형태이다. 이 때 유사트리의 환을 수축(contraction)하면 트리가 된다[2].

특히, 최단흐름생성경로문제의 기저는 외향성(out-directed) 유사포리스트이며, 그 구조 및 특성은 최단경로문제의 기저인 외향성 트리를 함축하고 있다.

여기서, 외향성트리는 루트노드에서 다른

노드까지의 경로를 구성하는 링크의 방향이 모두 경로의 방향과 일치하는 트리를 말하고, 외향성유사트리는 다음 두 가지 성질을 만족하는 유사트리를 말한다(그림 1).

① 유사트리의 환을 수축(contraction)하였을 때 만들어지는 트리가 외향성트리이다.
② 유사트리의 환을 구성하는 각 링크의 방향과 환을 일주하는 방향이 일치하고, 환을 구성하는 모든 링크의 호름이득계수의 곱이 1 보다 커야한다.

외향성유사포리스트는 각각의 노드를 머리노드로 하는 유일한 링크를 갖는다. 이 링크가 (i, j) 라면 노드 i는 노드 j의 전위자(前位者. predecessor)이며, 노드의 함수 $p: V \rightarrow V$를 사용하여 $p(j) = i$라 표기한다. 전위자는 트리구조를 일의적으로 표현하는 자료구조이지만, 유사트리구조 또한 일의적으로 표현해준다. 그리고, 유사트리상의 j-흐름생성경로는 노드 j에서 시작하여 전위자를 회귀적으로 이용하여 구할 수 있다. 즉, 전위자의 전위자를 계속 추적해나가면 (즉, $j \rightarrow p(j) \rightarrow p((j)) \rightarrow p(((j))) \rightarrow \cdots$) 탐색한 노드 중에서 다시 만나게 되는 노드가 있는데, 여기까지가 흐름생성경로를 구성하는 노드의 순열이고, 이 노드에서 흐름생성경로를 구성하는 환과 경로가 연결된다. 그림 1에서 굵은 선으로 표시된 링크가 j-흐름생성경로이다. 그리고, 노드 i가 j-흐름생성경로에 포함되면 i-흐름생성경로를 부분경로로 갖는다.

유사포리스트에 해당하는 가능기저행렬은 다음 몇 가지 흥미로운 성질을 가지고 있다.

[보조정리 1] 가능기저의 단조성

문제 (GP)의 우변상수가 $b \rangle 0$일 때의 가능기저 B는 $0 \le b' \le b$를 만족하는 모든 b'에 대하여 가능기저이다.

즉, $B^{-1}b \geq 0,\ b \rangle 0 \Rightarrow B^{-1}b' \geq 0\ \forall b',\ 0 \leq b' \leq b$

(증명)[19] ■

e는 모든 원소가 1인 벡터이고, e_j는 j번째 원소가 1이고 나머지 원소는 0인 벡터라 하자.

[보조정리 2] 부분최적의 원리

최단호름생성경로문제 ((GP)에서 $b=e$인 경우)의 최적기저를 B라 하자. 그러면, 이 때의 B는 최단 j-흐름생성경로문제((GP)에서 $b=e_j$인 경우)의 최적기저가 된다.

(증명) 먼저, A를 일반화된 네트워크의 인접행렬이라 하고, 편의상 각 열의 순서는 다음과 같이 배열되었다고 가정한다. $A=[B,N]$ $c=(c_B,c_N)$

B가 최단흐름생성경로문제의 최적기저일 필요충분조건은 다음과 같다.

원가능기저조건: $B^{-1}e \geq 0$

쌍대가능기저조건: $\pi N \leq c_N$, $\pi = c_B B^{-1}$

위를 만족하는 B가 최단 j-흐름생성경로문제의 최적기저가 되려면 원 · 쌍대가능기저조건을 만족하여야 한다. 비용계수 c가 변하지 않았으므로 쌍대가능기저조건은 자동적으로 충족된다. 최단 j-흐름생성경로문제의 우변상수는 e_j이므로 이때의 원가능기저조건은 $B^{-1} e_j \geq 0$이다. 그런데, $B^{-1} e \geq 0$, $e \geq e_j$이므로 [보조정리 1]에 의해 $B^{-1}e_j \geq 0$, 즉, B는 최단 j-흐름생성경로문제의 최적기저가 된다. ■

위 정리에서 최적유사포리스트상의 j-흐름생성경로(B^{-1}의 j번째 열)는 최단j-흐름생성경로($B^{-1}e_j$) 임을 알 수 있다. 또한, 최단 j-흐름생성경로상의 노드 i로 결정되는 부분경로(i-흐름생성경로)는 최단 i-흐름생성경로가 되는 부분최적의 원리가 성립함을 알 수 있다.

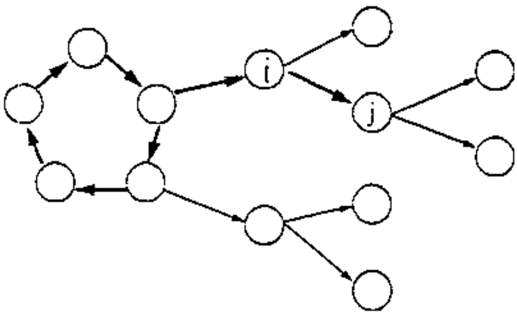

그림 1. 외향성유사트리

3.2 쌍대구조(dual structure)

최단흐름생성경로문제의 쌍대문제는 다음과 같다.

$$max\{ \Sigma_k d_k: -d_i + r_{ij} d_j \le c_{ij}, \forall(i,j) \in E\}$$

쌍대변수 d_j는 노드 j의 잠재가라 부른다. 그리고 최적잠재가 d_j*는 최단j-흐름생성경로의 비용과 동일하다.

개개의 쌍대부등식($-d_i + r_{ij} d_j \le c_{ij}$)은 각각의 링크($i,j$)와 일대일 대응하는데, 두 노드간의 잠재가에 관한 관계식으로, 한 노드의 잠재가가 고정되면 다른 노드의 잠재가의 상한(혹은 하한)을 얻을 수 있다는 특징이 있다. 이러한 특징은 방향이 일치하는 링크의 순열($s \equiv v_0 \rightarrow v_1 \rightarrow \cdots \rightarrow v_{N-1} \rightarrow v_N \equiv t$)로 정의되는 s,t-행로 (s,t-walk) W에도 적용될 수 있다. 여기서 s,t-행로는 이를 구성하는 각 노드가 중복되는 것도 허용한다. 단, 노드 s,t는 단 한번 나타난다고 가정한다. 즉, s,t-행로를 구성하는 각 링크와 대응되는 쌍대부등식의 집합을 생각해보자. 편의상 노드 v_i는 노드 i라하고, 링크 (v_{i-1}, v_i)의 이득계수와 비용계수는 각각 r_i, c_i라 하자.

$$\{-d_{i-1} + r_i\ d_i \le c_i,\ i=1,\cdots,N\}$$

k번째 부등식에 $\Pi_{i=1}^{k-1} r_i$배하여 모든 부등식을 더하면 변수 $d_i,\cdots,d_{N-1}$은 소거되어 다음의 부등식이 얻어진다.

$$-d_s + r(W)\, d_t \le c(W) \qquad \cdots \text{(식1)}$$

여기서, $r(W) = \Pi_{i=1}^{N} r_i$, $c(W) = \Sigma_{k=1}^{N} c_k (\Pi_{i=1}^{k} r_i)$

이 부등식은 쌍대부등식의 양의 선형결합으로 만들어진 것이므로 쌍대부등식체계에 추가되어도, 쌍대해집합의 구조를 변화시키지 않는 유효한 부등식(valid inequality)이 된다. 행로를 따라 얻어진 부등식 (식1)은 최단흐름생성경로에 관한 여러 유용한 정보를 준다. 이에 관해서 다음 두절에서 자세히 살펴보자. 단, 이후 언급되는 행로는 특별한 언급이 없어도 길이가 n이하인 것으로 가정한다.

3.2.1 잠재가의 조건부 상한

노드 s의 잠재가를 고정하면 ($d_s=g$), (식1)에서 행로 W를 따라 노드 t의 잠재가의 상한을 얻을 수 있다. 이 때 얻어지는 상한은 g의 함수 $\mu_t(g;W)$로서, g의 값에 따라 변하는 조건부상한이다. 즉,

$$d_t \le r(W)^{-1} g + r(W)^{-1} c(W) \equiv \mu_t(g;W)$$

조건부상한은 적절한 g 값에 따라 유효부등식을 제공한다.

[정리 1] $d_s \le g$ 이 유효부등식이면 $d_t \le \mu_t(g;W)$ 는 유효부등식이다.

(증명)부등식 $d_t \le \mu_t(g;W)$는 두 유효부등식 $-d_s + r(W)\, d_t \le c_t(W)$와 $d_s \le g$의 합으로 구성된 부등식이므로 유효부등식이다. ■

노드 s의 잠재가를 고정시킬 때 얻을 수 있는 잠재가 d_t의 조건부상한중 최소는 조건부상한중 가장 좋은 상한(tight bound)을 제공하는데, 다음의 알고리듬 *walk*에 의해 구할 수 있다. 이는 잠재가의 초기값(입력요소 *a*)을 $a_s=g$, $a_i=\infty$, $\forall i \neq s$로 놓고 구한 $d_t^{[n]}$이 된다.

```
Algorithm walk(a)
for j=1 to n do d_j^[0] ← a_j
for k=1 to n do
    for (i,j)∈E do
        d_j^[k] ← min{d_j^[k-1], min(d_i^[k-1] + c_ij)/r_ij) }
    end
end
```

그림 2. 알고리듬 walk

[정리 2] 알고리듬 *walk*의 계산 복잡도는 $O(nm)$이다.

(증명) 내부 루프는 모든 링크에 대한 연산이므로 $O(m)$이고 외부 루프는 n회 수행되므로 총계산 복잡도는 $O(nm)$이다. ■

[정리 3] 알고리듬 *walk* ($a_s=g$, $a_i=\infty$, $\forall i \neq s$)에서 얻어진 $d_j^{[k]}$는 g에 관한 선형함수이고 길이가 k이하인 s,j-행로 중에서 얻을 수 있는 최소의 조건부상한이다.

(증명) 귀납법

$k=0$, $d_j^{[k]}$=g 또는 ∞ 이므로 자명하다.

[귀납가설]$k \le l$일 때 성립한다고 하자.

그러면 $k=l+1$일 때 다음 두 가지 경우가 발생한다.

① $d_j^{[l+1]} \equiv d_j^{[l]}$

이 경우 $d_j^{[l+1]}$ 이 단계 l에서와 동일하므로 자명하다.

② $d_j^{[l+1]} \equiv (d_i^{[l]} + c_{ij})/r_{ij}$

이 경우 $d_j^{[l+1]}$ 이 단계 l에서의 길이가 l 이하인 s,i-경로와 링크 (i,j)를 통해 얻어졌으므로 그 경로의 길이가 $l+1$ 이하이다. 그리고, $d_i^{[l]} = \alpha g + \beta$이라면, $d_j^{[l+1]} = (\alpha/r_{ij})g + (\beta + c_{ij})/r_{ij}$ 형태의 g에 관한 선형함수이다. ■

지금까지의 논의에서 조건부상한의 최소를 길이가 n 이하인 행로를 대상으로 구하였는데, 만약 단순경로만을 대상으로 한다면 문제의 범위가 간단해질 것이라고 생각할 수 있다. 그러나 불행히도 단순경로를 따라 조건부상한의 최소를 구하는 문제는 *NP-hard*이다.

[정리 4] 단순경로를 따라 조건부상한의 최소를 구하는 문제(편의상 문제 U라 약칭하자)는 *NP-hard* 이다.

(증명)문제 U가 *NP-hard* 임을 증명하려면 임의의 *NP-complete* 문제가 문제 U로 귀납(reduce)됨을 보이면 된다. 여기서는 잘 알려진 *NP-complete*문제인 최장경로문제(the longest path problem)가 문제 U로 귀납됨을 보이

자. 즉, 링크 (i,j)의 거리계수가 $h_{ij}>0$로 주어진 경우 노드 s,t간의 최장경로문제를 고려하자.

$max\{\Sigma_{ij\in P} h_{ij} \mid P: s,t\text{-단순경로}\}$

이때, $r_{ij} = exp(h_{ij})$ 이라고 하면, 위 문제의 목적함수는 다음과 같다.

$\Sigma_{ij\in P} h_{ij} = \Sigma_{ij\in P} log(r_{ij}) = -log(\Pi_{ij\in P} r_{ij}^{-1})$

따라서, 위 문제는 다음과 동치가 된다.

$min\{\Pi_{ij\in P} r_{ij}^{-1} \mid P: s,t\text{-단순경로}\}$

그런데, 이 문제는 링크 (i,j)의 비용계수, 흐름이득계수가 각각 $c_{ij}=0$, $r_{ij}>1$이고, $d_s=g=1$일 경우, 단순 s,t-경로를 따라 d_t의 조건부상한의 최소를 구하는 문제이다. 즉, 임의의 최장경로문제가 문제 U로 귀납되므로 문제 U는 *NP-hard* 이다. ■

3.2.2 잠재가의 상한

행로 W가 닫힌 행로(t,t-행로)이고, 행로의 흐름이득계수 $r(W)$가 1보다 크다면, (식1)은 잠재가 d_t의 상한 $u_t(W)$을 제공하고 이는 유효부등식을 구성한다. 즉,

$$-d_t + r(W)\,d_t \le c(W) \Rightarrow d_t \le c(W)/\{r(W)-1\} \equiv u_{t(W)}$$

모든 t, t-행로를 따라 얻을 수 있는 최소의 잠재가상한 u_t*은 최적잠재가의 상한이 되며, 특히 노드 t가 최적유사포리스트의 환에 속하는 노드라면 최적잠재가와 일치한다.

[정리 5] 쌍대해가 존재할 경우 다음의 관계가 성립한다.

1) $d_t^* \le u_t^* \equiv min\{u_t(W): \forall W: t,t\text{-행로}, r(W)>1\}$

2) 노드 t가 최적유사포리스트의 환에 속하면 $d_t^* = u_t^*$ 이다.

(증명)

1) 부등식 ($d_t \le u_t^*$)은 유효부등식 ($d_t \le u_t(W)$)의 교집합이므로 쌍대해집합에 유효한 부등식이다. 따라서 가능해 d_t^*에 대해서도 이 부등식은 성립한다. 즉, $d_t^* \le u_t^*$

2) 노드 t가 속하는 환 W^*은 흐름이득계수가 1보다 큰 t,t-행로의 범주에 속한다. 따라서,

$d_t^* = u_t(W^*) \ge min\{u_t(W) : \forall W\} = u_t^*$

그리고, 1)에서 $d_t^* \le u_t^*$이므로 $d_t^* = u_t^*$이다. ■

또한 쌍대부등식체계와 이의 자료크기 $L = size(A,c)$와 관련하여 다음 유효부등식을 얻을 수 있다.

[정리 6] 쌍대최적해(최적잠재가)가 존재하면 $d_j^* \le 2^L, \forall j\in V$ ■

4. 해법

4.1 일반화된 라벨수정법

최단경로문제의 해법은 라벨수정법을 바탕으로 한 다양한 형태의 변종이 있지만, 그 해법들을 공통적으로 지탱하고 있는 것은 삼각연산이다.

$$if\ d_j > d_i + c_{ij} \quad then \quad d_j \leftarrow d_i + c_{ij}$$

삼각연산은 링크 (i,j)에 대해 잠재가 d_i, d_j가 주어졌을 때, 만약 $d_j > d_i + c_{ij}$이면 현재 노드 j의 잠재가를 d_i+c_{ij}로 수정한다. 즉, 삼각연산의 역할은 주어진 잠재가에 대해 쌍대부

등식 $-d_i + d_j \le c_{ij}$을 만족하지 않는 링크에 대해서 노드 j의 잠재가를 감소시켜 이를 만족시키는 것이다.

이러한 삼각연산의 논리는 최단흐름생성경로문제에도 적용될 수 있다.

$$\text{if } d_j > (d_i + c_{ij})/r_{ij} \quad \text{then} \quad d_j \leftarrow (d_i + c_{ij})/r_{ij}$$

앞장의 알고리듬 *walk* 의 잠재가 개선 방법은 삼각연산의 원리가 적용된 것으로 알고리듬의 입력요소인 초기잠재가를 $a=(2^L,\cdots,2^L)$로 두고 알고리듬을 진행하면 잠재가가 감소하여 최적잠재가에 접근하게 된다. 그러나 이러한 방식으로 최적잠재가를 구하는 것은 몇 가지 문제점을 야기하므로 이를 살펴보고 해결하는 방안을 고려하자.

첫째, 잠재가가 환을 무한 순환하면서 개선되는 경우가 발생한다. 최단경로문제의 경우 이러한 현상이 발생하면 음의 환이 존재하여 쌍대비가능, 즉 원문제가 무한해를 갖는다고 판별하고 알고리듬을 종료하게 된다. 그러나, 최단흐름생성경로문제의 경우 쌍대가능인 경우에도 이러한 현상이 발생한다.

편의상 N개의 노드로 구성된 환에서 이를 살펴보자. 잠재가의 수정은 $1 \to 2 \to \cdots \to N \to 1$의 순으로 진행된다고 가정하고, 노드 1을 방문하는 횟수k에 따라 단계 k를 설정하자. 그러면 단계 k와 단계 $k+1$에서 노드 1의 잠재가는 다음 관계를 갖는다.

$$d^{(k+1)} = r(W)^{-1} d^{(k)} + r(W)^{-1} c(W), \ \forall k \ge 0$$

윗식에서 $d^{(k)}$를 구하면 다음과 같다.

```
Algorithm generalized label correcting(a)

   for k=1 to n do d[k]←a_k; p[k] := k ;
   NOW ←V; NEXT ← φ;
 L: while NOW ≠ φ do
   begin
     i ← pick( NOW );
     Ni ← out neighbor of i:
     while Ni ≠ φ do
     begin
        j ← pick( Ni );
        Δ ←( c_ij + d[i] )/r_ij
        if d[j] > Δ then
        begin
           find i-flow generating path P
           if j ∈ P then
        identify new cycle H with (i,j)
               case of ρ(H)
                   >1 : update d[H]
                        p[j] := i
                        NEXT ← NEXT ∪ H
                   ≤1 : stop (unbounded)
           else
           d[j] ← Δ;
           p[j] ← i;
           NEXT ← NEXT ∪ {j}
        end
      end
   end
   if NEXT = φ then stop
    else replace NOW by NEXT and NEXT ← φ and goto L
```

그림 3. 알고리듬 generalized label correcting

$$d^{(k)} = r(W)^{-k} d^{(0)} + c(W)(1-r(W)^{-k})/(1-r(W)^{-1}),$$
단 $r(W) \ne 1$

환의 이득계수 $r(W)$가 1보다 작다면 단계가 증가됨으로서 음의 무한대로 발산하고, 1

보다 크다면 환에서의 최적잠재가 $d^* = c(W)/\{r(W)-1\}$로 수렴한다. 그러나 정확한(exact) 최적잠재가에는 도달하기 위해서는 무한단계를 거쳐야 하므로 환에서의 잠재가 수정은 계속 반복되게 된다. 이렇게 환을 계속 순환하게 되는 현상을 방지하기 위해서는 잠재가를 수정할 때마다 환의 발생여부를 조사하여 환의 잠재가는 따로 계산해주어야 한다. 알고리듬 진행시 링크 (i,j)가 고려대상이 라고 하자. 이 때 환이 생성되는 유일한 경우는 노드 j가 i-흐름생성경로 상에 존재하는 경우이다. i-흐름생성경로 는 전위자를 통해 구해진 노드의 순열 I로 판별할 수 있다. 이 때 순열 I는 다음과 같다 하고, 노드 i_N을 환과 경로가 만나는 노드라 하자.

$$i\text{-}i_1\text{-}i_2\text{-}\cdots\text{-}i_N\text{-}i_{N+1}\text{-}i_{N+2}\text{-}\cdots\text{-}i_{N+M}$$

새로 생성되는 환은 노드 j가 i_k일 경우, i에서 i_k 까지의 부분순열로 결정되는 j,i-경로와 링크 (i,j)로 구성된다. 이 때, $k<N$이면 새로운 환이 추가생성되고, $k \geq N$이면 i-흐름경로를 구성하던 기존의 환이 새로운 환으로 대체된다. 만약 알고리듬 도중에 새로 만들어지는 환이 흐름생성환(흐름이득계수> 1)일 경우에는 알고리듬을 계속진행하고, 흐름흡수환(흐름이득계수가<1)일 경우에는 쌍대비가능문제로 판별하고 알고리듬을 종료한다.

둘째, 링크 (i,j)에서 노드 i의 잠재가가 개선되면 노드 j의 잠재가가 개선된다. 역으로, 노드 j의 잠재가개선은 노드 i의 잠재가개선을 전제로 한다. 따라서, 모든 링크를 검색하는 번잡함을 피하기 위하여 잠재가가 개선된 노드의 전방링크집합만을 다음단계에서 검색하도록 한다. 이러한 상황을 고려하여 구성된 해법이 알고리듬 *generalized label correcting*이다.

[정리 7] 알고리듬 *generalized label correcting*은 다음 특성을 갖는다.

1) 알고리듬진행도중에 생긴 환이 흐름흡수환이면 문제는 무한가능해를 갖는다(즉, 쌍대비가능문제).

2) 초기잠재가를 $a = u^*$로 두면, 알고리듬에서 환의 생성여부를 판별하는 루틴은 불필요하고, $O(nm)$의 계산복잡도로 최적잠재가 d^*를 구한다.

(증명)

1) 편의상 환이 발생하는 상황은 앞의 논의와 같다고 하자.

링크 (i,j)에서 삼각연산을 하기 직전의 노드 j의 잠재가를 a_j라 하자. 이는 i-흐름생성경로의 부분경로인 j-흐름생성경로를 따라 구해진 것이므로 $d_j \leq a_j$는 유효부등식이다. 그리고, 링크 (i,j)를 통해 새로 만들어지는 환 W에서 다음의 유효부등식을 얻을 수 있다.

$$-d_j + r(W)\, d_j \leq c(W) \;\Rightarrow\; d_j \geq c(W)/\{r(W)-1\}$$

그런데 링크 (i,j)에서 삼각연산이 이뤄지기 위해서는 두 유효부등식의 우변상수간에는 다음관계가 성립한다.

$$a_j > r(W)^{-1} a_j + r(W)^{-1} c(W) \Rightarrow a_j < c(W)/\{r(W)-1\}$$

그러므로 두 유효부등식은 동시에 만족할 수 없다. 즉, 쌍대비가능이다.

2) u_j^*는 환을 포함한 모든 j,j-행로 상에서

얻어진 최소잠재가이므로 알고리듬진행도중에 환을 순환하면서 잠재가가 감소할 수 없다. 환이 생기지 않는 경우는 최단경로문제에 라벨수정법을 적용할 때와 동일한 행태로 알고리듬이 진행되므로 그 계산복잡도는 O (mn) 이다. ■

알고리듬 *generalized label correcting* 에 사용된 함수 *pick*(·)는 다음과 같다. 즉, $a \leftarrow$ *pick*(*LIST*)는 *LIST*의 원소중 하나를 변수 a에 지정하고, 그 원소는 *LIST*에서 삭제함을 의미한다.

4.2 강성다항식 해법

만약 모든 노드 t에 대해서, t,t-행로상의 잠재가의 최소상한을 강성다항식 시간에 얻을 수 있다면, [정리 7]에 의해 최단흐름생성경로문제는 강성다항식시간에 해결 가능함을 알 수 있다. 따라서, 이 절에서는 t,t-행로상의 잠재가의 최소상한 u_t^*을 구하는 방법을 고찰한다.

앞 장에서 논의된 잠재가의 상한 $d_t(W)$는 조건부상한 $\mu_t(g;W)$의 고정점(fixed point)-다시 말해 $\mu_t(g;W)=g$의 해-임을 알 수 있는데, 잠재가의 최소상한 u_t^*는 다음의 함수 $\gamma_t(g)$와 유사한 관계를 갖는다.

$$\gamma_t(g)=min\{\mu_t(g;W):\ \forall W:\ t\text{-행로}\}$$

먼저 $\gamma_t(g)$의 특징을 살펴보자. 대략적인 개형은 그림 4에 나타나 있다.

[정리 8] 함수 $\gamma_t(g)$는 오목구간선형(concave piecewise linear) 함수이고, 증가함수이다.

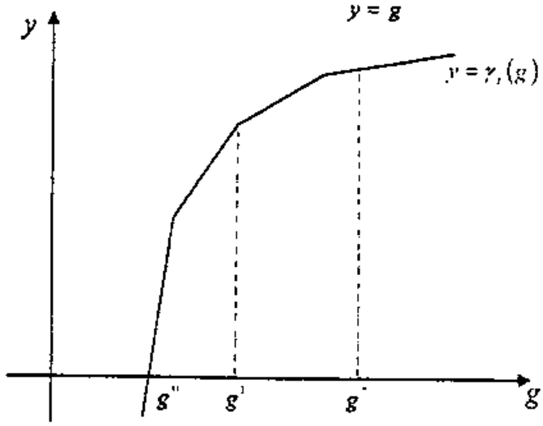

그림 4. 함수 γ_t(g)의 개형

(증명) 함수 $\mu_t(g;W)$는 선형함수이고 그 기울기는 t,t-행로의 흐름이득계수(>0)의 역수이므로 증가함수이다. 따라서, $\gamma_t(g)$는 증가선형함수군의 최소함수이므로 증가하는 오목구간선형함수이다. ■

잠재가의 최소상한 u_t^*과 조건부잠재가의 최소는 다음의 관계를 갖는다.

$$\begin{aligned} u_t^* &= min\{u_t(W):\ \forall W:\ t\text{-행로},\ r(W)>1\} \\ &= min\{[g:\ g=\mu_t(g;W)]:\ \forall W:\ t\text{-행로},\ r(W)>1\} \\ &= [g:\ g=min\{\mu_t(g;W):\ \forall W:\ t\text{-행로},\ r(W)>1\}] \\ &= \{g:\ g=\gamma_t(g),\ \partial^+\gamma_t(g)>1\} \end{aligned}$$

즉, 잠재가의 최소상한을 구하는 문제는 함수 $\gamma_t(g)$의 고정점을 찾는 문제로 환원된다.

[정리 8]에서 나타난 $\gamma_t(g)$의 특성 때문에, $\gamma_t(g)$의 고정점은 많아야 2개이다. 만약 고정점이 존재하지 않는다면 쌍대비가능이고, 이 경우 모든 g에 대해 $\gamma_t(g)<g$임을 의미한다. $\gamma_t(g)$와 $\gamma_t(g)$의 우측기울기 $\partial^+\gamma_{t(g)}$에 따라 실선분$(-\infty<g<\infty)$은 다음 상호배타적인 영역으로 구별된다.

영역 Ⅰ : $g \le g^0 \Leftrightarrow \gamma_t(g) \le g,\ \partial^+\gamma_t(g) > 1$

영역 Ⅱ : $g^0 \le g \le g^1 \Leftrightarrow \gamma_t(g) \ge g,\ \partial^+\gamma_t(g) > 1$

영역 Ⅲ : $g \le g \le g^* \Leftrightarrow \gamma_t(g) \ge g,\ \partial^+\gamma_t(g) < 1$

영역 Ⅳ : $g^* \le g \Leftrightarrow \gamma_t(g) \le g,\ \partial^+\gamma_t(g) < 1$

그런데 $\gamma_t(g)$는 최악의 경우 지수 함수 개수의 선형함수의 최소로 정의 되므로, 이를 명시적으로 구하는 것은 어렵다. 그러나, 주어진 점 g에 대해 $\gamma_t(g)$와 우측기울기 $\partial^+\gamma_t(g)$는 알고리듬 *eval(t,g)*를 통해 구해진다.

알고리듬 *eval*은 알고리듬 *walk*과 동일한 구조인데, $\gamma_t(g)$와 $\partial^+\gamma_t(g)$를 구하기 위하여 산각연산부분을 조금 수정하였다. 그리고 $\gamma_t(g)$는 변수 $d_t^{[n]}$에, $\partial^+\gamma_t(g)$는 변수 α_t에 저장된다.

이제 주어진 g에 대하여 $\gamma_t(g)$, $\partial^+\gamma_t(g)$를 알고리듬 *eval*에 의해 $O(nm)$에 구할 수 있으므로, 이를 이용하여 잠재가의 최소상한 u_t^* (즉, 그림 4에서 g*)를 구하는 방법을 고찰하자.

```
Algorithm eval(t,g)

d_t^[0] ← g;  α_t ← 1; β_t ← 0;
for k ≠ t do d_i^[0] ← ∞,  α_i ← 0, β_i ← 0
for k=1 to n do
    for (i,j) ∈ E do
        δ ← (d_i^[k-1] + c_ij)/r_ij
        if ( δ < d_j^[k-1]) then
            d_j^[k-1] ← δ
            α_j ← α_i/r_ij
            β_j ← ( β_i + c_ij)/r_ij
    end
end
```

그림 5. 알고리듬 *eval(t,g)*

먼저, g*에 대한 추정치 v를 가지고 시작한다. g*는 영역 Ⅲ과 영역 Ⅳ의 경계에 존재하므로 V가 영역 Ⅰ, Ⅱ, Ⅲ에 속하면 새로운 추정치 v′(>v)를 사용하고, 영역 Ⅳ에 속하는 경우 새로운 추정치 v′(<v)를 사용한다. 이러한 방식으로 추정치를 개선하면 추정치 v′는 g*(존재할 경우)에 수렴해간다.

따라서, 잠재가의 최소상한을 구하는 데 걸리는 시간은 (추정치의 개선회수)×(알고리듬 *eval*의 계산 복잡도)가 된다.

즉 g*의 존재 범위($-2^L < g^* < 2^L$)을 설정하고 이진 탐색법을 적용한다면 O(nmL)의 시간이 걸린다. 그런데 Meggido의 연구결과에 의하면 추정치의 개선회수는 알고리듬 *eval*에 사용되는 비교연산의 회수면 충분하다고 알려져 있다.

이는 매개변수 탐색법으로 불리우는데 그 개념은 다음과 같다.

알고리듬 *eval*은 함수 $\gamma_t(g)$와 $\partial^+\gamma_t(g)$를 계산하기 위한 것인데, 만약 매개변수 G를 고정된 수치가 아닌 하나의 문자로 가정하고 알고리듬 *eval*을 실행하는 경우를 상정하자.

*eval*에 사용되는 연산은 사칙연산과 비교연산이다. 사칙연산은 d_j의 계산에서 사용된다. 그런데 d_j는 g에 대해 선형함수이므로 그 계수의 순서쌍 (α_j , β_j)의 개별적 사칙연산으로 환원되어, 기호대수적(symbolically)으로 수행될 수 있다.

그런데 문제는 비교연산과정에서 발생한다. 즉, 비교연산은 $d_j^{[k-1]}$과 D의 대소를 판별하는 조건문에서 사용되는데, $d_j^{[k-1]}$과 D가 g에 관한 일차함수이므로 일의적으로 대소를

결정할 수 없다. 즉, 두 일차함수 $f_i(g) = \alpha_i g + \beta_i$ (i = 1, 2)의 대소는 $\alpha_1 \neq \alpha_2$ 일 경우, 임계점 $g_c=(\beta_2 - \beta_1)/(\alpha_1 - \alpha_2)$를 전후로 뒤바뀌게 된다. 즉, 임계점 g_c가 속하는 영역의 판별은 알고리듬 EVAL에 의해 가능하고, 우리가 관심을 가지는 영역은 g*가 속하는 영역이므로, 매개변수 g의 범위를 $\{g|g \geq g_c\}$ 혹은 $\{g|g \leq g_c\}$를 한정할 수 있다.

즉, 매개변수 g의 존재영역을 한정해 줌으로써 두 선형함수 $f_1(g)$, $f_2(g)$의 대소비교는 명시적으로 가능해진다. 이러한 방식으로 알고리듬 *eval*이 진행되면, g*의 존재영역이 감소하고 결국 g*를 찾을 수 있게 된다.

이러한 개념하에서 알고리듬 *parametric-search(t)*은 u_t*을 강성다항식 시간에 찾을 수 있게 해 준다.

알고리듬 *parametric-search(t)*에 사용된 함수 *cross-point*(·)와 *slope-min*(·), *slope-max*(·)는 두 선형함수 $f_i(g)=a_i g+b_i$, i=1,2를 입력으로 하며 그 출력은 다음과 같다.

cross-point(f_1,f_2)는 f_1과 f_2의 교차점을 출력하고, *slope-min*(f_1,f_2), slope-max(f_1,f_2)는 각각 f_1과 f_2중 기울기 α_i가 작은(큰) 함수 f_i를 출력한다.

[정리 9] 최단흐름 생성경로문제는 $O(n^3m^2)$의 시간에 해결 가능하다.

(증명) 노드 t에 대해 u_t*을 알고리듬 parametric-search(t)에 의해 구해진다. 이때 계산복잡도는 (추정개선회수)×*eval*의 계산 복잡도이다. 그런데, 추정개선회수는 *eval*의 비교연산회수와 동일하므로 O(nm×nm)이다. 따라서 모든 노드에 대해 u_t*은 $O(n^3m^2)$의 시간에 구해지고, 이를 통해 d_t*는 O(nm)에 구해지므로 전체계산복잡도는 $O(n^3m^2 + nm) = O(n^3m^2)$이다. ■

```
algorithm parametric-search(t)

[Lo,Up]←[-2^L,2^L]
v<- Up
eval(i,v)
case of
  α>1 : Up ←h ; v ← Up
        eval(i, Up)
        if (v=r) then return (u_t*←r)
  α<1 : return (u_t*←∞)
  α=1 : if β<0, then return (infeasible)
for ∀k≠t {d_k^[0]←∞} ; d_t^[0] ← g
 for i←1 to n
    d_i^new(g) ← (d_j^[k-1](g) + c_ij)/r_ij
    v← cross-point(d_j^[k-1](g), d_i^new(g))
    if (v∈[Lo,Up]) then
     eval(i,v);
    case of
     α>1, r<v: Lo ← h
     α≥1, r≥v: Lo ← v
     α<1, r≥v: Lo ← r, Up ← min(Up,h)
     α<1, r>v: Up ← h
      if ( α=1, r<v) or (Up<Lo) then end(infeasible)
      if (Up is updated) then v←Up, eval(i,v);
              if (v=r) then end
   if (v ≤ L)
     then d_i^[k] ← slope_min(d_j^[k-1](g), d_i^new(g))
   if (v ≥ U)
     then d_i^[k] ← slope_max(d_j^[k-1](g), d_i^new(g))
 end
end
```

그림 6. 알고리듬 *parametric-search(t)*

V. 결론

본 연구에서는 일반화된 네트워크흐름의 잠재가측면이 부각된 가장 간단한 형태의 문제로 최단흐름생성경로문제를 제안하고, 이의 정당성 및 문제의 구조를 고찰하였다. 또한 이 문제가 강성다항식계산복잡도로 해결됨을 보였다. 최단흐름생성경로문제의 강성다항식해법은 기존의 일반화된 최대흐름문제의 다항식해법[8]과 어울려 일반화된 네트워크흐름의 이론체계를 정립하는 좋은 출발점이 될 것이라 기대된다.

참고문헌

[1] ADLER, I. and COSARES, S., 1991, Strongly polynomial algorithm for a special class of linear programs, *Operations Research* 39, pp.959-995

[2] AHUJA, R.K., MAGNANTI, T.L., and ORLIN, J.B., 1993, *Network flows,* Prentice Hall

[3] CUNNINGHAM, W.H., 1979, Theoretical properties of the network simplex method, *Mathematics of Operations Research 4*, pp. 196-208

[4] ELAM, J., GLOVER, F. and KLINGMAN, D., 1979, A strongly convergent primal simplex algorithm for generalized networks, *Mathematics of Operations Research 4*, pp. 39-59

[5] ERVOLINA, T.R. and McCORMICK, S. T., 1990, Two strongly polynomial cut cancelling algorithms for minimum cost network flow, Technical Report, *Faculty of Commerce and Business Administration*, University of British Columbia, Vancouver, Canada

[6] GALIL, Z. and TARDOS, E., 1986, An $O(n^2(m+nlogn)logn)$ min-cost flow algorithm, *Journal of ACM 35*, pp.374-386

[7] GLOVER, F., KLINGMAN, D., and PHILLIPS, N., 1985, A new polynomially bounded shortest path algorithm, *Operations Research 33*, pp.65-73

[8] GOLDBERG, A.V., PLOTKIN, S.A., and TARDOS, E., 1991, Combinatorial algorithms for the generalized circulation problem, *Mathematics of Operations Research 16*, pp.351-381

[9] GOLDFARB, D., HAO, J. and KAI, S., 1990, Efficient shortest path simplex algorithms, *Operations Research 38*, pp.624-628

[10] GOLDFARB, D., HAO, J. and KAI, S., 1990, Anti-stalling pivot rules for the network simplex algorithm, *Networks 20*, pp.624-628

[11] JENSEN, P.A., and BHAUMIK, B., 1977, A flow augmentation approach to the network with gains minimum cost flow problem, *Management Science 23*, pp. 631-643

[12] JEWELL, W.S., 1962, Optimal flow through networks with gains, *Operations Research 10*, pp.476-499

[13] KENNINGTON, J.L., and HELGASON, R.V., 1980, *Algorithms for Network Programming*, Wiley-Interscience, New York

[14] KOEHLER, G.J., WHINSTON, A.B., and WRIGHT, G.P., 1975, *Optimization over Leontief Substitution Systems*, North-Holland, Amsterdam

[15] LAWLER, E.L., 1976, *Combionatorial Optimization*, Holt, Rinehart and Winston

[16] MALEK-ZAVAREI, M. and AGGARWAL, J.K., 1972, Optimal Flows in Networks with gains and cost, *Networks, Vol.1, No. 4* pp.355-365

[17] MEGGIDO, N., 1979, Combinatorial optimization with rational objective functions, *Mathematics of Operations Research 4*, pp.414-424

[18] MEGGIDO, N., 1983, Towards a genuinely polynomial algorithm for linear programming, *SIAM J.COMPUT Vol.12*, No. 2. pp.347-353

[19] ORLIN, J.B., 1985, On the simplex algorithms for networks and generalized networks, *Mathematical Programming Study 24*, pp.166-178

[20] ORLIN, J.B., 1993, A faster strongly polynomial minimum cost flow algorithm, *Operations Research Vol.41, No.2*, pp. 338-350

[21] TARDOS, E., 1985, A strongly polynomial minimum cost circulation algorithm, *Combinatorica 5*, pp.247-255

[22] VEINOTT, A.F., 1968, Extreme points of Leontief Substitution Systems, Linear Alg., *Its Appl.1*, pp.181-194

96년 12월 최초 접수, 97년 6월 최종 수정