배정 문제에 대한 다중 부가경로 해법

An efficient multi-augmenting algorithm for the assignment problem

정의석*, 강자영**, 박성수***

* (quasitree@gmail.com)
** 한국과학기술원 산업공학과 (kangdding@kaist.ac.kr)
*** 한국과학기술원 산업공학과 (sspark@kaist.ac.kr)

Abstract

In this paper, the assignment problem is considered. We propose an approach based on the solution of a sequence of shortest path sub-problem. We extend the cost reduction method, which is used for finding initial assignment, to solve these sub-problems. The use of the extended reduction method makes it possible to devise an efficient multi-augmenting algorithm.

1. 서론

배정 문제 assignment problem 는 이분할 그래프 bipartite graph $G=(S\cup T,E)$와 에지 edge (i,j)의 비용 c_{ij}이 주어졌을 때 최소비용의 (최대) 매칭 (max cardinality) matching 을 찾는 문제이다. 이 때 에지 $(i,j)\in A$는 $i\in S, j\in T$ 형태이며, 에지 (i,j)의 비용을 c_{ij}라 할 때, 행렬 $\boldsymbol{c}=(c_{ij})_{i\in S, j\in T}$를 비용행렬이라 한다. 매칭은 소속된 에지가 인접하지 않는 부분그래프로 정의된다. 이 문제에는 헝가리안 알고리듬을 필두도 다양한 알고리듬들이 연구되었는데, 특히, 헝가리안 알고리듬은 후에 이론적 틀이 체계화되어 배정 문제에 대한 최단부가경로 shortest augmenting path 알고리듬으로 재탄생하였다[Edmonds and Karp, 1972]. 이 이론적 틀은 선형계획문제에 연역적으로 확대되어 원쌍대 primal-dual 알고리듬의 근간이 되었고, 다양한 조합최적화 문제의 해법의 이론적 근거로 활용되며 [Papadimitriou and Steiglitz, 1982], 그 이론적 유연성에 힘입어 세부적인 적용방법에 따라 동일한 문제에 대해서도 다양한 형태의 알고리듬이 도출된다 [Ahuja, Magnanti and Orlin, 1993]. 특히, 이의 최소비용 흐름 min cost flow 문제에 적용된 형태가 부가경로 augmenting path 법으로 이는 세부적으로 단일 부가경로법과 다중 부가경로법으로 나뉘는데, 전자는 최단경로 shortest path 문제 하나당 하나의 부가경로를, 후자는 최대 흐름 max flow 문제 하나당 다수의 부가경로를 얻지만, 전자 보다 후자의 경우가 국소적인 계산복잡도 측면에선 불리한 측면이 있다. 하지만, 최소비용 흐름문제의 특화된 형태인 배정 문제는 최단경로문제만으로도 다중 부가경로를 구할 수 있음이 알려져 있다[Carraresi and Sodini, 1986]. 즉, 최단경로트리 shortest path tree 로도 다중 부가경로에 대한 충분한 정보가 얻어지는 것이다. 배정 문제의 최단부가경로법은 초기해를 구하는 간단한 비용행렬 할인 cost matrix reduction 방법이 존재하는데, 놀랍게도 이러한 간단한 방법이 랜덤 그래프 random graph 에 대해 평균적으로 최대매칭의 약 80%를 초기해로 제공함이 알려져있다[Nawijn and Dorhout, 1989].

본 논문은 이러한 비용행렬할인법과 다중 최단부가경로법과의 밀접한 관계에 착안하여, 비용행렬할인법의 특징을 효과적으로 반영한 (즉, 알고리듬의 각 단계에서 도출된 최단경로 트리의 정보를 더욱 효과적으로 이용하는) 새로운 다중 부가경로법을 제시하는 것을 목적으로 한다.

2. 비용행렬 할인법

배정 문제의 특성 중 하나가 비용행렬의 임의의 행이나 열에 임의의 수를 일괄 가감하여도 최적 매칭이 변하지 않는다는 점이다. 이러한 특성이 비용행렬할인을 통한 초기해 도출방법을 가능하게 한 것이고, 결국 헝가리안 알고리듬의 탄생에 기여하게 된 것이다[Schrijver, 2003]. 통상 비용행렬할인은 다음의 두 단계로 진행된다. 비용행렬의 각 행의 최소값을 각 행에서 제하고 난 다음(단계 1), 앞서 할인된 행렬의 각 열의 최소값을 제한 행렬(단계 2)의 0인 원소에 해당하는 부분그래프에 대해서 최대매칭을 찾으면 상보여유정리를 만족하는 초기해가 된다. 일반적인 이분할 그래프에서 최대매칭을 찾는 알고리듬은 $O(\sqrt{n}m)$의 계산복잡도를 갖지만, 만약 최대매칭을 찾는 부분그래프를 각 행과 열에서 0인 에지를 하나씩만 취사선택한 그래프로 정의하면, 최대매칭은 다음 알고리듬 CMR에 의해 $O(m)$의 계산복잡도로 구할 수 있다. 알고리듬 CMR에 사용된 함수 **get** 은 입력 집합의 임의의 원소 하나를 넘겨주는 역할을 한다. 여기서 '임의'란 말은 확률적 의미가 아닌 아무 원소나 넘겨줘도 무

방하다는 의미이다.

algorithm CMR
1 $mate(i) \leftarrow \varnothing,\ v_i \leftarrow 0 \quad \forall i \in S \cup T$
2 $v_j \leftarrow$ **min**$\{c_{ij} : i \in S\} \quad \forall j \in T$
3 $\sigma_j \leftarrow$ **get**$\{i \in S : c_{ij} - v_j = 0\} \quad \forall j \in T$
4 $S_\sigma \leftarrow \{\sigma_j : j \in T\}$
5 $N_a(i) \leftarrow \{j \in T : \sigma_j = i\} \quad \forall i \in S_\sigma$
6 $v_i \leftarrow$ **min**$\{c_{ij} - v_j\} \quad \forall i \in S \backslash S_\sigma$
7 $\tau_i \leftarrow$ **get**$\{j \in T : c_{ij} - v_i - v_j = 0\} \quad \forall i \in S \backslash S_\sigma$
8 $T_\tau \leftarrow \{\tau_i : i \in S \backslash S_\sigma\}$
9 $N_\tau(j) \leftarrow \{i \in S \backslash S_\sigma : \tau_i = j\} \quad \forall j \in T_\tau$
10 $S_\sigma{}^* \leftarrow \{i \in S_\sigma : N_\sigma(i) \backslash T_\tau \neq \varnothing\}$
11 $mate(j) \leftarrow$ **get**$(N_\tau(j)) \quad \forall j \in T_\tau$
12 $mate(i) \leftarrow$ **get**$(N_\sigma(i) \backslash T_\tau) \quad \forall i \in S_\sigma{}^*$

알고리듬 CMR에서 라인 1은 각 변수의 초기화, 라인 2-5는 비용행렬할인의 단계 1, 라인 6-9는 비용행렬할인의 단계 2에 해당하고, 라인 10-12는 초기매칭을 구하는 부분이다. 단계 1과 단계 2에서 선택된 에지의 집합을 각각 $E_\sigma = \{(\sigma_j, j) : j \in T\}$, $E_\tau = \{(i, \tau_i) : i \in S \backslash S_\sigma\}$라 하고, 이로 구성되는 그래프를 $G_\sigma = (S \cup T, E_\sigma)$, $G_\tau = (S \cup T, E_\tau)$ 그리고, 두 에지집합의 합으로 구성된 그래프를 $G_\mu = (S \cup T, E_\sigma \cup E_\tau)$라 하자.
그래프 G_σ와 G_τ는 항상 몇 개의 트리 tree 로 이뤄진 포리스트 forest 구조이고, 에지집합 E_σ와 E_τ는 상호 배타적이다. 즉, $E_\sigma \cap E_\tau = \varnothing$.
알고리듬 CMR의 계산복잡도는 라인 3, 7이 $O(m)$이고, 라인 5, 9 는 G_σ, G_τ의 연결 컴포넌트 connected component 를 구하는 작업으로, 깊이 우선 탐색 Depth First Search 를 이용해서 $O(n)$, 그리고 나머지 라인들은 자명하게 $O(n)$이므로 전체 계산복잡도는 $O(m)$이다.

앞으로 논의 될 다중 부가경로와 관련해서 그래프 G_σ, G_μ에서의 최대 매칭은 중요한 의미가 있다.
그래프 $G_\mu{}^*$를 E_σ 중에서 T_τ와 인접한 에지를 제거한 그래프 즉, $E_\sigma{}^* = \{(\sigma_j, j) : j \in T \backslash T_\tau\}$라 할 때 $G_\mu{}^* = (S \cup T, E_{\sigma^*} \cup E_\tau)$라 하자. 그러면, 그래프 G_σ, G_μ가 주어진 상태에서 최대 매칭은 각각 G_σ, $G_\mu{}^*$의 고립된 노드 isolated node 를 제외한 포리스트의 개수와 같고 이는 깊이 우선 탐색으로 계산복잡도 $O(n)$으로 해결 가능하다.

정리1 (1) 그래프 G_σ의 최대 매칭은 G_σ의 고립된 노드를 제외한 포리스트의 개수와 같고, 그 개수는 $|S_\sigma|$ 이다.
(2) 그래프 G_μ의 최대 매칭은 G_μ의 고립된 노드를 제외한 포리스트의 개수와 같고, $|S_\sigma| + |T_\tau|$이다.
(즉, 알고리듬 CMR에서 산출된 매칭은 G_μ의 최대 매칭이다.)
(증명) König-Egervary 정리[Lawler, 1976](이분할 그래프에서 최대 매칭 = 최소 커버링 min covering)를 이용한다. (1) 매칭은 각 포리스트에서 임의의 에지를 취한 것이라 하자. 그 개수는 $|S_\sigma|$이다. 그런데 S_σ가 G_σ의 모든 에지를 커버하므로 그 개수가 매칭과 같으므로 언급된 매칭과 커버링은 최적이다.
(2) 매칭은 알고리듬 CMR의 얻어진 것을 고려하자. 이는 $G_\mu{}^*$의 고립된 노드를 제외한 포리스트에서 취해진 것이다. 그리고, 그 개수는 $|S_\sigma{}^*| + |T_\tau|$이다. 그런데, 노드집합 $S_\sigma{}^* \cup T_\tau$이 모든 에지를 커버하므로 커버링이 되고, 언급된 매칭과 커버링의 개수가 같으므로 최적이다.□

2. 비용행렬 할인법

배정 문제는 유방향 그래프 $G = (S \cup T, A)$에서 최소비용흐름 문제로 표현되며, 다음의 선형계획문제 **(P)**로 기술되고, 이의 쌍대문제는 다음의 **(D)**와 같다. 여기서 S, T는 각각 출발, 도착 노드집합이며, 편의상 $S = \{1,...,s\}$, $T = \{s+1,...,s+t\}$ 라 한다. 모든 아크 arc $(i,j) \in A$ 는 $i \in S, j \in T$ 형태이며, 아크 (i,j)의 비용을 c_{ij}라 할 때, 행렬 $\boldsymbol{c} = (c_{ij})_{i \in S, j \in T}$를 비용행렬이라 한다.

$$\textbf{(P)}\ \textbf{min} \sum_{(i,j) \in A} c_{ij} x_{ij}$$
$$\textbf{s.t.} \sum_{\{j \in T : (i,j) \in A\}} x_{ij} = 1 \quad \forall i \in S$$
$$\sum_{\{i \in S : (i,j) \in A\}} x_{ij} = 1 \quad \forall j \in T$$
$$x_{ij} \geq 0 \quad \forall (i,j) \in A$$

$$\textbf{(D)}\ \textbf{max} \sum_{i \in S} v_i + \sum_{j \in T} v_j$$
$$\textbf{s.t.}\ v_i + v_j \leq c_{ij} \quad \forall (i,j) \in A$$

최단 부가경로법은 주어진 매칭에 대해 부가 네트워크 augmented network 을 구성한 뒤 부가 네트워크에서 최단경로를 구한 다음 최단경로 트리 상의 아크만을 이용하여 부가경로를 구하고 이를 통해 새로운 매칭을 만들고, 최단경로 거리 distance를 통해 쌍대해를 개선하는 것을 최대매칭이 찾아질 때까지 반복한다. 전체적인 구성은 다음 알고리듬 SAP 과 같다.

algorithm SAP
1 Initialize
2 **while**($M \neq$ maximum)
3 construct augmented network G_M with $\bar{c}_{ij}$
4 $(d, T) \leftarrow$ ShortestPath(G_M)
6 updated dual variable $v_i \ \forall i \in S \cup T$
7 $P \leftarrow$ set of augmenting path from T
8 $M \leftarrow M \oplus P \ \forall P \in P$
9 **end**

자세한 논의를 위하여 다음 몇 가지를 정의하자. 이분할 그래프 $G = (S \cup T, E)$ 와 매칭 M 이 주어

졌을 때, 매칭 M 에 속하는 노드의 집합을 S_M, T_M 이라 하고, 속하지 않는 노드의 집합을 S_X, T_X 라 한다. 부가경로 $(i \rightarrow j)$ 는 노드 $i \in S_X, j \in T_X$를 잇는 경로로 매칭 M 에 속하는 에지와 속하지 않는 에지를 교대로 배치되는 경로이다.

(라인 1) 초기 매칭과 쌍대변수를 구한다. 통상 앞 절의 알고리듬 CMR이 이용된다.

(라인 4) 주어진 매칭 M에 대해 부가 네트워크를 구성 하는 방법은 원네트워크 G 에 대해 매칭 M 에 속한 에지는 방향을 역전시키고 이 네트워크와 가상노드 $s(,t)$ 간의 아크를 첨가한다. 그리고 각 아크의 비용은 (라인 3)에서 계산된 할인비용 reduced cost $\bar{c}_{ij}$를 사용한다. 가상노드를 부가하는 방법은 i) $s \rightarrow S_X$ ii)$T_X \rightarrow t$ iii) i)과 ii)의 조합 세 가지가 있고, 생성된 네트워크를 각각 G_M^+, G_M^-, G_M이라 한다.

(라인 5) 부가 네트워크에 따라 G_M^+는 노드 s에서 다른 노드까지 최단경로를 찾고(이 때 최단경로 트리는 s를 루트 root로 하는 외향성 트리 outward tree), G_M^-는 다른 노드에서 노드 t까지 최단경로를 찾는다(이 때 최단경로 트리는 t를 루트로 하는 내향성 트리 inward tree).

(라인 6) 어느 방법으로 최단경로를 찾더라도 그 최단경로 거리 d 는 쌍대 변수개선에 이용되는데 G_M^+ 경우는 $v_i \leftarrow v_i - d_i \ \ \forall i \in S$, $v_j \leftarrow v_j + d_j \ \forall j \in T$ 이고, G_M^- 경우는 $v_i \leftarrow v_i + d_i \ \ \forall i \in S$, $v_j \leftarrow v_j - d_j \ \forall j \in T$ 이다. 이 때 구해진 쌍대변수는 항상 쌍대가능 해이다.

(라인 7) G_M을 사용하는 경우가 통상의 최단 (단일) 부가경로법이며 T 에서 (s,t)-경로만을 부가경로 사용한다. G_M^+를 사용하는 경우가 최단 다중 부가경로법으로, 주어진 최단경로 트리에서 노드집합이 최대 개수의 상호배타적인 node-disjoint $(S_X \rightarrow T_X)$-경로를 구한다. 이 경로가 부가경로이며 그 개수는 G_M^+, G_M^- 경우에 포리스트 $T \setminus \{s\}$($T \setminus \{s\}$)의 고립된 노드를 제외한 포리스트의 개수와 동일하고, 새로 추가될 매칭의 개수와 같다.

4. 제안 알고리듬

앞 절의 논의에서 최단 다중 부가경로의 경우 논의된 트리 T 에 대해 다음과 같은 에지 집합 E_T을 갖는 이분할 그래프 $G_T = (S_X \cup T_X, E_T)$ 를 고려하자. 여기서, $(i,j) \in E_T \leftrightarrow (i,j) - path \subset T$. 이분할 그래프 G_T는 비용행렬 할인법을 논의 할 때 단계 1에 등장한 G_o의 구조와 동일하다. 즉, T_X에 유일하게 에지 하나씩이 인접하고 S_X는 에지가 인접하는 노드들과 그렇지 않는 노드로 구성됨을 알 수 있다. 즉, 통상의 다중 부가경로법이 비용행렬할인법의 단계 1과 연관성이 있음을 알 수 있다. 따라서 본 논문에서는 비용행렬 할인법의 단계 2에 해당하는 부분을 다중 부가경로법에 적용하여 새로운 부가경로법을 제시한다. 전체적인 구성은 다음알고리듬 newSAP 과 같다

algorithm newSAP
1 $M \leftarrow \varnothing$, $v_i \leftarrow 0 \ \ \forall i \in S \cup T$
2 **while**($M \neq$ maximum)
3 $\bar{c}_{ij} \leftarrow c_{ij} - v_i - v_j$
4 construct augmented network G_M^+ with $\bar{c}_{ij}$
5 $(d, T^+) \leftarrow$ ShortestPath(G_M^+)
6 updated dual variable $v_i \ \forall i \in S \cup T$
7 $\bar{c}_{ij} \leftarrow c_{ij} - v_i - v_j$
8 construct augmented network G_M^- with $\bar{c}_{ij}$
9 $(d, T^-) \leftarrow$ ShortestPath(G_M^-)
10 updated dual variable $v_i \ \forall i \in S \cup T$
11 $P \leftarrow$ set of augmenting path from $T^+ \cup T^-$
12 $M \leftarrow M \oplus P \ \ \forall P \in P$
13 **end**

라인 1과 같이 초기화를 하고 한 회를 수행하면 그 때의 매칭과 쌍대변수는 알고리듬 CMR에서 얻어진 것 과 동일하다. 라인 3-6은 CMR의 단계 1에 상응하는 부분이고 라인 7-10이 단계 2에 상응하는 부분이다. 단계 2에 상응하는 부분의 최단경로 문제는 단계 1에서의 최단경로 문제보다 다음과 같이 계산량을 줄일 수 있다. 즉, 단계 1에서 얻어진 포리스트 $T \setminus \{s\}$ 중의 각 컴포넌트 중에서 T_X를 포함하는 컴포넌트들은 모두 하나의 노드로 축소시켜서 부가네트워크 $\hat{G}_M^-$을 구성할 수 있기 때문이다. 왜냐하면 $\hat{G}_M^-$에서 최단경로를 구하면 앞에 언급한 컴포넌트들에 속하는 노드의 최단경로의 거리는 0이고, 그 구조는 그 컴포넌트 안에 포함되기 때문이다.

5. 결론

본 논문에서는 배정에 대한 새로운 최단 다중 경로 알고리듬을 제시하였다. 제안된 알고리듬과 기존의 알고리듬 간에는 한 회에서의 개선량과 계산량이 상충하지만, 계산복잡도는 동일하다.

참고문헌

Ahuja, R.K., Magnanti, T.S., and Orlin, J.B. (1993), *Network Flows: Theory, Algorithms, and Applications*, Prentice Hall, Upper Saddle River, New Jersey.

Carraresi, P., and Sodini, C. (1986), An efficient algorithm for the bipartite matching problem, *European Journal of Operational Research*, **23**, 86-93.

Nawijn, W.W., and Dorhout, B. (1989), On the expective number of assignments in reduced matrices for the linear assignment problem, *Operations Research Letterse*, **8**, 329-335.

Edmonds, J., and Karp, R.M. (1972), Theoretical

improvements in algorithmic efficiency for network flow problems, *Jurna of the ACMl*, **19**.

Lawler, E. (1976), *Combimatorial Optimization: Networks and Matroids*, Dover Publications, Mineola, New York.

Papadimitriou, C.H., and Steiglitz, K.. (1982), *Combimatorial Optimization: Algorithms and Complexity*, Dover Publications, Englewood Cliffs, New Jersey, U.S.A..

Schrijver, A. (2003), *Combinatorial Optimization: Polyhedra and Efficiency*, Springer-Verlag, Berlin Heidelberg.