OPTIMAL KNOWLEDGE FLOW ON THE INTERNET

Burak ORDİN*, Urfat NURİYEV**

ABSTRACT
The flow problem and the minimum spanning tree problem are both fundamental in operational research and computer science. We are concerned with a new problem which is a combination of maximum flow and minimum spanning tree problems. The applied interpretation of the expressed problem is to correspond an optimal knowledge flow on the internet. Although there are polynomial algorithms for the maximum flow problem and the minimum spanning tree problem, the defined problem is NP-Complete. It is shown that the optimal solution of the problem corresponds an equilibrium state in subproblem which is an auxiliary problem of Cutting Angle Method in solving of the Global Optimization Problems and the developed algorithms for solving of the subproblem could be used to solve the expressing problem.

Keywords: Optimal Knowledge Flow, Maximum Flow Problem, Minimum Spanning Tree Problem, Cutting Angle Method, Global Optimization

İNTERNET ÜZERİNDE OPTİMAL BİLGİ AKIŞI

ÖZET
Akış ve Minimum Kapsayan Ağaç problemleri Yöneylem Araştırması'nda ve Bilgisayar Bilimleri'nde karşılaşılan temel problemlerdendir. Yapılan çalışmada, maksimum akış problemi ve minimum kapsayan ağaç probleminin bileşimi şeklinde ele alınabilecek yeni bir problem incelenmiştir. İfade edilen problemle, bilgi akışının olduğu internet ortamında karşılaşılmaktadır. Maksimum Akış Problemi ve Minimum Kapsayan Ağaç problemi için polinom zamanda çözüm veren algoritmalar bulunmasına rağmen tanımlanan problem NP-Tam sınıftandır. Problemin optimal çözümü, Global Optimizasyon problemleri'nin geniş bir sınıfının çözümünde karşılaşılan "Yardımcı Altproblem" in çözümündeki denge durumuna karşı gelmektedir. Gösterilmiştir ki, Yardımcı Alt problem'in çözümü için geliştirilen algoritmalar, bu çalışmada incelenen probleme de uyarlanabilir.

Anahtar Kelimeler: Optimal Bilgi Akışı, Maksimum Akış Problemi, Minimum Kapsayan Ağaç Problemi, Kesen Açılar Yöntemi, Global Optimizasyon.

* *Ege ÜniversitesiFen Fakültesi, Matematik Bölümü,Bilgisayar Bilimleri AB, Bornova, İzmir, burak.ordin@ege.edu.tr*
** *Ege ÜniversitesiFen Fakültesi, Matematik Bölümü,Bilgisayar Bilimleri AB, Bornova, İzmir, urfat.nuriyev@ege.edu.tr*

1. INTRODUCTION

A lot of problems in computer science and economy are interpreted as using of the maximum flow problem and the minimum spanning tree problem. Maximum flow problem is to determine a least cost shipment of a commodity through a network G = (V, E) which is a graph with V vertices and E edges in order to satisfy demands at certain nodes from available supplies at other nodes. As for the *minimum spanning tree problem that* is defined as follows: find an acyclic subset *T* of *E* that connects all of the vertices in the graph and whose total weight is minimized, where the total weight is given by *w(T)* = sum of *w(u,v)* over all *(u,v)* in *T*, where *w(u,v)* is the weight on the edge *(u,v).T* is called the *spanning tree (Papadimitriou, 1982).*

In this paper, it is studied a combination of above problems. That is, we have a source point, total flow and we want to find n sinks in m points and flow (capacity) for each line as different from maximum flow problem where is given source points, sinks, the capacities of the edges for a graph G=(V, E) and it is wanted to be determined maximum flow. Also, the problem is thought as a minimum spanning problem where the purpose is to cover all points by a chosen tree having a minimal total weight, when we determine flows in each line.

The applied interpretation of the defined problem is to correspond that: a knowledge is to be carried from one main computer to n servers and from the n servers to m clients ($m \geq n$). The processing must be ensured the following conditions:

- The sending time is minimum.
- All servers are satisfied.

Although there are polynomial algorithms for the maximum flow problem and the minimum spanning tree problem, the expressing problem is NP-Complete. It is shown that the optimal solution of the problem corresponds an equilibrium state in subproblem which is an auxiliary problem of Cutting Angle Method in solving of the Global Optimization Problems. Also, a combinatorial optimization problem that equals to the subproblem is used to solve the expressed problem.

2. A GLOBAL OPTIMIZATION TECHNIQUE

The Cutting Angle Method (CAM) has been developed as global optimization technique since 1997. In this method the original global optimization problem is reduced to a sequence of subproblems where the objective function is the maximum of special min-type functions (Andramonov et.al., 1999). This method is an iterative one requiring the solution of a subproblem (minimizing functions *f*(*x*), defined on the set $S=\{x : \sum_{i=1}^{n} x_i = 1,\ x_i \geq 0,\ i = 1, \ldots, n\}$, where $x = (x_1, \ldots, x_n)$), which is, generally, a global optimization problem. Computational efficiency of the CAM method is significantly affected by the efficiency of solving the subproblem, which is solved at each iteration.

The formulation of the Subproblem could be given as follow.
Let (l_i^k) be an ($m * n$) matrix, $m \geq n$, with m rows l^k, k = 1, …, m, and n columns, i = 1, ……, n. All elements l_i^k are nonnegative and the first n rows of (l_i^k) matrix form a diagonal matrix, i.e., l_i^k >0, only for $k = i$, i=1, …, n.
Introduce the function $h(x)= \max_k \min_{i \in I(l^k)} l_i^k x_i$, where $I(l^k)=\{i: l_i^k > 0\}$.
And the subproblem

Minimize h(x) (2.1)

subject to

$$x \in S=\{x: \sum_{i=1}^{n} x_i = 1,\ x_i \geq 0,\ i=1,\dots,n.\} \tag{2.2}$$

3. FLOW BALANCING PROBLEM WITH MINIMUM SPANNING TREE

Let us take a graph $G=(V, E)$ which is directed and weighted. it corresponds to a matrix (l_i^k) as follow for $i=1, 2,\dots,n;\ k=1,\dots,n,\ n+1,\dots,\ m.$

According to the columns, the elements of the matrix (l_i^k) for $k=n+1,\ n+2,\dots,m$ (namely, last p rows, p=m-n) are sorted like (3.1).

$$l_i^{j_1^i} \leq l_i^{j_2^i} \leq \dots \leq l_i^{j_p^i} \qquad , i=1,\dots,n \tag{3.1}$$

Here, the set of the vertices, V is defined as
$$V = \{0, 1,\dots, n, n+1, n+2,\dots, m\}, \qquad |V| = m+1,$$

Vertex 0 which has degree n is the root of the building a spanning tree. In the other words, each of vertices 1,…,n connects vertex 0 with one edge.

The first n vertices (1, 2,…, n) correspond to the first n rows of the matrix and remaining p vertices (n+1, n+2, …,m) correspond to the last p rows of the matrix.

The set of the edges E is defined as

$$E=\{\ \overrightarrow{(0,1)},\overrightarrow{(0,2)},...,\overrightarrow{(0,\ n)}\ ;\ \overrightarrow{(1,\ j_1^1)},\ \overrightarrow{(j_1^1,\ j_2^1)},...,\ \overrightarrow{(j_{p-1}^1,\ j_p^1)}\ ;$$

$$\overrightarrow{(2,j_1^2)},\ \overrightarrow{(j_1^2,j_2^2)},...,\overrightarrow{(j_{p-1}^2,j_p^2)};...\overrightarrow{(i,j_1^i)},\ \overrightarrow{(j_1^i,j_2^i)},...,\ \overrightarrow{(j_{p-1}^i,\ j_p^i)}\ ;$$
$$...\overrightarrow{(n,j_1^n)},\overrightarrow{(j_1^n,j_2^n)},...,\overrightarrow{(j_{p-1}^n,j_p^n)}\ \}$$

In E, the edges in the first row connect vertex 0 with vertices 1, 2, …,n. The edges in second row join for i=1 vertex 1 with j_1^1, j_1^1 with j_2^1 and finally j_{p-1}^1 with j_p^1 according to the line (3.1). The edges in third row join for i=2 vertex 2 with j_1^2, j_1^2 with j_2^2 and at last j_{p-1}^2 with j_p^2, according to the line (3.1). At the end, the edges in the last row connect for i=*n* vertex n with j_1^n, j_1^n with j_2^n and finally j_{p-1}^n with j_p^n. Clearly, there are n edges in row 1 and there are p edges in each other n rows. The total number of edges is $n+pn=n(p+1)=n[(m-n)+1]=nm-n^2+n$.

Each edge $\overrightarrow{(0,i)}$, *(i=1,...,n)* has l_i^i weight and the weights of the edges $\overrightarrow{(j_{t-1}^i,j_t^i)}$ are equal to $\left(l_i^{j_t^i}-l_i^{j_{t-1}^i}\right)$. Here, we assume that $j_0^i=i$, *(i=1,...,n; t=1,...,p)*. The vertex 0 that is the initial point is the source (main computer) and at the source point, the flow which ensures conditions (2.1)-(2.2) is divided into n pieces (servers): $x_1+x_2+...+x_n=1$ (an x_i corresponds to each line *(0,i) (i=1,…,n)*). By choosing x_i's, n sinks are determined among 1,2,...,m vertices. Then there are (m-n) clients.

In conclusion, there is a flow problem which has a source vertex and n sinks. When x_i, is determined the weights of the edges are $l_1^i.x_i$ for edges $\overrightarrow{(0,i)}$ *(i=1,2,…,n)* and $(l_i^{j_t^i}-l_i^{j_{t-1}^i}).x_i$ for edges $(j_{t-1}^i,j_t^i)\,(i=1,2,...,n;\ t=1,2,...,p)$. Chosen tree for the graph determined by new edges is the Minimal Weighted Spanning tree. In other words, we have a main computer, n servers and (m-n) clients. Also, we know arrival costs from servers to each client. Then the problem is to determine the best servers for clients.

Flow Balancing Problem with Minimum Spanning Tree equals to subproblem and the developed methods for subproblem could be used to solve Flow Balancing Problem with Minimum Spanning Tree.

4. TRANSFORMATION OF THE SUBPROBLEM TO AN EQUIVALENT PROBLEM

The following notation is used for simplicity, (Nuriyev and Ordin, 2003; Nuriyev and Ordin, 2004).

$$p = m - n, \quad u_i^j = \frac{1}{l_i^i} - \frac{1}{l_i^{j+n}}, \quad i = 1,2,\ldots,n\,;\ j = 1,2,\ldots,p.$$

Clearly, u_i^j is the increment of the denominator of the fraction that expresses the function h in the substitution $l_i^{j+n} \rightarrow l_i^i$.

Let us define the following function:

$Sg(x) = \begin{cases} 1, & if\ x \geq 0, \\ 0, & \text{if } x < 0 \end{cases}$ and consider variables x_i^j, i = 1,2,…,n ; j = 1,2,…,p

:

$$x_i^j = \begin{cases} 1, & if\ the\ substitution\ l_i^{j+n} \rightarrow l_i^j \text{ is accomplished} \\ 0, & \text{otherwise} \end{cases}$$

So the Subproblem (2.1)-(2.2) is transformed to the following Boolean (0 – 1) programming problem :

$$\sum_{i=1}^{n} \sum_{j=1}^{p} u_i^j x_i^j \rightarrow \min_{x_i^j} \tag{4.1}$$

$$\sum_{i=1}^{n} x_i^j \leq 1, \quad j=1,2,\ldots,p, \tag{4.2}$$

$$\sum_{j=1}^{p} x_i^j \leq 1, \quad i=1,2,\ldots,n, \tag{4.3}$$

$$\sum_{i=1}^{n} \sum_{j=1}^{p} x_i^j \geq 1, \tag{4.4}$$

$$\sum_{i=1}^{n} y_i^j \geq 1, \quad j = 1,2,\ldots,p, \tag{4.5}$$

$$x_i^j = 0 \vee 1, \quad i = 1,2,\ldots,n;\ j=1,2,\ldots,p, \tag{4.6}$$

$$y_i^j = Sg\left(\max_{k=\overline{1,p}} \left\{ u_i^k x_i^k \right\} - u_i^j \right), \quad i = 1,2,\ldots,n\,;\ j = 1,2,\ldots,p, \tag{4.7}$$

THEOREM 4.1. The Subproblem (2.1)-(2.2) and the problem (4.1)-(4.7) are equivalent, (Nuriyev, 2005).
Let us call the problem (4.1)-(4.7) as “Dominating Subset with Minimal Weight (DSMW)”. The problem can be interpreted as follows:
Let (u_i^j) be a matrix, with p rows, j = 1, 2, …, p and n columns, i = 1, 2, …, n and nonnegative u_i^j for all i, j .

The task is to choose some elements of the matrix such that:

i) The sum of the chosen elements is minimal,

ii) Each row contains a chosen element, or contains some element which is less than some chosen element located in its column.

It is given the following applied interpretation of this problem: A task consisting of p (j = 1, 2, …, p) operations can be accomplished by n (i = 1, 2, …,n) processors. Suppose that the matrix (u_i^j) gives the time necessary for accomplishment of the task as follows: If

$$u_i^{j_1} \leq u_i^{j_2} \leq \ldots \leq u_i^{j_p} \tag{4.8}$$

for column i, then $u_i^{j_1}$ is the time (or cost) for the accomplishment of operation j_1 by processor i; $u_i^{j_2}$ is the time for the accomplishment of operations j_1 and j_2 by processor i, and so on. At last $u_i^{j_p}$ is the time for the accomplishment of all operations ($j_1, j_2, \ldots, j_p$) by processor i.

The problem is to distribute operations among the processors minimizing the total time (or the total cost) required for the accomplishment of all tasks.

Clearly, the problem is generalized of the Assignment problem. Although the assignment problem can be solved by Hungarian method at a complexity of $O(r^3)$ ($r = max\{p, n\}$), Papadimitriou(1982).

THEOREM 4.2. DSMW problem is NP-Complete, Nuriyev et al (2003). There are good heuristics for solving of the DSMW problem Nuriyev et al (2004).
EXAMPLE 4.1. Let us take a subproblem matrix that is diagonal.

$$(l_i^k) = \begin{bmatrix} 2 & 0 \\ 0 & 4 \\ 3 & 8 \\ 5 & 6 \\ 4 & 7 \end{bmatrix}$$

Here, $n=2$, $m=5$ and $p=3$. For the matrix, condition (3.1) is determined as the following:

$$l_1^3 < l_1^5 < l_1^4, l_2^4 < l_2^5 < l_2^3 \tag{4.9}$$

Namely, $j_1^1 = 3, j_2^1 = 5, j_3^1 = 4$ and $j_1^2 = 4, j_2^2 = 5, j_3^2 = 3$. For the above conditions, the corresponding graph is taken as follows.

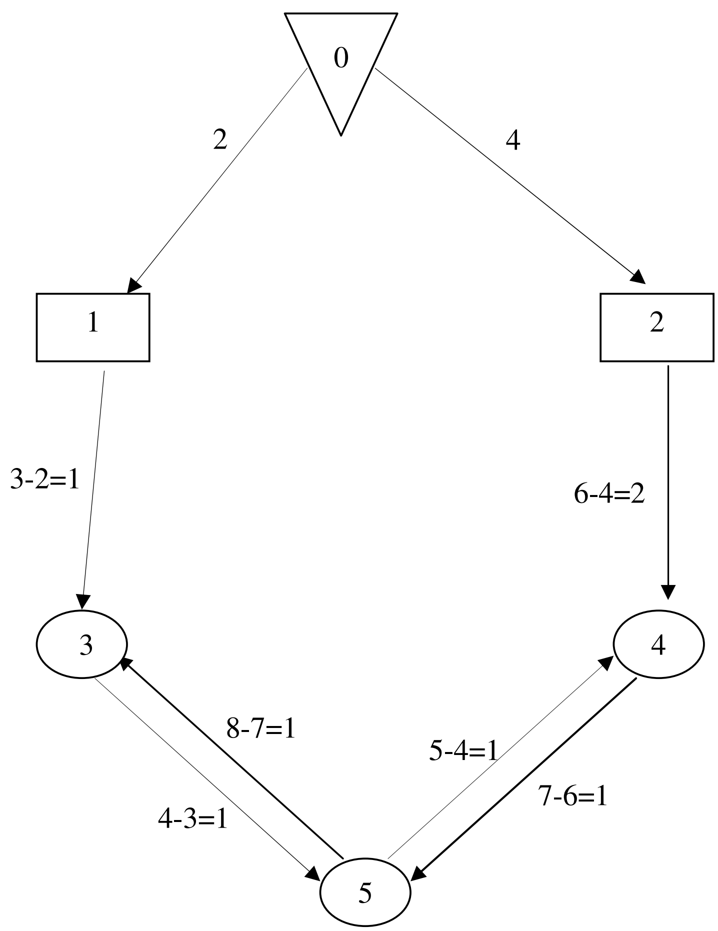


Figure 4.1. The Graph Notation of the Problem

For the solution of the problem, subproblem matrix is transformed into Dominating Subset with the Minimal Weight (DSMW) problem matrix by the expressed way.

$$(u_i^j) = \begin{bmatrix} 0.16 & 0.12 \\ 0.30 & 0.08 \\ 0.25 & 0.10 \end{bmatrix}$$

Then, DSMW problem for the matrix is solved by the method proposed in Nuriyev et al (2004) and for this we take $x_2^1 = 1$ and $x_i^j = 0$, $i \neq 2$, $j \neq 1$. Then we have the following result.

$$h(x) = \frac{1}{\frac{1}{2} + \frac{1}{8}} = \frac{16}{10} ,$$

the values of x_i : $x_1 = \frac{16}{10} : 2 = \frac{8}{10}$, $\qquad x_2 = \frac{16}{10} : 8 = \frac{2}{10}$.

A matrix is taken as follows for $x_1 = \frac{8}{10}, x_2 = \frac{2}{10}$

$$\begin{bmatrix} 2 \times \frac{8}{10} = \frac{16}{10} & 0 \\ 0 & 4 \times \frac{2}{10} = \frac{8}{10} \\ 3 \times \frac{8}{10} = \frac{24}{10} & 8 \times \frac{2}{10} = \frac{16}{10} \\ 5 \times \frac{8}{10} = \frac{40}{10} & 6 \times \frac{2}{10} = \frac{12}{10} \\ 4 \times \frac{8}{10} = \frac{32}{10} & 7 \times \frac{2}{10} = \frac{14}{10} \end{bmatrix}$$

The corresponding graph for the above matrix is as follows:

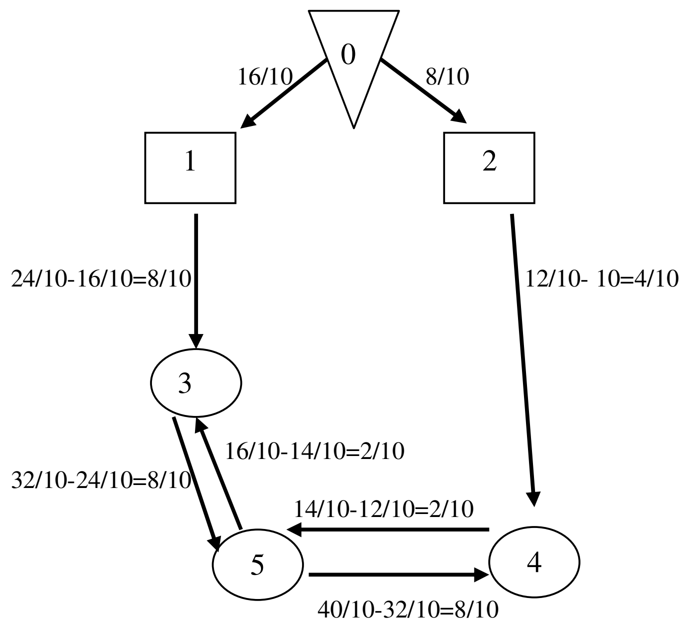


Figure 4.2. The Expressing of the Problem as Subproblem

That is, vertex 2 is chosen for i=1 and vertex 3 is chosen for i=2. In other words, flow is from 0 to 1 and from 0 to 2, from 2 to 4, from 4 to 5, from 5 to 3. For the graph, The Minimal weighted spanning tree is taken as:

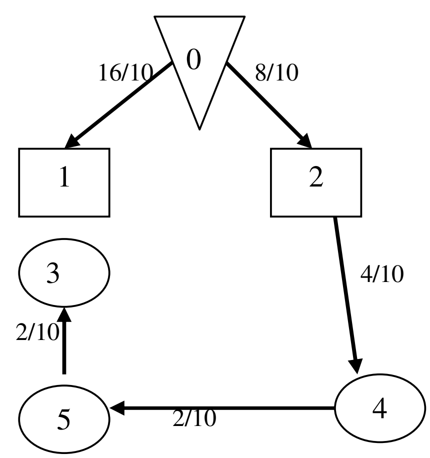
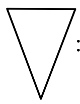
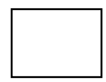


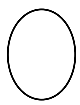
Figure 4.3. The Notation of Solving of the Problem as Tree Graph

NOTE 4.1. In this example, small values of m and n are taken for simplicity.

Besides, for graphs, we use the following description


: Source Point (Main Computer), (Servers)


Main Distribution Points


: Points (Clients)

5. CONCLUSION

In this work it is expressed a new problem which is a combination of flow problem and the minimum spanning tree problem. The problem is to correspond an optimal knowledge flow on the internet. Although there are polynomial algorithms for the maximum flow problem and the minimum spanning tree problem, the problem is NP-Complete. The optimal solution of the problem corresponds an equilibrium state in subproblem which is an auxiliary problem of CAM in solving of the Global Optimization Problems and the algorithms that are developed for solving of the DSMW problem could be used effectively for solving of the expressed problem.

6. REFERENCES

Andramonov, M.Y., Rubinov, A.M. and Glover, B.M., (1999), "Cutting Angle Methods in Global Optimization", Applied Mathematics Letters, 12, 95 - 100.

Babayev, D.A., (2000), "An Exact Method for Solving the Subproblem of the Cutting Angle Method of Global Optimization", In Book "Optimization and Related Topics", In Kluwer Academic Publishers, Ser. "Applied Optimization", Dordrecht, 47, 15 – 26.

Nuriyev, U.G., (2005), "An Approach to the Subproblem of the Cutting Angle Method of Global Optimization", Journal of Global Optimization, 31, 2005, 353-370.

Nuriyev U.G. and Ordin B., (2004), "Computing Near-Optimal Solutions for the Dominating Subset with Minimal Weight Problem", International Journal of Computer Mathematics, 81, 11.

Nuriyev U.G. and Ordin B., (2003), "Analysis of the Complexity of the Dominating Subset with the Minimal Weight Problem", EURO / INFORMS Conferences, July, 2003, Istanbul, Turkey.

Ordin B., (2004), "On the Subproblem of the Cutting Angle Method of Global Optimization", The Euro Summer Institute-ESI XXII Optimization and Data Mining, July 9-25, 2004, Ankara, Turkey.

Papadimitriou, C. H., Steiglitz, K., (1982), "Combinatorial Optimization: Algorithms and Complexity", Prentice Hall, 496.