社団法人 情報処理学会 研究報告
IPSJ SIG Technical Report

2006－AL－107 (2)
2006／7／3

グラフの最小コスト部分分割について

永持 仁[†] 上土井 陽子[††]

† 京都大学情報学研究科
606-8501 京都市左京区吉田本町
†† 広島市立大学情報科学部
731-3194 広島市安佐南区大塚東三丁目４－１
E-mail: †yoko@ce.hiroshima-cu.ac.jp, ††nag@amp.i.kyoto-u.ac.jp

あらまし 枝重み付きグラフ $G=(V,E)$，節点部分集合 $S\subseteq V$，整数 $k\geq 1$，実数 $b\geq 0$ が与えられたときに最小部分分割問題を $1\leq i\leq k$ なる各 i において $d(X_i)\leq b$ を満たし，かつ，$\sum_{1\leq i\leq k} d(X_i)$ を最小とする互いに素な空でない k 個の部分集合 $X_1,X_2,\ldots,X_k\subseteq S$ を見つける問題とする．ここで，$d(X)$ は X と $V-X$ の間の枝の重みの総和とする．本稿では最小部分分割問題が $O(mn+n^2\log n)$ で解決できることを示す．また，上記の結果を最小コスト k 分割問題，グラフ強度問題に適用することでそれらの問題に対する２倍近似アルゴリズムの時間計算量を $O(mn+n^2\log n)$ に改良できることを示す．

Minimum Cost Subpartitions in Graphs

Hiroshi NAGAMOCHI[†] and Yoko KAMIDOI[††]

† Dept. of Applied Mathematics and Physics, Kyoto University
Sakyo, Kyoto, 606-8501, Japan
†† Faculty of Information Sciences, Hiroshima City University
3-4-1, Ohzuka-Higashi, Asaminami-ku, Hiroshima, 731-3194, Japan
E-mail: †yoko@ce.hiroshima-cu.ac.jp, ††nag@amp.i.kyoto-u.ac.jp

Abstract Given an edge-weighted graph $G=(V,E)$, a subset $S\subseteq V$, an integer $k\geq 1$ and a real $b\geq 0$, the minimum subpartition problem asks to find a family of k nonempty disjoint subsets $X_1,X_2,\ldots,X_k\subseteq S$ with $d(X_i)\leq b$, $1\leq i\leq k$ so as to minimize $\sum_{1\leq i\leq k} d(X_i)$, where $d(X)$ denotes the total weight of edges between X and $V-X$. In this paper, we show that the minimum subpartition problem can be solved in $O(mn+n^2\log n)$ time. The result is then applied to the minimum k-way cut problem and the graph strength problem to improve the previously best time bounds of 2-approximation algorithms for these problems to $O(mn+n^2\log n)$.

1. Introduction

Let $G=(V,E)$ be a simple undirected graph with a vertex set V and an edge set E such that each edge e is weighted by a nonnegative real $w(e)$. The vertex set and the edge set of G may be denoted by $V(G)$ and $E(G)$, respectively. Let $n=|V(G)|$ and $m=|E(G)|$. For a subset $F\subseteq E(G)$, we denote $\sum_{e\in F} w(e)$ by $w(F)$. For nonempty sets $X,Y\subseteq V$, $E(X,Y)$ denotes the set of edges in G such that one end vertex is in $X-Y$ and the other is in $Y-X$, and $d(X,Y)$ denotes $w(E(X,Y))$. We may denote $d(X,V-X)$ by $d(X)$.

A *k-subpartition* of a subset X of V is a set of k disjoint nonempty subsets of X, and is called a *k-partition* of X if the union of all its subsets is X. For a real $b\geq 0$, a k-subpartition (resp., k-partition) of a subset X is called a (k,b)-subpartition (resp., (k,b)-partition) $\mathcal{S}$ if $d(X)\leq b$ for all $X\in\mathcal{S}$. We consider the following problem.

Minimum Subpartition Problem

Input: An instance $I=(G,S,k,b)$ which consists of an edge-weighted graph G, a nonempty subset

$S \subseteq V(G)$, an integer $k \in [1, |S|]$, and a real $b \geqq 0$.
Feasible solution: A (k, b)-subpartition $\mathcal{S}$ of S.
Goal: Minimize $cost(\mathcal{S}) := \sum_{X \in \mathcal{S}} d(X)$.

In this paper, we prove the next results.

[Theorem 1] Given an edge-weighted graph G, a subset $S \subseteq V$ and a real $b \geqq 0$, a minimum (k, b)-subpartition of S (if any) for each $k \in [1, |S|]$ can be obtained in $O(mn + n^2 \log n)$ time. ◇

2. Preliminaries

A set consisting of a single vertex is called *trivial*. Two subsets $X, Y \subseteq V$ *intersect* each other if $X \cap Y \neq \emptyset$, $X - Y \neq \emptyset$ and $Y - X \neq \emptyset$ hold. A family $\mathcal{Y} \subseteq 2^V$ of subsets of V is called *laminar* if no two subsets in $\mathcal{Y}$ intersect each other. Let $\mathcal{Y} \subseteq 2^S$ be a laminar family of subsets of a subset $S \subseteq V$. We can represent $\mathcal{Y}$ with $S \in \mathcal{Y}$ by a rooted tree $\mathcal{T}$ as follows, where we use term "nodes" for the vertices in tree representations.

(i) The node set $V(\mathcal{T})$ of $\mathcal{T}$ consists of nodes each of which corresponds to a subset $X \in \mathcal{Y}$, i.e., $V(\mathcal{T}) = \mathcal{Y}$, where the root corresponds to S.

(ii) For two nodes $X, Y \in V(\mathcal{T})$, X is a child of Y in $\mathcal{T}$ if and only if $X \subset Y$ holds and $\mathcal{Y}$ contains no set X' with $X \subset X' \subset Y$.

From this, we observe that $|\mathcal{Y}| \leqq 2|S| - 1$ holds.

Let $\mathcal{X} \subseteq 2^S$ be a laminar family of subsets of a subset $S \subseteq V$, where $|\mathcal{X}| \leqq 2|S| - 1$ holds. We easily see that $2|S| - 1 - |\mathcal{X}|$ new subsets of S can be added to $\mathcal{X}$ so that the resulting family $\mathcal{X} \cup \mathcal{X}'$ remains laminar, where $\mathcal{X}'$ denotes the set of added subsets. We first add all trivial sets $\{v\} \notin \mathcal{X}$, and then consider the tree representation $\mathcal{T}'$ of the resulting laminar family $\mathcal{X} \cup \{\{v\} \notin \mathcal{X}\}$. We transform $\mathcal{T}'$ into a binary rooted tree by inserting $2|S| - 1 - |\mathcal{X} \cup \{\{v\} \notin \mathcal{X}\}|$ new nodes, where each new node corresponds to a new subset in $\mathcal{X}'$. A family $\mathcal{X} \cup \mathcal{X}'$ can be represented by a rooted binary tree $\mathcal{T}$, as observed in the above. We call such a binary tree $\mathcal{T}$ *consistent with* S and $\mathcal{X}$.

a) Extreme vertex sets

Extreme vertex sets are first introduced by Watanabe and Nakamura [11] to solve the edge-connectivity augmentation problem. A nonempty proper subset X of V is called an *extreme vertex set* of G if

$$d(Y) > d(X) \text{ for all nonempty proper subsets } Y \text{ of } X.$$

We denote by $\mathcal{X}(G)$ the family of all extreme vertex sets of G. Any trivial set $\{v\}$, $v \in V$ is an extreme vertex set. It is known that the family $\mathcal{X}(G)$ is laminar (see [7] for example).

[Lemma 2] [7] Given an edge-weighted graph $G = (V, E)$, the family $\mathcal{X}(G)$ of extreme vertex sets and $\{d(X) \mid X \in \mathcal{X}(G)\}$ can be found in $O(mn + n^2 \log n)$ time. ◇

For a subset $S \subseteq V$ and a real $b \geqq 0$, we denote by $\mathcal{X}_S(G)$ (resp., $\mathcal{X}_{S,b}(G)$) the family of all extreme vertex sets $X \in \mathcal{X}(G)$ with $X \subseteq S$ (resp., $X \subseteq S$ and $d(X) \leqq b$).

[Lemma 3] Let $I = (G, S, k, b)$ be an instance of the minimum subpartition problem. Then

(i) There is a (k, b)-subpartition of S if and only if $\mathcal{X}_{S,b}(G)$ contains a family of k disjoint subsets.

(ii) There is a minimum (k, b)-subpartition of S which consists of extreme vertex subsets in $\mathcal{X}_{S,b}(G)$.

Proof. (i) The sufficiency is trivial. We show the necessity, assuming that I has a minimum (k, b)-subpartition $\mathcal{S} = \{X_1, \ldots, X_k\}$ of S. For each subset X_i, let X_i^* be an inclusionwise minimal subset of X_i with $d(X_i^*) \leqq b$. By the minimality, $d(Y) > b \geqq d(X_i^*)$ holds for all nonempty proper subsets $Y \subset X_i^*$, implying that X_i^* is an extreme vertex set such that $X_i^* \subseteq S$ and $d(X_i^*) \leqq b$. Therefore, $\{X_1^*, \ldots, X_k^*\}$ is a family of k disjoint extreme vertex sets in $\mathcal{X}_{S,b}(G)$, proving the necessity.

(ii) Let $\mathcal{S} = \{X_1, \ldots, X_k\}$ be a minimum (k, b)-subpartition of S, where we assume without loss of generality that $\mathcal{S}$ minimizes $\sum_{1 \leq i \leq k} |X_i|$ among all minimum (k, b)-subpartitions of S. Since $d(X_i) \leqq b$, it suffices to show that each X_i is an extreme vertex set. If X_i contains a subset Y with $d(Y) \leqq d(X_i)$, then $(\mathcal{S} - \{X_i\}) \cup \{Y\}$ remains a minimum (k, b)-subpartition of S, which however contradicts the minimumity of $\sum_{1 \leq i \leq k} |X_i|$. Therefore, $\mathcal{S}$ consists of subsets in $\mathcal{X}_{S,b}(G)$. ◇

3. Algorithm

To prove Theorem 1, we first compute the family $\mathcal{X}(G)$ of all extreme vertex sets in G. This can be done in $O(mn + n^2 \log n)$ time by Lemma 2. We then construct a binary tree $\mathcal{T} = (\mathcal{V} = \mathcal{X}_S \cup \mathcal{X}', \mathcal{E})$ consistent with S and $\mathcal{X}_S(G)$. (Actually the algorithm in [7] can be easily modified so as to output such a binary tree in the same time complexity.)

Let $opt(X, k)$ denote the minimum cost of a (k, b)-subpartition of a subset X, where we define

$opt(X,0) = 0$ and $opt(X,k) = +\infty$ for all k such that G has no (k,b)-subpartition of X. We compute $opt(X,k)$, $k \in [1,|X|]$, $X \in \mathcal{V}$ by dynamic programming as follows.

The set of leaves in $\mathcal{T}$ consists of trivial sets $\{u\}$, $u \in S$. For each leaf $X = \{u\} \in \mathcal{X}_S(G)$, we have

$$opt(X,1) = \begin{cases} d(\{u\}) & \text{if } d(\{u\}) \leq b, \\ +\infty & \text{if } d(\{u\}) > b. \end{cases} \quad (1)$$

Consider a nonleaf $X \in \mathcal{V}$, and let Y_1 and Y_2 be the two children of X in $\mathcal{T}$, where it hold

$$\mathcal{X}_{X,b}(G) - \{X\} = \mathcal{X}_{Y_1,b}(G) \cup \mathcal{X}_{Y_2,b}(G)$$
$$\text{and } \mathcal{X}_{Y_1,b}(G) \cap \mathcal{X}_{Y_2,b}(G) = \emptyset. \quad (2)$$

By Lemma 3, there is a minimum (k,b)-subpartition of X (if any) which consists of subsets in $\mathcal{X}_{X,b}(G)$. Hence by (2) we see that, for each $k \in [2,|X|]$, there is an integer $i \in [0,k]$ such that $opt(X,k) = opt(Y_1,i) + opt(Y_2,k-i)$. Thus it holds

$$opt(X,k) = \min_{0 \leq i \leq h} \{opt(Y_1,i) + opt(Y_2,k-i)\}. \quad (3)$$

For $k = 1$,

$$opt(X,1) = \begin{cases} d(X) & \text{if } X \in \mathcal{X}_{S,b}(G), \\ \min\{opt(Y_1,1),\ opt(Y_2,1)\} & \text{if } X \notin \mathcal{X}_{S,b}(G) \end{cases} \quad (4)$$

(recall that $d(Y) > d(X)$ holds for all proper subset Y of $X \in \mathcal{X}(G)$).

Algorithm SUBPARTITION(S,b)

Input: A binary tree $\mathcal{T} = (\mathcal{V} = \mathcal{X}_S \cup \mathcal{X}', \mathcal{E})$ consistent with S and $\mathcal{X}_S(G)$, and a real $b \geq 0$.

Output: $\{opt(X,k) \mid 1 \leq k \leq |X|\}$ for all nodes $X \in \mathcal{V}$.

For each leaf $X = \{u\} \in \mathcal{X}_S(G)$,
  compute $opt(X,1)$ according to (1);
while there is unprocessed node in $\mathcal{T}$ **do**
  Choose a loweset unprocessed node X, and
    compute $opt(X,1)$ and
  $\{opt(X,h) \mid 2 \leq h \leq |X|\}$ according to (4) and (3),
    respectively
end /* while */

The time complexity of algorithm SUBPARTITION is analyzed as follows. Note that $\{opt(X,k) \mid 1 \leq k \leq |X|\}$ can be computed in $O(|Y_1||Y_2|)$ time if $\{opt(Y_j,k) \mid 1 \leq k \leq |Y_j|\}$, $j = 1,2$ is available.

[Lemma 4] Given a binary tree $\mathcal{T}$ consistent with S and $\mathcal{X}_S(G)$ and a real $b \geq 0$, we can compute $\{opt(X,k) \mid 1 \leq k \leq |X|\}$ for all nodes $X \in \mathcal{V}$ in $O(|S|^2)$ time.

Proof. Let $t(n)$ denote the time required to compute $\{opt(X,k) \mid 1 \leq k \leq |X|\}$ for a subset X with $|X| = n$. Choose a constant c such that $t(1) \leq c$ and $\{opt(X,k) \mid 1 \leq k \leq |X|\}$ can be computed in at most $2c|Y_1||Y_2|$ time from $\{opt(Y_j,k) \mid 1 \leq k \leq |Y_j|\}$, $j = 1,2$. We show that $t(n) \leq cn^2$. For $n = 1$, this is trival. Assuming that $t(\hat{n}) \leq c\hat{n}^2$ holds $\hat{n} \in [1, n-1]$, we prove $t(n) \leq cn^2$ holds. Consider a nonleaf $X \in \mathcal{V}$ with $|X| = n$, and its two children Y_1 and Y_2 of X in $\mathcal{T}$. Let $n_1 = |Y_1|$ and $n_2 = |Y_2|$. Then by induction hypothesis, we have

$$t(n) \leq t(n_1) + t(n_2) + 2cn_1n_2 \leq cn_1^2 + cn_2^2 + 2cn_1n_2 = cn^2.$$

This proves the lemma. ◇

For each $k \in [1,|S|]$, if $opt(S,k) = +\infty$, then S has no (k,b)-subpartition. Otherwise, a minimum (k,b)-subpartition of S can be obtained by retrieving the integers i that attain the minimum in the recursive formula, which takes $O(|S|)$ time. This establishes Theorem 1.

4. Applications

4.1 Minimum k-Way Cut

For an edge-weighted graph $G = (V,E)$, a subset F of edges is called a k-way cut if removal of F from G results in at least k connected components. We denote by $\mu(G,k)$ the cost of a minimum k-way cut of G. We easily see that there is a minimum k-way cut F which is given by $\cup_{1 \leq i < j \leq k} E(V_i, V_j)$ for some k-partition $\{V_1, V_2, \ldots, V_k\}$ of V. Thus, the minimum k-way cut problem asks to find a k-partition $\mathcal{Z} = \{V_1, V_2, \ldots, V_k\}$ of V that minimizes $cost(\mathcal{Z}) = \sum_{V_i \in \mathcal{Z}} d(V_i)$. Goldschmidt and Hochbaum [2] proved that the problem is NP-hard if k is an input parameter, and presented an $O(n^{k^2/2-3k/2+4}F(n,m))$ time algorithm, where $F(n,m)$ denotes a time bound of a maximum flow algorithm in an edge-weighted graph with n vertices and m edges. Recently the time bound is improved to $O(n^{4k/(1-1.71/\sqrt{k})-31})$ [4]. Karger and Stein [6] proposed a Monte Carlo algorithm with time bound of $O(n^{2(k-1)} \log^3 n)$.

Several 2-approximation algorithms for the minimum k-way cut problem have been proposed. Saran and Vazirani [10] first proposed a $2(1-1/k)$-approximation algorithm which successively finds minimum cuts until the graph is partitioned into k components and runs in $O(mn^2 \log(n^2/m))$ time. Kapoor [5] gave an $O(k(nm + n^2 \log n))$ time $2(1-1/k)$-approximation algorithm. Zhao *et al.* [12] presented an $O(kmn^3 \log(n^2/m))$ time approximation algorithm that has the performance ratio $2 - 3/k$ for an odd k and $2 - (3k-4)/(k^2-k)$ for an even k. Naor and Rabani [8] showed a 2-approximation algorithm based on an LP relaxation of the minimum k-way cut. Ravi and Sinha [9] gave a 2-approximation algorithm based on an algorithm for computing the strength of graphs.

A minimum k-partition $\mathcal{S}^*$ of V, where $cost(\mathcal{S}^*) = 2\mu(G,k)$ holds, is a k-subpartition of V. Hence

$$opt(V,k) \leqq cost(\mathcal{S}^*) = 2\mu(G,k).$$

Let $\mathcal{S}$ be a minimum k-subpartition of V, which can be found in $O(mn + n^2 \log n)$ time by Theorem 1 with $b = +\infty$. If $\mathcal{S}$ is a k-partition of V, then it is optimal to the minimum k-way cut problem. Otherwise, we choose a subset $X \in \mathcal{S}$ with the maximum $d(X)$, and replace X with subset $V - \cup_{Y \in \mathcal{S} - \{X\}} Y$ in $\mathcal{S}$. For the resulting k-partition $\mathcal{S}'$ of V, the set $F' = \cup_{Y,Y' \in \mathcal{S}'} E(Y, Y')$ satisfies

$$\begin{aligned} w(F') = opt(V,k) - d(X) &\leqq (1 - \frac{1}{k}) opt(V,k) \\ &\leqq 2(1 - \frac{1}{k})\mu(G,k). \end{aligned}$$

Since minimum k-subpartitions for all $k \in [2,n]$ can be obtained in $O(mn + n^2 \log n)$ time by Theorem 1, we have the following result.

[Theorem 5] Given an edge-weighted graph $G = (V,E)$, $2(1-\frac{1}{k})$-approximate solutions to the minimum k-way cut problem for all $k \in [2,n]$ can be obtained in $O(mn + n^2 \log n)$ time. ◇

4.2 Graph Strength

Given an edge-weighted graph G, the *strength* $\sigma(G)$ of G was introduced by Gusfield [3] and Cunningham [1] as a measure of network invulnerability, which is defined as

$$\sigma(G) = \min_{2 \leqq k \leqq n} \left\{ \frac{w(F_k)}{k-1} \mid F_k \text{ is a minimum } k\text{-way cut of } G \right\}.$$

The strength $\sigma(G)$ can be found in $O(mn^2(m + n \log n))$ time [1].

We can compute 2-approximate solution F'_k to the minimum k-way cut problem for all $k \in [2,n]$ in $O(mn + n^2 \log n)$ time by Theorem 5. We choose a solution F'_k that minimizes $w(F'_k)/(k-1)$ among $F'_2, F'_3, \ldots, F'_n$. There is a minimum k^*-way cut F_{k^*} with $w(F_{k^*})/(k^*-1) = \sigma(G)$. We see that $w(F'_k)/(k-1) \leqq w(F'_{k^*})/(k^*-1) \leqq 2w(F_{k^*})/(k^*-1) = 2\sigma(G)$. Hence we have the next result.

[Theorem 6] Given an edge-weighted graph $G = (V,E)$, a partition $\mathcal{S}$ of V such that $\sigma(G) \leqq cost(\mathcal{S})/(|\mathcal{S}|-1) \leqq 2\sigma(G)$ can be determined in $O(mn + n^2 \log n)$ time. ◇

Acknowledgement

This research was partially supported by the Scientific Grant-in-Aid from Ministry of Education, Science, Sports and Culture of Japan.

文　献

[1] W. H. Cunningham, Optimal attack and reinforcement of a network. J. Assoc. Comput. Mach. 32, (1985) 549–561

[2] O. Goldschmidt and D. S. Hochbaum, Polynomial algorithm for the k-cut problem for fixed k, Mathematics of Operation Research, 19 (1994) 24–37.

[3] D. Gusfield, Connectivity and edge-disjoint spanning trees. Inf. Process. Lett. 16, (1983) 87–89

[4] Y. Kamidoi, N. Yoshida and H. Nagamochi, A deterministic algorithm for finding all minimum k-way cuts, 4th Japanese-Hungarian Symposium on Discrete Mathematics and Its Applications, Budapest, Hungary, (2005) 224–233.

[5] S. Kapoor, On minimum 3-cuts and approximating k-cuts using cut trees, Lecture Notes in Computer Science, Springer, 1084 (1996) 132–146.

[6] D. R. Karger and C. Stein, A new approach to the minimum cut problems, J. ACM, 43 (1996) 601–640.

[7] H. Nagamochi, Graph algorithms for network connectivity problems, J. Operations Research Society of Japan, 47 (2004) 199–223.

[8] J. Naor and Y. Rabani, Tree packing and approximating k-cuts, Proc. 14th Annual ACM-SIAM Symposium on Discrete Algorithms (2001) 26–27.

[9] R. Ravi and A. Sinha, Approximating k-cuts via network strength, Proc. 15th Annual ACM-SIAM Symposium on Discrete Algorithms (2002) 621–622.

[10] H. Saran and V. V. Vazirani, Finding k cuts within twice the optimal, SIAM J. Comput., 24 (1995) 101–108.

[11] T. Watanabe and A. Nakamura, Edge-connectivity augmentation problems, J. Comp. System Sci., 35 (1987) 96–144.

[12] L. Zhao, H. Nagamochi and T. Ibaraki, Approximating the minimum k-way cut in a graph via minimum 3-way cuts, J. of Combinatorial Optimization, 5 (2001) 397–410.