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

2004－AL－98　(1)
2004／11／5

グラフの変形操作における単純性の保存

伊藤 大雄
京都大学 大学院情報学研究科 通信情報システム専攻
〒 606-8501 京都市左京区吉田本町
itohiro@i.kyoto-u.ac.jp

概要　ラベル付きグラフ上の半順序関係である、カットサイズ順序、枝長和順序、交叉操作順序の三つは同値であることが知られている。グラフ G がこの半順序関係において G' に先行するとき、G を G' に変形する交叉操作の列が存在するが、仮に G と G' の両方が単純グラフだとしても、中間に単純でないグラフが現れる可能性がある。このことは、Hakimi による名高い次数不変変換においても同様に起こる問題である。本稿では、このような場合に単純グラフのみを介して、変形しうるか否かという問題を取り扱う。結果、Hakimi の次数不変変換については、常に単純グラフによる変形が可能であることを示した。しかし一方で、上記の半順序関係に基づく変形の場合は、例え交叉操作以外の操作も許容したとしても、与えられた操作の種類が有限だとすると、その半順序関係にありながら単純グラフのみを介しての変換ができないグラフの組が必ず存在することを証明した。

On Transformation of Graphs with Preserving their Simpleness

ITO Hiro
Dept. of Communications and Computer Engineering,
School of Informatics, Kyoto University,
Kyoto 606-8501, JAPAN.
itohiro@i.kyoto-u.ac.jp

Abstract　Three partial orders, cut-size order, length order, and operation order, defined between labeled graphs with the same order are known to be equivalent. If G precedes G' with regard to the partial order, there is a sequence of cross-operations such that G is transformed into G' by using the sequence. Even if both G and G' are simple graphs, non-simple graphs may appear on the way of the transformation. The same problem occurs in the well-known Hakimi's d-invariant transformation. This paper considers whether we can transform them without using non-simple graphs. First, we show that we can do it in Hakimi's d-invariant transformation. Second, however, we prove that there is no efficient finite set of operations for the partial order's transformation in the general case.

1 Introduction

Let $N = \{x_0, x_1, \ldots, x_{n-1}\}$ be a set of vertices of a convex polygon P in the plane, where the vertices are arranged in this order counter-clockwisely, and hence (x_i, x_{i+1}) is an edge of P for

$i = 0, 1, \ldots, n-1$ (We adopt the residue class on n for treating integers in N, i.e., $i \pm j$ is $i' \in N$ such that $i' \equiv i \pm j \pmod{n}$). An internal angle of P may be π. We consider graphs whose node set corresponds to N, i.e., the node set is $\{0, 1, \ldots, n-1\}$ and each node i is assigned to x_i, and each edge $e = (i, j)$ of the graph is represented by a line segment $x_i x_j$.

We adopt the cyclic order for treating integers (or numbered vertices) in N. Thus for $i, j \in N$,

$$[i, j] \quad = \quad \begin{cases} \{i,\ i+1,\ \ldots,\ j\}, & \text{if } i \leq j, \\ \{i,\ i+1,\ \ldots,\ n-1,\ 0,\ 1,\ \ldots,\ j\}, & \text{if } i > j; \end{cases}$$

for $i, j, k \in N$, $i \leq j \leq k$ means $j \in [i, k]$; for $i, j, k, h \in N$, $i \leq j \leq k \leq h$ means $i \leq j \leq k$, $j \leq k \leq h$, $k \leq h \leq i$, and $h \leq i \leq j$; and so on. For notational simplicity, $[i, i]$ may be written as $[i]$ or i. For a graph G, $E(G)$ means the edge set of G.

2 Preliminaries

In this section we consider weighted graphs, i.e., we introduce a weight function $w_G : E(G) \to \boldsymbol{R}$. a weighted graph G always has a weight function w_G in this paper. We introduce some terms as follows.

Linear cuts. For a graph G and a pair of distinct nodes $i, j \in N$, a **linear cut** $C_G(i, j)$ is an edge set:

$$C_G(i, j) = \{(k, h) \in E(G) \mid k \in [i, j-1],\ h \in [j, i-1]\}.$$

The capacity of a linear cut $C_G(i, j)$ is defined as

$$c_G(i, j) = \sum_{e \in C_G(i,j)} w_G(e).$$

For two subsets N' and N'' of nodes, $w_G(N', N'') = \sum_{i \in N', j \in N''} w_G(i, j)$. The **degree** of a node $i \in N$ of a graph G is defined as $c_G(i, i+1) = w_G(i, [i+1, i-1])$ and may be simply denoted by $d_G(i)$. We introduce a relation based on sizes of linear cuts as follows. For two weighted graph G and G', $G \preceq_c G'$ means that $c_G(i, j) \leq c_{G'}(i, j)$ for all $i, j \in N$, $i \neq j$. This relation is known to be a partial order, since the following result presented by Skiena [5].

Theorem 1 *For two weighted graphs G and G', if $c_G(i, j) = c_{G'}(i, j)$ for all $i, j \in N$, $i \neq j$, then $G = G'$.* □

Sum of edge lengths. For an edge (i, j) of a weighted graph G and a convex n-gon P, let $\mathrm{dist}(i, j)$ be a length of the line segment $x_i x_j$. We define a sum of weighted edge length of G with respect to P as

$$s_P(G) = \sum_{e \in E(G)} w(e) \cdot \mathrm{dist}(e).$$

We introduce a relation based on the measure as follows. For two weighted graph G and G', $G \preceq_l G'$ means that $s_P(G) \leq s_P(G')$ for all convex n-gons P.

Cross operation. We introduce an operation transforming a graph to another one. For a weighted graph G, two distinct $i, j \in N$ and a real value Δ, $\mathrm{ADD}_G(i, j; \Delta)$ means adding Δ to $w(i, j)$ (if $(i, j) \notin E(G)$, adding a edge (i, j) to $E(G)$ previously). The reverse operation of ADD can be defined, i.e., $\mathrm{REMOVE}_G(i, j; \Delta)$ means $\mathrm{ADD}_G(i, j; -\Delta)$. We extend these operations in the case $i = j$, i.e., both $\mathrm{ADD}_G(i, i; \Delta)$ and $\mathrm{REMOVE}_G(i, i; \Delta)$ mean doing nothing. For nodes $i, j, k, h \in N$ with $i \leq j \leq k \leq h$ and a positive $\Delta > 0$, a **cross operation** $X_G(i, j, k, h; \Delta)$ is applying

$$\text{REMOVE}_G(i,j;\Delta),\ \text{REMOVE}_G(k,h;\Delta),\ \text{ADD}_G(i,k;\Delta),\text{ and } \text{ADD}_G(j,h;\Delta).$$

If some of $\{i,j,k,h\}$ are equal, a cross operation may increase edges. In fact, if $i = j < k < h < i$ or $i = j < k = h < i$ (or the cases symmetric with respect to one of them), then the total edge weights increases. If $j = k$ or $i = h$, the edge set is not changed. We introduce a relation based on cross operations as follows. For two weighted graph G and G', $G \preceq_o G'$ means that G' can be obtained from G by applying finite number (including zero) of cross operations.

Equivalence of the three relations. We have the following theorem.

Theorem 2 *Three relations $\preceq_c$, $\preceq_l$, and $\preceq_o$ are equivalent.* □

This theorem was shown in [4] for graphs with the same size (number of edges), and for the general case in [3].

Corollary 1 *Three relations $\preceq_c$, $\preceq_l$, and $\preceq_o$ are all partial orders.*

Proof: Clear from Theorem 2 and that $\preceq_c$ is a partial order. □

From Theorem 2, these three partial orders can be denoted by $\preceq$ simply.

3 Preserving their simpleness

A cross operation sometimes violates the simpleness of graphs: Let G and G' be a pair of weighted graphs with $G \preceq G'$. This means there is a sequence of weighted graphs

$$G = G_0 \preceq G_1 \preceq \cdots \preceq G_k = G',$$

such that G_i is obtained by applying a cross operation on G_{i-1} for $i = 1, 2, \ldots, k$. Even if G and G' are both simple (all weights are 1 or 0), there may be a nonsimple graph on the sequence.

3.1 Hakimi's d-invariant transformations

Cross operation is very similar to Hakimi's d-invariant transformation [1, 2]. Let G be a graph and (i,j) and (k,h) are two distinct edges (i,j,k,h are all distinct) of G. A d-invariant transformation of G is removing (i,j) and (k,h) from G and adding two edges (i,k) and (j,h). Note that if (i,k) or (j,h) have existed in the original G, the new graph becomes including parallel edges. Hakimi showed that all two graphs G and G' with the same node set N and $d_G(i) = d_{G'}(i)$ for all $i \in N$ can be transformed each other by using a finite sequence of d-invariant transformations.

As mentioned above, however, there may be a nonsimple graph in the sequence even if G and G' are both simple. In fact, if we use the operation shown in Hakimi's proof, we can easily construct such examples. For this problem, we found that simple graphs are enough for transforming any pair of simple graphs as shown in the following theorem.

Theorem 3 *For a pair of simple graphs G and G' with the same node set N and $d_G(i) = d_{G'}(i)$ for all $i \in N$, there is a sequence which includes only simple graphs of d-invariant transformations transforming G into G'.* □

For proving this theorem we prepare a lemma. Let E and E' be the edge set of G and G', respectively. We define an edge colored complete graph G^* representing the symmetric difference between G and G' as follows. An edge (i,j) of G is colored

- red if $(i,j) \in E$ and $(i,j) \notin E'$,

- blue if $(i,j) \notin E$ and $(i,j) \in E'$,
- black if $(i,j) \in E$ and $(i,j) \in E'$,
- white if $(i,j) \notin E$ and $(i,j) \notin E'$.

A **red-blue alternating cycle** is a cycle such that it consists of red and blue edges only and a red edge and a blue edge appears alternatively by traversing the cycle. The length of a red-blue alternating cycle is clearly even.

Lemma 1 *If $G \neq G'$, then G^* has a red-blue alternating cycle*

$$\langle i_0, i_1, \ldots, i_k, j_k, j_{k-1}, \ldots, j_0, i_0 \rangle$$

such that $i_0 \neq j_0$, $i_1 \neq j_1$, $\ldots$, $i_{k-1} \neq j_{k-1}$.

Proof: If $G \neq G'$, there exists at least one red edge and one blue edge. For each node, the number of red edges incident to the node is equal to the one of blue edges incident to the node. Thus, by traversing red edges and blue edges alternatively, a red-blue alternating cycle is found. Let C be a red-blue alternating cycle with the shortest length in G^*. We show that C must be a desired one. We express C as $\langle c_0, c_1, \ldots, c_{2k-1}, c_0 \rangle$. First, we label c_0 by i_0, c_1 by i_1, and so on. If there is no $h \in \{1, \ldots, k-1\}$ such that $i_h = j_h$, C is a desired one, and hence we assume that there is at least one $h \in \{1, \ldots, k-1\}$ such that $i_h = j_h$. Then we find a pair of the same nodes. Second, we label c_1 by i_0, c_2 by i_1, and so on. From the same discussion, we can assume that there is at least one $h \in \{1, \ldots, k-1\}$ such that $i_h = j_h$, and we find a pair of the same nodes in this case also. Note that this pair is a different from the above one. This discussion can be used until c_{k-1} is labeled by i_0. Then we can get k distinct pairs. We express the pairs by chords of C. We have k distinct chords and $2k$ nodes. From this we can see that there is at least one pair of crossing chords. Let (c_i, c_k) and (c_j, c_h) be the crossing chords (c_i, c_j, c_k, c_h appear in this order when we traverse C). We obtain four path by cutting C at c_i, c_j, c_k, and c_h. We denote them as P_{ij}, P_{jk}, P_{kh}, P_{hi} (P_{ij} is a path between c_i and c_j, and so on). If path $P_{ij} \cup P_{jk}$ (or $P_{jk} \cup P_{kh}$) has even length, then it constructs a shorter red-blue alternating cycle, contradiction. Then we assume that both $P_{ij} \cup P_{jk}$ and $P_{jk} \cup P_{kh}$ have odd length. Hence $P_{ij} \cup P_{kh}$ has an even length, and we can easily observe that it constructs a shorter red-blue alternating cycle, contradiction. $\square$.

Proof of Theorem 3: E_r and E_b denote the set of red edges and blue edges, respectively. If $E_r \cup E_b = \emptyset$, $G = G'$. Then we suppose that $E_r \cup E_b \neq \emptyset$. We will outline a process to find a d-invariant transformation which preserve simpleness of the graph and deletes the size of $E_r \cup E_b$. By applying the process iteratively, we can finally find the desired sequence of d-invariant transformations.

Let C be a red-blue alternating cycle satisfying the condition of Lemma 1, whose existence is guaranteed by the lemma. We can suppose edge (i_0, j_0) is red without loss of generality (If it is blue, we can change the name of red and blue each other). Hence (i_0, i_1) and (j_0, j_1) are blue. If (i_1, j_1) is red or black, we find a desired d-invariant transformation, which consists of removing (i_0, j_0) and (i_1, j_1) from G and adding (i_0, i_1) and (j_0, j_1) to G. Then we assume (i_1, j_1) is blue or white. Here, if (i_2, j_2) is blue or white, we find a desired d-invariant transformation, which consists of adding (i_1, j_1) and (i_2, j_2) from G and removing (i_1, i_2) and (j_1, j_2) to G. Then we assume (i_1, j_1) is red or black. We apply the above discussion iteratively, until we find a desired d-invariant transformation. Assume that we can't find such a transformation until we reached onother side of C. This results in that

(i_0, j_0), (i_2, j_2), (i_4, j_4), ... are red or black, and

(i_1, j_1), (i_3, j_3), (i_5, j_5), ... are blue or white.

However (i_k, j_k) is blue if k is odd, and red otherwise from the definition of red-blue alternating cycle (note that we assume (i_0, j_0) is red), contradiction. Therefore we can find a desired d-invariant transformation in any case. □

3.2 Cross operations

Here we return to the subject on cross operations. We have a conjecture that simple graphs are enough in this case too.

Conjecture 1 *For a pair of simple graphs G and G' with the same node set N such that $d_G(i) = d_{G'}(i)$ for all $i \in N$ and $G \preceq G'$, there is a sequence which includes only simple graphs of cross operations transforming G into G'.* □

The above conjecture considers a pair of graphs with the same number of edges. However, Theorem 2 assures us cross operations can transform any graph $G = (N, E)$ to any other graph $G' = (N, E')$ as long as $G \preceq G'$ even if $|E| < |E'|$. Here we reached to another question: does Conjecture 1 hold even if the degree condition is deleted?

To this question, we have a simple counter example shown in Fig. 1. If we don't mind breaking the simpleness, there is a sequence $X(0, 0, 1, 3; 0.5)$, $X(2, 2, 3, 1; 0.5)$, $X(1, 1, 2, 0; 0.5)$, $X(3, 3, 0, 2; 0.5)$. However there is no efficient cross operation that doesn't break the simpleness. Thus we should to introduce other operations to complete transformations in general case. The example of Fig. 1 can be solved if we introduce a new operation consisting of REMOVE(i, k), REMOVE(j, h), ADD(i, j), ADD(j, k), ADD(k, h), and ADD(h, i), for $i \le j \le k \le h$. Precisely, it may not be sufficient for general case. If we find another counter example, which can't be solved yet, then we should add another new operation.

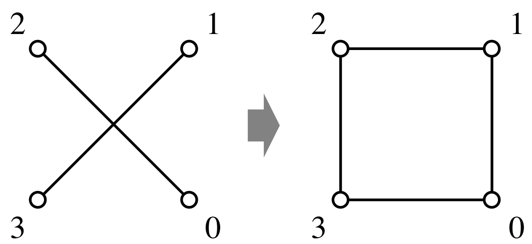

Figure 1: Counter example.

Then we reach to another question: is there a finite set of operations that transforms general pair of simple graphs without breaking simpleness of the graphs? For treating this question, we must define what is a "operation"? The following definition seems appropriate.

An **operation** is a set of finite number of ADDs and REMOVEs such that it doesn't decrease the size of any linear cut.

For example, cross-operation consists of two ADDs and two REMOVEs, and it doesn't decrease the size of any linear cut. The operation introduced for solving the example of Fig. 1 consists of four ADDs and two REMOVEs, and it doesn't decrease the size of any linear cut also. Thus they satisfy the above definition.

Let O be a finite set of operations and let G and G' be simple graphs with $G \preceq G'$. A **simple transformation sequence** of O for G and G' is a set of operations of O such that it transforms G into G' and every graph appeared in the process of the transformation is simple. We obtained the following theorem.

Theorem 4 *There is no finite set of operations such that there is a simple transformation sequence of the set for any pair of simple graphs G and G' with $G \preceq G'$.*

For assuming this theorem, we define an even-complete graph K_n^e and an odd-complete graph K_n^o of even order n as follows.

$$\begin{aligned} K_n^e &= (N, E_n^e = \{(i,j) \mid i,j \in N, |i-j| \text{ is even}\}), \\ K_n^o &= (N, E_n^o = \{(i,j) \mid i,j \in N, |i-j| \text{ is odd}\}). \end{aligned}$$

For any even n, they are well defined. By comparing the sizes of linear-cuts on them, we see that

$$c_{K_n^e}(i,j) < c_{K_n^o}(i,j) \quad \text{if } |i-j| \text{ is odd, and} \tag{1}$$
$$c_{K_n^e}(i,j) = c_{K_n^o}(i,j) \quad \text{if } |i-j| \text{ is even.} \tag{2}$$

Hence,

$$K_n^e \prec K_n^o.$$

We will show that K_n^e can't be transformed into K_n^o if n is enough large as follows.

Proof of Theorem 4: Suppose otherwise, i.e., there is a finite set of operations O satisfying the condition of the theorem. For an operation o in O, the **order** of o is the number of nodes such that one or more edges incident to the node are added or removed by o. For example, the order of cross-operation is four.

Let k^* be the maximum order of the operations in O. Let $n \geq 2k$ be an even number. We will prove that any operation in O can't transform K_n^e to another graph G such that $K_n^e \prec G \preceq K_n^o$. (This is enough to prove the theorem.)

Assume otherwise, i.e., there is an operation $o \in O$ that transform K_n^e to another graph G such that $K_n^e \prec G \preceq K_n^o$. We construct a $\{-1,1\}$-weighted graph G_o as follows. G_o has a node set N, an edge set $E_o = \{(i,j) \mid \text{ADD}(i,j) \text{ or REMOVE}(i,j) \text{ is in } o\}$ and a weight function w_o such that $w_o(i,j) = 1$ if $\text{ADD}(i,j) \in o$ and $w_o(i,j) = -1$ if $\text{REMOVE}(i,j) \in o$.

We can easily see that the edge set of G is

$$E = \{(i,j) \mid w_{E_n^e}(i,j) + w_o(i,j) = 1\}.$$

(Remember that the weights of a simple graphs are defined as 1 if there is an edge and 0 otherwise.) From $G \preceq K_n^o$, every linear-cut on G_1 has the size zero or one.

Let $i_0, i_1, \ldots, i_{k-1}$ ($i_0 < i_1 < \cdots < i_{k-1}$) be the nodes which have at least one edge incident to the nodes in G_o, where k is the order of o. Since $k \leq n/2$, we can assume $i_0 - i_{k-1} \geq 2$ w.l.o.g. (Note that the residue class is used for the difference.) Thus for any $j \in \{1, \ldots, k-1\}$ there is a linear-cut separating $i_0, \ldots, i_{j-1}$ and $i_j, \ldots, i_{k-1}$ such that the size on both K_n^e and K_n^o are the same, and hence, we see that the size of these linear-cuts on G_o is zero, i.e.,

$$c_{G_o}(i_0, i_j) = 0 \quad \text{for every } j \in \{1, \ldots, k-1\}. \tag{3}$$

Thus

$$\begin{aligned} 0 &= c_{G_o}(i_0, i_1) \\ &= w_{G_o}(i_0, i_1) + w_{G_1}(i_0, [i_2, i_{k-1}]), \\ 0 &= c_{G_o}(i_0, i_2) \\ &= w_{G_o}(i_1, [i_2, i_{k-1}]) + w_{G_o}(i_0, [i_2, i_{k-1}]). \end{aligned}$$

Therefore,

$$w_{G_o}(i_0, i_1) = w_{G_o}(i_1, [i_2, i_{k-1}]). \tag{4}$$

Moreover,

$$\begin{aligned} 0 &\le c_{G_o}(i_1, i_2) \\ &= w_{G_o}(i_0, i_1) + w_{G_o}(i_1, [i_2, i_{k-1}]) \\ &\le 1. \end{aligned} \tag{5}$$

From (4), (5), and that all edge weights are integers, we obtain

$$w_{G_o}(i_0, i_1) = w_{G_o}(i_1, [i_2, i_{k-1}]) = 0. \tag{6}$$

For an assumption for an induction, we assume that

$$w_{G_o}(i, j) = 0 \quad \text{for all } i, j \in \{i_0, \dots, i_q\}. \tag{7}$$

By using this assumption, we will derive the result such that

$$w_{G_o}(i, j) = 0 \quad \text{for all } i, j \in \{i_0, \dots, i_{q+1}\}$$

as follows.

$$\begin{aligned} a_h &= w_{G_o}(i_h, i_{h+1}) \quad \text{for } h = 0, \dots, q, \\ A_h &= w_{G_o}(i_h, [i_{q+2}, i_{k_1-1}]) \\ &\qquad \text{for } h = 0, \dots, q+1. \end{aligned}$$

From (7),

$$\begin{aligned} 0 &= c_{G_o}(i_0, i_1) = a_0 + A_0, \\ 0 &= c_{G_o}(i_0, i_2) = a_0 + a_1 + A_0 + A_1 \\ &= a_1 + A_1, \\ 0 &= c_{G_o}(i_0, i_3) \\ &= a_0 + a_1 + a_2 + A_0 + A_1 + A_2 \\ &= a_2 + A_2, \\ &\vdots \end{aligned}$$

Generally, we have

$$0 = a_h + A_h \quad \text{for } h = 0, \dots, q. \tag{8}$$

By summing them from $h = 1$ to q,

$$0 = \sum_{h=0}^{q} (a_h + A_h) \tag{9}$$

Moreover,

$$0 = c_{G_o}(i_0, i_{q+2}) = \sum_{h=0}^{q+1} A_h. \tag{10}$$

By comparing (9) and (10), we have

$$\sum_{h=0}^{q} a_h = A_{q+1} \tag{11}$$

The capacity of any linear-cut of G_1 is between zero and one, and hence

$$0 \le c_{G_o}(i_{q+1}, i_{q+2}) = \sum_{h=0}^{q} a_h + A_{h+1} \le 1. \tag{12}$$

From (11), (12), and that all edge weights are integers, we see

$$\sum_{h=0}^{q} a_h = A_{q+1} = 0. \tag{13}$$

From $\sum_{h=0}^{q} a_h = 0$ (13) and $0 = a_h + A_h$ (8), we get

$$\sum_{h=0}^{q-1} a_h = A_q. \tag{14}$$

Moreover,

$$0 \le c_{G_o}(i_q, i_{q+2}) = \sum_{h=0}^{q-1} a_h + A_q + A_{q+1} \le 1.$$

Since $A_{q+1} = 0$ (13),

$$0 \le \sum_{h=0}^{q-1} a_h + A_q \le 1. \tag{15}$$

From (14), (15), and that all edge weights are integers, we see

$$\sum_{h=0}^{q-1} a_h = A_q = 0.$$

By using similar discussions, we finally obtain the following equation.

$$A_q = A_{q-1} = \cdots = A_0 = 0.$$

From this and (8), we see

$$a_q = a_{q-1} = \cdots = a_0 = 0.$$

That is

$$w_{G_o}(i, j) = 0 \quad \text{for all } i, j \in \{i_0, \dots, i_{q+1}\}.$$

By induction, we conclude that there is no edge between any pair of nodes in $\{i_0, i_1, \dots, i_{k-1}\}$, i.e., o consists of no ADD and REMOVE, contradiction. □

4 Concluding remarks

First, this paper presents a simple proof for the equivalence of the three partial orders defined between labeled graphs. Next, it considers graph transformation preserving simpleness. For this problem, we show that there always be such a transformation for Hakinmi's d-invariant transformation. However, we proved that there is no efficient finite set of operations for the partial order's transformation in the general case.

If the number of edges of the graphs are the same, we conjecture that simple transformation sequences of cross-operations are always exists (Conjecture 1). It remains for future research.

References

[1] S. L. Hakimi, On realizability of a set of integers as degrees of the vertices of a linear graph. I, J. Soc. Indust. Appl. Math., **10** (1962) 496–506.

[2] S. L. Hakimi, On realizability of a set of integers as degrees of the vertices of a linear graph II. Uniqueness, J. Soc. Indust. Appl. Math., **11** (1963) 135–147.

[3] H. Ito, Relation among edge length of convex planar drawings, size of linear cuts, and cross-operations on graphs, IPSJ SIG Notes, **2002**, 29 (2002) 27–34.

[4] H. Ito, Sum of edge lengths of a multigraph drawn on a convex polygon, Computational Geometry, **24** (2003) 41–47.

[5] S. S. Skiena, Reconstructing graphs from cut-set sizes, Information Processing Letters, **32** (1989) 123–127.