

# Bisecting a Four-Connected Graph with Three Resource Sets

Toshimasa Ishii <sup>a,\*</sup>, Kengo Iwata <sup>b</sup>, Hiroshi Nagamochi <sup>c</sup>,

<sup>a</sup>*Department of Information and Management Science,  
Otaru University of Commerce, Hokkaido, 047-8501, Japan*

<sup>b</sup>*Mazda Motor Corporation, Hiroshima 730-8670, Japan*

<sup>c</sup>*Department of Applied Mathematics and Physics,  
Graduate School of Informatics, Kyoto University, Kyoto 606-8501, Japan*

---

## Abstract

Let  $G = (V, E)$  be an undirected graph with a node set  $V$  and an arc set  $E$ .  $G$  has  $k$  pairwise disjoint subsets  $T_1, T_2, \dots, T_k$  of nodes, called resource sets, where  $|T_i|$  is even for each  $i$ . The partition problem with  $k$  resource sets asks to find a partition  $V_1$  and  $V_2$  of the node set  $V$  such that the graphs induced by  $V_1$  and  $V_2$  are both connected and  $|V_1 \cap T_i| = |V_2 \cap T_i| = |T_i|/2$  holds for each  $i = 1, 2, \dots, k$ . The problem of testing whether such a bisection exists is known to be NP-hard even in the case of  $k = 1$ . On the other hand, it is known that if  $G$  is  $(k + 1)$ -connected for  $k = 1, 2$ , then a bisection exists for any given resource sets, and it has been conjectured that for  $k \geq 3$ , a  $(k + 1)$ -connected graph admits a bisection. In this paper, we show that for  $k = 3$ , the conjecture does not hold, while if  $G$  is 4-connected and has  $K_4$  as its subgraph, then a bisection exists and it can be found in  $O(|V|^3 \log |V|)$  time. Moreover, we show that for an arc-version of the problem, the  $(k + 1)$ -edge-connectivity suffices for  $k = 1, 2, 3$ .

*Key words:* graph algorithm, graph partition problem, graph connectivity, embedding

---

<sup>\*</sup> Corresponding author.

*Email addresses:* [ishii@res.otaru-uc.ac.jp](mailto:ishii@res.otaru-uc.ac.jp) (Toshimasa Ishii),  
[nag@i.kyoto-u.ac.jp](mailto:nag@i.kyoto-u.ac.jp) (Hiroshi Nagamochi).

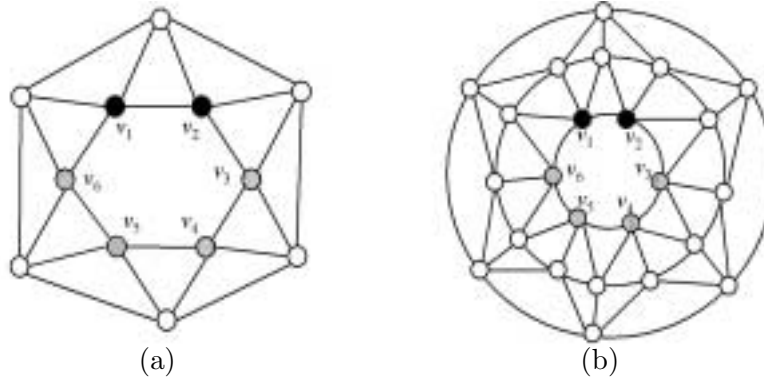


Fig. 1. Illustration of instances of 4-connected graphs which have no 3-bisection, where  $T_1 = \{v_1, v_2\}$ ,  $T_2 = \{v_3, v_4\}$ , and  $T_3 = \{v_5, v_6\}$  in both (a) and (b). Note that the graph (b) is also 5-connected.

## 1 Introduction

In this paper, we consider the following graph partition problems: given an undirected graph  $G = (V, E)$  with a set  $V$  of nodes a set  $E$  of arcs, and  $k$  pairwise disjoint sets  $T_1, T_2, \dots, T_k$  of nodes, called *resource sets*, where each  $|T_i|$  is even, find a partition  $V_1$  and  $V_2$  of  $V$  such that the graphs induced by  $V_1$  and  $V_2$  are both connected and  $|V_1 \cap T_i| = |V_2 \cap T_i| = |T_i|/2$  holds for each  $i = 1, 2, \dots, k$ . This problem is called *the bisection problems with  $k$  resource sets*, and such a bisection is called  *$k$ -bisection (with respect to  $T_1, \dots, T_k$ )*.

Such a problem of partitioning a graph into connected subgraphs under fair-division type of constraints appears in many applications such as political districting [1,7,15], the paging system in operating systems [13] and the image processing [6]. For the political districting, a dual graph of the map which consists of regions is required to be divided into connected subgraphs, each of which represents an electoral zone, so that both the area and the number of voters in each zone is balanced over all zones.

So far, for general graphs, the problem was shown to be NP-hard even if  $k = 1$  holds, since it is NP-hard to test whether a 1-bisection exists or not [3,4]. On the other hand, when  $k = 1, 2$ , it was shown that if a given graph is  $(k + 1)$ -connected, then for any given resource sets, such a  $k$ -bisection exists and it can be found in linear time for  $k = 1$  by Suzuki et al. [12] and by Wada and Kawaguchi [14], and in  $O(|V|^2 \log |V|)$  time for  $k = 2$  by Nagamochi et al. [11]. For a general  $k \geq 3$ , to our knowledge, any nontrivial sufficient condition for which a  $k$ -bisection exists is not known, while it was conjectured in [11] that every  $(k + 1)$ -connected graph admits a  $k$ -bisection.

In this paper, we consider the case of  $k = 3$ . We first show that there exist 4-connected graphs which have no 3-bisection, as shown in Figure 1. This indicates a negative answer to the above conjecture for  $k = 3$ . In particular, the

graph in Figure 1(b) is also 5-connected, which shows that even 5-connected graphs may have no 3-bisection (this also indicates a negative answer to the above conjecture for  $k = 4$ ). Instead, in this paper, we give a sufficient condition for which a 3-bisection exists; we prove that if  $G$  is 4-connected and has a complete graph  $K_4$  of four nodes as its subgraph, then a 3-bisection exists. We also show that it can be found in  $O(|V|^3 \log |V|)$  time. A key technique of the proof, which is an extension of the method by Nagamochi et al. [11], is a reduction of the problem to a geometrical problem. We first prove that every 4-connected graph containing a complete graph  $K^*$  of four nodes as its subgraph can be embedded in the 3-dimensional space  $\mathfrak{R}^3$ , in such a way that the following (i)(ii) hold: (i) the convex hull of its nodes is a trigonal pyramid corresponding to the  $K^*$ , (ii) every node not in  $K^*$  is in the convex hull of its neighbors (precise definition is given in Section 2.2). This will guarantee that, for any given plane  $H$  in  $\mathfrak{R}^3$ , each of the two subgraphs of  $G$  separated by  $H$  remains connected. Given such an embedding in  $\mathfrak{R}^3$ , we apply the so-called ham-sandwich cut algorithm, which is well known in computational geometry, to find a plane  $H^*$  that bisects  $T_1, T_2$ , and  $T_3$  simultaneously. Consequently, the two subgraphs by the plane  $H^*$  indicates a 3-bisection. We give an algorithm for finding such a plane  $H^*$  in  $O(|V|^3 \log |V|)$  time.

Moreover, we consider an arc-version of the bisection problem; given an undirected graph  $G = (V, E)$  and  $k$  pairwise disjoint sets  $T_1, \dots, T_k$  of arcs, where each  $|T_i|$  is even, find a partition  $E_1$  and  $E_2$  of  $E$  such that the graphs induced by  $E_1$  and  $E_2$  are both connected and  $|E_1 \cap T_i| = |E_2 \cap T_i| = |T_i|/2$  holds for each  $i = 1, \dots, k$ . We call such a bisection *k-bisection of the arc set*. For this problem, we show that a  $(k + 1)$ -edge-connected graph admits a bisection for any given resource sets for  $k = 1, 2, 3$ .

The paper is organized as follows. Some definitions and preliminaries are described in Section 2. Section 3 describes an algorithm for finding a 3-bisection in a 4-connected graph with  $K_4$ , and Section 4 gives a proof for its correctness. In Section 5, we make some remarks on the problem for a general  $k$  and discuss the arc-version of the problem.

## 2 Preliminaries

Let  $G = (V, E)$  stand for an undirected simple graph with a set  $V$  of *nodes* and a set  $E$  of *arcs*, where we denote  $|V|$  by  $n$  and  $|E|$  by  $m$ . A singleton set  $\{x\}$  may be simply written as  $x$ , and “ $\subset$ ” implies proper inclusion while “ $\subseteq$ ” means “ $\subset$ ” or “ $=$ ”. For a subgraph  $G'$  of  $G$ , the sets of nodes and arcs in  $G'$  are denoted by  $V(G')$  and  $E(G')$ , respectively. For a set  $X$  of nodes in  $G$ , a node  $v \in V - X$  is called a *neighbor* of  $X$  if it is adjacent to some node in  $X$ , and the set of all neighbors of  $X$  is denoted by  $N_G(X)$ .

For an arc  $e = (u, v)$ , we denote by  $G/e$  the graph obtained from  $G$  by contracting  $u$  and  $v$  into a single node (deleting any resulted self-loop), and by  $G - e$  the graph obtained from  $G$  by removing  $e$ . We also say that  $G/e$  is obtained from  $G$  by contracting the arc  $e$ . A graph  $G$  is  $k$ -connected if and only if  $|V| \geq k + 1$  and the graph  $G - X$  obtained from  $G$  by removing any set  $X$  of  $(k - 1)$  nodes remains connected. A graph  $G$  is  $k$ -edge-connected if and only if the graph  $G - F$  obtained from  $G$  by removing any set  $F$  of  $(k - 1)$  arcs remains connected.

The main results of this paper are described as follows.

**Theorem 1** *Let  $G = (V, E)$  be a 4-connected graph which contains a complete graph with four nodes as its subgraph. Let  $T_1, T_2, T_3$  be pairwise disjoint subsets of  $V$  such that  $|T_i|$  is even for  $i = 1, 2, 3$ . Then  $G$  has a 3-bisection with respect to  $T_1, T_2$ , and  $T_3$ , and it can be found in  $O(n^3 \log n)$  time.  $\square$*

**Theorem 2** *Let  $G = (V, E)$  be a  $(k + 1)$ -edge-connected graph with pairwise disjoint subsets  $T_i, i = 1, \dots, k$  of  $E$  such that each  $|T_i|$  is even. If  $k = 1, 2, 3$ , then a  $k$ -bisection of the arc set exists.  $\square$*

In the sequel, we first give a constructive proof of Theorem 1 by reducing the problem to a geometrical problem as mentioned in Section 1. For this, we give some geometric notations in the next two subsections.

### 2.1 Convex hull and ham-sandwich cut

Consider the  $d$ -dimensional space  $\mathfrak{R}^d$ . For a non-zero  $a \in \mathfrak{R}^d$  and a real  $b \in \mathfrak{R}^1$ ,  $H(a, b) = \{x \in \mathfrak{R}^d \mid \langle a \cdot x \rangle = b\}$  is called a *hyperplane*, where  $\langle a \cdot x \rangle$  denotes the inner product of  $a, x \in \mathfrak{R}^d$ . Moreover,  $H^+(a, b) = \{x \in \mathfrak{R}^d \mid \langle a \cdot x \rangle \geq b\}$  (resp.,  $H^-(a, b) = \{x \in \mathfrak{R}^d \mid \langle a \cdot x \rangle \leq b\}$ ) is called a *positive closed half space* (resp., *negative closed half space*) with respect to  $H = H(a, b)$ .

For a set  $P = \{x_1, \dots, x_\ell\}$  of points in  $\mathfrak{R}^d$ , a point  $x' = \alpha_1 x_1 + \dots + \alpha_\ell x_\ell$  with  $\sum_{i=1, \dots, \ell} \alpha_i = 1$  and  $\alpha_i \geq 0, i = 1, \dots, \ell$  is called a *convex combination* of  $P$ , and the set of all convex combinations of  $P$  is denoted by  $\text{conv}(P)$ . If  $P = \{x_1, x_2\}$ , then  $\text{conv}(P)$  is called a *segment* (connecting  $x_1$  and  $x_2$ ), denoted by  $[x_1, x_2]$ . A subset  $S \subseteq \mathfrak{R}^d$  is called a *convex set* if  $[x, x'] \subseteq S$  for any two points  $x, x' \in S$ . For a convex set  $S \subseteq \mathfrak{R}^d$ , a point  $x \in S$  is called a *vertex* if there is no pair of points  $x', x'' \in S - x$  such that  $x \in [x', x'']$ . For two vertices  $x_1, x_2 \in S$ , the segment  $[x_1, x_2]$  is called an *edge* of  $S$  if  $\alpha x' + (1 - \alpha)x'' = x \in [x_1, x_2]$  for some  $0 \leq \alpha \leq 1$  implies  $x', x'' \in [x_1, x_2]$ . The intersection  $S$  of a finite number of closed half spaces is called a *convex polyhedron*, and is called a *convex polytope* if  $S$  is non-empty and bounded.

Given a convex polytope  $S$  in  $\mathfrak{R}^d$ , the *vertex-edge graph*  $G_S = (V_S, E_S)$  is defined to be an undirected graph with node set  $V_S$  corresponding to the vertices of  $S$  and arc set  $E_S$  corresponding to those pairs of vertices  $x, x'$  for which  $[x, x']$  is an edge of  $S$ . For a convex polyhedron  $S$ , a hyperplane  $H(a, b)$  is called a *supporting hyperplane* of  $S$  if  $H(a, b) \cap S \neq \emptyset$  and either  $S \subseteq H^+(a, b)$  or  $S \subseteq H^-(a, b)$ . We say that a point  $p \in S$  is *strictly inside*  $S$  if there is no supporting hyperplane of  $S$  containing  $p$ . If  $S$  has a point strictly inside  $S$  in  $\mathfrak{R}^d$ ,  $S$  is called *full-dimensional* in  $\mathfrak{R}^d$ . The set of points strictly inside  $\text{conv}(P)$  is denoted by  $\text{int}(\text{conv}(P))$ .

Let  $P_1, \dots, P_d$  be  $d$  sets of points in  $\mathfrak{R}^d$ . We say that a hyperplane  $H = H(a, b)$  in  $\mathfrak{R}^d$  *bisects*  $P_i$  if  $|P_i \cap H^+(a, b)| \geq \lceil |P_i|/2 \rceil$  and  $|P_i \cap H^-(a, b)| \geq \lceil |P_i|/2 \rceil$  hold. Thus if  $|P_i|$  is odd, then any bisector  $H$  of  $P_i$  contains at least one point of  $P_i$ . If  $H$  bisects  $P_i$  for each  $i = 1, \dots, d$ , then  $H$  is called a *ham-sandwich cut* with respect to  $P_1, \dots, P_d$ . The following result is well-known (see, e.g., [5]).

**Theorem 3** *Given  $d$  sets  $P_1, \dots, P_d$  of points in the  $d$ -dimensional space  $\mathfrak{R}^d$ , there exists a hyperplane which is a ham-sandwich cut with respect to the sets  $P_1, \dots, P_d$ .  $\square$*

In [2], Chi-Yuan et al. showed that a ham-sandwich cut with respect to given sets  $P_1, P_2, \dots, P_d$  of points in  $\mathfrak{R}^d$  with  $\sum_{i=1}^d |P_i| = p$  can be found in  $O(p^{3/2})$  time for  $d = 3$ ,  $O(p^{8/3})$  time for  $d = 4$ , and  $O(p^{d-1-a(d)})$  time with certain small constant  $a(d) > 0$  for  $d \geq 5$ .

## 2.2 Convex embedding of a graph

In this section, we introduce a strictly convex embedding of a graph in  $\mathfrak{R}^d$ , which was first defined by Nagamochi et al. [11].

Given a graph  $G = (V, E)$ , an *embedding* of  $G$  in  $\mathfrak{R}^d$  is a mapping  $f : V \rightarrow \mathfrak{R}^d$ , where each node  $v$  is represented by a point  $f(v) \in \mathfrak{R}^d$ , and each arc  $e = (u, v)$  by a segment  $[f(u), f(v)]$ , which may be written by  $f(e)$ . For two arcs  $e, e' \in E$ , segments  $f(e)$  and  $f(e')$  may cross each other. For a set  $\{v_1, \dots, v_p\} = Y \subseteq V$  of nodes, we denote by  $f(Y)$  the set  $\{f(v_1), \dots, f(v_p)\}$  of points, and we denote  $\text{conv}(f(Y))$  by  $\text{conv}_f(Y)$ .

A strictly convex embedding of a graph is defined as follows (see Figure 2).

**Definition 4** [11] *Let  $G = (V, E)$  be a graph without isolated nodes and let  $G' = (V', E')$  be a subgraph of  $G$ . A strictly convex embedding (or SC-embedding, for short) of  $G$  with boundary  $G'$  is an embedding  $f$  of  $G$  into  $\mathfrak{R}^d$  in such a way that*

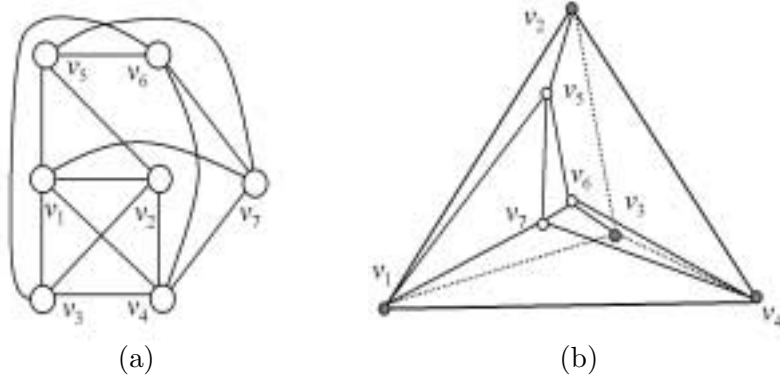


Fig. 2. Illustration of an instance of an SC-embedding; (b) shows an SC-embedding of the graph in (a) with boundary  $(\{v_1, v_2, v_3, v_4\}, \bigcup_{1 \leq i, j \leq 4} (v_i, v_j))$  into  $\mathbb{R}^3$ .

(i) *the vertex-edge graph of the full-dimensional convex polytope  $\text{conv}_f(V')$  is isomorphic to  $G'$  (such that  $f$  itself defines an isomorphism),*

(ii)  *$f(v) \in \text{int}(\text{conv}_f(N_G(v)))$  holds for all nodes  $v \in V - V'$ ,*

(iii) *the points of  $\{f(v) \mid v \in V\}$  are in general position.  $\square$*

From this definition, we can see that the vertices of  $\text{conv}_f(V)$  are precisely the points in the boundary  $f(V')$ .

The following lemma implies that given an SC-embedding of  $G = (V, E)$  into  $\mathbb{R}^d$ , each two sets of nodes obtained by bisecting  $f(V)$  with an arbitrary hyperplane in  $\mathbb{R}^d$  induce connected graphs.

**Lemma 5** [11, Lemma 4.2] *Let  $G = (V, E)$  be a graph without isolated nodes and let  $f$  be an SC-embedding of  $G$  into  $\mathbb{R}^d$ . Let  $f(V_1) \subseteq H^+(a, b)$  and  $f(V) \cap (H^+(a, b) - H(a, b)) \subseteq f(V_1)$  hold for some hyperplane  $H = H(a, b)$  and for some  $\emptyset \neq V_1 \subseteq V$ . Then  $V_1$  induces a connected graph.  $\square$*

By Theorem 3 and this lemma, if there is an SC-embedding of a given graph  $G = (V, E)$  into  $\mathbb{R}^k$ , then by bisecting the embedded graph with a hyperplane which is a ham-sandwich cut with respect to  $T_1, \dots, T_k$ , we can obtain a partition  $V_1$  and  $V_2$  of  $V$  bisecting each  $T_i$  such that each  $V_j$  induces a connected graph, that is, a  $k$ -bisection. Based on this observation, we give an algorithm for finding a 3-bisection in the next section.

### 3 Algorithm for bisecting resource sets

In this section, we give an algorithm, named BISECT3 for finding a 3-bisection in a 4-connected graph with  $K_4$  in  $O(n^3 \log n)$  time, which proves Theorem 1.

#### Algorithm BISECT3

**Input:** A 4-connected graph  $G = (V, E)$  which has a complete graph  $K$  with 4 nodes, and three pairwise disjoint node sets  $T_1, T_2$ , and  $T_3$  where each  $|T_i|$  is even.

**Output:** A 3-bisection of  $G$  with respect to  $T_1, T_2$ , and  $T_3$ .

**Phase 1:** Find an SC-embedding  $f$  of  $G$  with boundary  $K$  into  $\mathfrak{R}^3$ .

**Phase 2:** By applying a ham-sandwich cut algorithm to  $f(V)$  in  $\mathfrak{R}^3$ , find a plane  $H$  in  $\mathfrak{R}^3$  which bisects  $T_1, T_2$ , and  $T_3$ . Output the bisection  $\{V_1, V_2\}$  of  $V$  divided by  $H$ .  $\square$

As mentioned in Section 2.1, a ham-sandwich cut bisecting each  $T_i$  in  $\mathfrak{R}^3$  exists, and it can be found in  $O(n^{3/2})$  time. Hence, for proving the correctness of algorithm BISECT3, it suffices to show that Phase 1 can find an SC-embedding of  $G$  with boundary  $K$  into  $\mathfrak{R}^3$  in  $O(n^3 \log n)$  time. In the next section, we give a proof for this.

### 4 SC-embedding of a graph into $\mathfrak{R}^3$

In this section, given a 4-connected graph  $G$  which contains a complete graph with four nodes, denoted by  $K$ , we propose an algorithm, named EMBED3, for finding an SC-embedding of  $G$  with boundary  $K$  into  $\mathfrak{R}^3$  in  $O(n^3 \log n)$  time. Figure 2 shows an instance of such an SC-embedding of a 4-connected graph into  $\mathfrak{R}^3$ .

The algorithm EMBED3, which is an extension of the algorithm in  $\mathfrak{R}^2$  given in [11], consists of two steps. First, we contract arcs, one by one, until a small sized graph whose SC-embedding can be found easily is attained, and next, we embed the remaining nodes while tracing the process of the contraction reversely. More precisely, in the first step, we contract arcs in  $E - E(K)$ , one by one, while preserving the 4-connectivity until a complete graph  $G^*$  with 5 nodes containing  $K$  is obtained. Then we can easily obtain an SC-embedding  $f$  of  $G^*$  with boundary  $K$  into  $\mathfrak{R}^3$ ; we find an embedding  $f'$  of  $V(K)$  by putting them in general position (which shapes a trigonal pyramid), and we embed the

node  $v$  with  $\{v\} = V(G^*) - V(K)$  in  $\text{int}(\text{conv}_{f'}(V(K)))$ . In the second step, by tracing the process of the contraction reversely and embedding the contracted arcs into  $\mathfrak{R}^3$ , we convert the embedding  $f$  into the one for the original graph. The outline of algorithm EMBED3 is described as follows.

### Algorithm EMBED3

**Input:** A 4-connected graph  $G = (V, E)$  which has a complete graph  $K$  with 4 nodes.

**Output:** An SC-embedding of  $G$  with boundary  $K$  into  $\mathfrak{R}^3$ .

**Step 1:** While  $|V(G)| \geq 6$  holds, execute the following procedure (1) and (2).

(1) Find an arc  $e \in E(G) - E(K)$  such that  $G/e$  remains 4-connected, and contract the arc  $e$ .

(2) Let  $G := G/e$ .

/\*\* The current graph  $G$  obtained by Step 1 is a complete graph with 5 nodes containing  $K$ . \*\*/

**Step 2:** Embed  $G$  into  $\mathfrak{R}^3$  so that its embedding is an SC-embedding  $f$  with boundary  $K$ . Next, by tracing the process of the contraction in Step 1 reversely and embedding the contracted arcs into  $\mathfrak{R}^3$ , one by one, we convert the embedding  $f$  into the one for the original graph.  $\square$

In the subsequent sections, we prove the correctness of algorithm EMBED3 by describing the details for each step, analyze the time complexity of each step.

#### 4.1 Correctness of Step 1

We give a proof of the following theorem for the correctness of Step 1.

**Theorem 6** *Let  $G = (V, E)$  be a 4-connected graph which has a complete graph  $K$  with 4 nodes. Then there exists an arc  $e \in E - E(K)$  such that  $G/e$  is 4-connected.  $\square$*

We first introduce the following preparatory theorem about the contraction of arcs in 4-connected graphs.

**Definition 7** *A graph  $G$  is called uncontractible  $k$ -connected if  $G$  is  $k$ -connected and  $G/e$  is not  $k$ -connected for any arc  $e \in E(G)$ .  $\square$*



**Theorem 8** [9] *A graph  $G$  is uncontractible 4-connected if and only if  $G$  satisfies the following properties:*

(i)  $G$  is 4-connected,

(ii) the degree of each node in  $V(G)$  is exactly 4, and

(iii) for each arc  $(u, v) \in E(G)$ , there exists a node  $w \in V(G) - \{u, v\}$  with  $\{(u, w), (v, w)\} \subseteq E(G)$ .  $\square$

**PROOF of Theorem 6.** Let  $V_1 = V - V(K)$ . We construct the new graph  $G^*$  from  $G = (V, E)$ , defined as follows.  $V(G^*) = V_1 \cup V(K) \cup V_2$ , where  $V_2$  is a copy of  $V_1$ . An arc  $(u_1, u_2)$  belongs to  $E(G^*)$  if and only if (a)  $(u_1, u_2) \in E$ , (b)  $u_1, u_2 \in V_2$  and  $u_i, i = 1, 2$  is the copy of  $v_i \in V_1$  such that  $(v_1, v_2) \in E$ , or (c)  $u_1 \in V(K)$ ,  $u_2 \in V_2$ , and  $u_2$  is the copy of  $v_2 \in V_1$  such that  $(u_1, v_2) \in E$ . Note that  $G^*$  is also 4-connected. Since  $|N_{G^*}(v)| \geq 5$  holds for a node  $v \in V(K)$ , Theorem 8 implies that  $G^*$  has an arc  $e \in E(G^*)$  such that  $G^*/e$  is 4-connected. Without loss of generality, let  $e \in E - E(K)$  (note that  $e \notin E(K)$  since  $|V(K)| = 4$  and  $G^* - V(K)$  is not connected).

We claim that  $G/e$  remains 4-connected, proving the theorem. Assume by contradiction that  $G/e$  would have a node set  $X$  with  $|X| \leq 3$  such that  $(G/e) - X$  is not connected. Then there is a component  $C_\ell$  of  $(G/e) - X$  with  $V(C_\ell) \subseteq V_1$  since  $K$  is the complete graph. Indeed, if there would exist two distinct components  $C_i, C_j$  of  $(G/e) - X$  with  $V(K) \cap V(C_i) \neq \emptyset \neq V(K) \cap V(C_j)$ , then two nodes  $u \in V(K) \cap V(C_i)$  and  $v \in V(K) \cap V(C_j)$  would satisfy  $(u, v) \notin E$ , contradicting that  $K$  is a complete graph. Also in  $G^*/e$ ,  $N_{G^*/e}(V(C_\ell)) \subseteq X$  holds.  $V(G^*/e) - V(C_\ell) - X \neq \emptyset$  and this contradicts the 4-connectivity of  $G^*/e$ .  $\square$

Finally, we show that Step 1 can be implemented to run in  $O(n^3\alpha(n, n))$  time. First, we compute a sparse spanning 4-connected subgraph  $G'$  of  $G$  with  $V(G) = V(G')$  and  $O(n)$  arcs. Such a sparse spanning subgraph exists and it can be computed in linear time [10]. In the subsequent arguments about the time complexity of algorithm, let us assume that  $|E| = O(n)$ .

Now it was shown in [8] that it can be checked in  $O(n\alpha(n, n))$  time whether  $G$  is 4-connected or not, where  $\alpha$  denotes the inverse of the Ackermann's function. From  $|E| = O(n)$ , we can find a contractible edge in  $E(G) - E(K)$  in  $O(n^2\alpha(n, n))$  time. The number of the contraction is  $O(n)$ , and it follows that Step 1 can be implemented to run in  $O(n^3\alpha(n, n))$  time.

## 4.2 Correctness of Step 2

In this section, for a graph  $G = (V, E)$  and a subgraph  $G_1$  of  $G$ , we consider a situation where a graph  $G/e$  obtained from  $G$  by contracting some arc  $e = (u_1, u_2)$  with  $\{u_1, u_2\} - V(G_1) \neq \emptyset$  has an SC-embedding  $f'$  of  $G/e$  with boundary  $G_1$  into  $\mathfrak{R}^d$ . For proving the correctness of Step 2, we will show by the following Lemma 9 that if  $|N_G(u_i)| \geq d + 1$  holds for  $i = 1, 2$ , then we can find an SC-embedding of  $G$  with boundary  $G_1$  into  $\mathfrak{R}^d$ . Since Step 1 in algorithm EMBED3 contracts arcs while preserving the 4-connectivity, it follows that the degree of every node is always at least 4 in the current graph. Also note that any arc in boundary  $K$  is not contracted through the algorithm. Hence, we can observe that the following Lemma 9 proves the correctness of Step 2.

**Lemma 9** *Let  $G = (V, E)$  be a graph without isolated nodes and let  $f'$  be an SC-embedding of  $G/e$  with boundary  $G_1$  into  $\mathfrak{R}^d$  for an arc  $e = (u_1, u_2)$  with  $\{u_1, u_2\} - V(G_1) \neq \emptyset$ . Assume that for each node  $u_i$ ,  $i = 1, 2$ ,  $|N_G(u_i)| \geq d + 1$  holds if  $u_i \in V - V(G_1)$ . Then there is an SC-embedding of  $G$  with boundary  $G_1$  into  $\mathfrak{R}^d$ .  $\square$*

Before proving Lemma 9, we give some notations and one preparatory lemma for an embedding of a new point into  $\mathfrak{R}^d$ . For a convex polyhedron  $S$  in  $\mathfrak{R}^d$ , a supporting hyperplane  $H$  of  $S$  is called a *facet* of  $S$  if the dimension of  $H \cap S$  is  $d - 1$ . It is well-known that every full-dimensional convex polyhedron can be uniquely represented by all of its facets.

**Definition 10** *For a full-dimensional convex polyhedron  $S$  in  $\mathfrak{R}^d$ , let  $x$  be a vertex of  $S$ . Let  $\mathcal{H}_x$  denote the family of all facets  $H(a, b)$  of  $S$  containing the point  $x$  such that  $S \subseteq H^+(a, b)$ . Define the following polyhedron (see Figure 3) :*

$$D(x, S) = \bigcap_{H(a,b) \in \mathcal{H}_x} (H^-(a, b) - H(a, b)). \quad \square$$

It is not hard to see the following property.

**Lemma 11** *Let  $P$  be a set of points in  $\mathfrak{R}^d$  such that  $\text{conv}(P)$  is full-dimensional, and let  $x$  be a vertex of  $\text{conv}(P)$ . Then for a point  $y \in \mathfrak{R}^d$ ,  $x \in \text{int}(\text{conv}(P \cup \{y\}))$  if and only if  $y \in D(x, \text{conv}(P))$ .*

**PROOF.** Assume that  $y \notin D(x, \text{conv}(P))$ . Then from the definition of  $D(x, \text{conv}(P))$ , there exists a facet  $H(a, b)$  of  $\text{conv}(P)$  containing  $x$  such that  $P \cup \{y\} \subseteq H^+(a, b)$ . This indicates that  $x \notin \text{int}(\text{conv}(P \cup \{y\}))$ .

Assume by contradiction that  $y \in D(x, \text{conv}(P))$  and there is a facet  $H(a, b)$  of  $\text{conv}(P \cup \{y\})$  containing  $x$ . If  $H(a, b)$  contains  $y$ , then it follows from

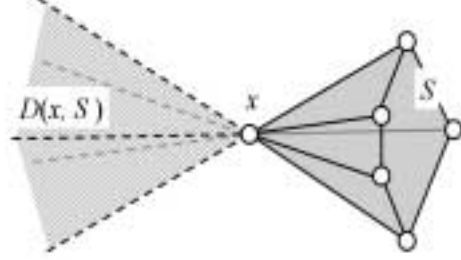


Fig. 3. Illustration for definition of  $D(x, S)$ .

the definition of  $D(x, \text{conv}(P))$  that  $(H^+(a, b) - H(a, b)) \cap \text{conv}(P) \neq \emptyset \neq (H^-(a, b) - H(a, b)) \cap \text{conv}(P)$ , a contradiction. Hence,  $H(a, b)$  does not contain  $y$ , and without loss of generality  $\text{conv}(P \cup \{y\}) \subseteq H^+(a, b)$  holds. It follows that  $H(a, b)$  is also a facet of  $\text{conv}(P)$ , which contradicts that  $y \in H^-(a, b) - H(a, b)$  holds (from the definition of  $D(x, \text{conv}(P))$ ).  $\square$

**PROOF of Lemma 9.** Let  $u^* \in V(G/e)$  denote the node obtained by contracting  $u_1$  and  $u_2$  in  $G$ . Without loss of generality, assume  $u_2 \in V - V(G_1)$  (this is possible from the assumption  $\{u_1, u_2\} - V(G_1) \neq \emptyset$ ). Hence  $|N_G(u_2)| \geq d+1$  holds. We give a constructive proof of the lemma; we show a way of finding an SC-embedding  $f$  of  $G$  with boundary  $G_1$  into  $\mathfrak{R}^d$ . Let  $f(v) := f'(v)$  for each node  $v \in V(G/e) - \{u^*\} = V(G) - \{u_1, u_2\}$  and  $f(u_1) := f'(u^*)$  (see Figure 4(a)). Note that  $G_1$  also plays the role as  $G'$  in Definition 4 (i), and that every node  $v \in V(G) - V(G_1) - (N_G(u_2) \cup \{u_2\})$  satisfies  $v \in \text{int}(\text{conv}_f(N_G(v)))$ . We prove this lemma by showing that  $u_2$  can be embedded so that each node  $v \in \{u_2\} \cup N_G(u_2) - V(G_1)$  is strictly inside the convex hull of its neighbors.

First, observe that  $u_2$  needs to be embedded in  $\text{int}(\text{conv}_f(N_G(u_2)))$  for the convexity for  $u_2$  (see Figure 4(b)). Note that the position of each node  $v \in V(G) - \{u_2\}$  has been fixed, so  $\text{int}(\text{conv}_f(N_G(u_2)))$  is well-defined. Since  $|N_G(u_2)| \geq d+1$  holds and the points of  $\{f(v) \mid v \in N_G(u_2)\}$  are in general position, it follows that  $\text{int}(\text{conv}_f(N_G(u_2))) \neq \emptyset$ .

For the convexity for each  $v \in N_G(u_2) - V(G_1)$ ,  $u_2 \in D_v$  must hold by Lemma 11, where  $D_v = D(v, \text{conv}_f(N_G(v) \cup \{v\} - \{u_2\}))$  if  $v$  is a vertex of  $\text{conv}_f(N_G(v) \cup \{v\} - \{u_2\})$ ,  $D_v = \mathfrak{R}^d$  otherwise (see Figure 4(c)). Here note that in the case of  $D_v = D(v, \text{conv}_f(N_G(v) \cup \{v\} - \{u_2\}))$ ,  $\text{conv}_f(N_G(v) \cup \{v\} - \{u_2\})$  is full-dimensional in  $\mathfrak{R}^d$ . Indeed, if  $v = u_1$  holds, then  $u_1 \notin V(G_1)$  indicates  $|N_G(u_1)| \geq d+1$  and hence we have  $|N_G(u_1) \cup \{u_1\} - \{u_2\}| \geq d+1$ . If  $v \neq u_1$  holds, then  $v \in \text{int}(\text{conv}_{f'}(N_{G/e}(v)))$  indicates that  $|N_{G/e}(v)| \geq d+1$ ,

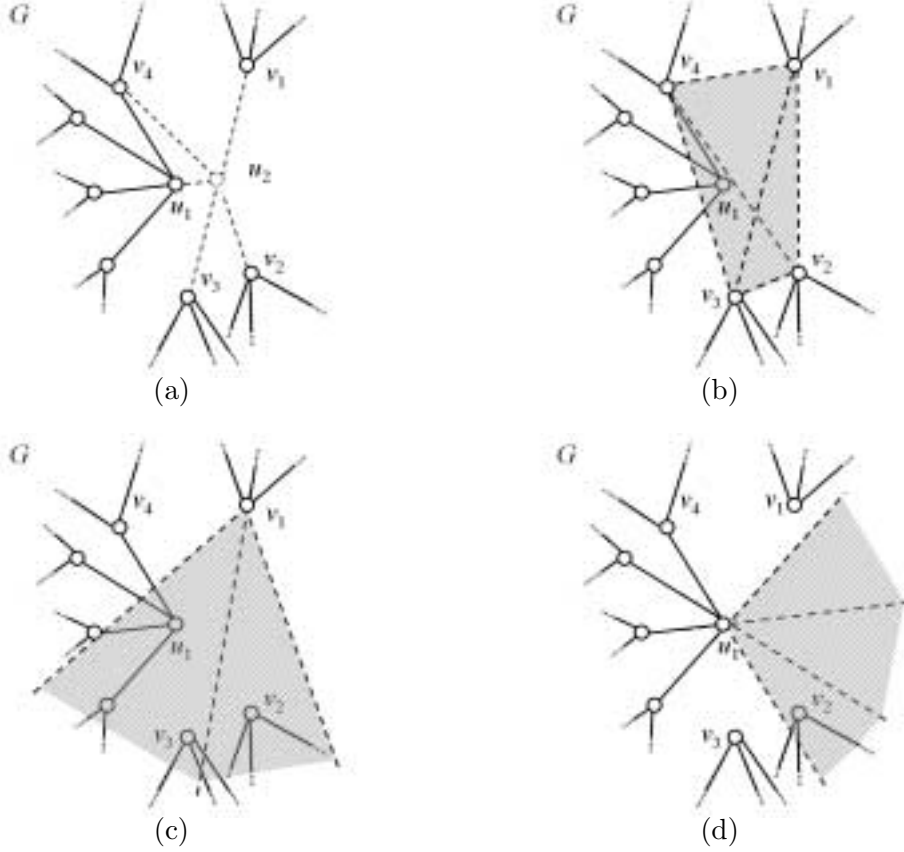


Fig. 4. Illustration for the positions  $f(u_2)$ : (a) shows an embedding of nodes in  $V(G) - \{u_2\}$ , where  $f(v) = f'(v)$  for each node  $v \in V(G) - \{u_1, u_2\}$  and  $f(u_1) = f'(u^*)$ . (a) also shows  $N_G(u_2) = \{u_1, v_1, v_2, v_3, v_4\}$  for the node  $u_2$  which has not been embedded. (b), (c), and (d) show  $\text{int}(\text{conv}_f(N_G(u_2)))$ ,  $D(v_1, \text{conv}_f(N_G(v_1) \cup \{v_1\} - \{u_2\}))$ , and  $D(u_1, \text{conv}_f(N_G(u_1) \cup \{u_1\} - \{u_2\}))$ , respectively.

$|N_G(v) - \{u_2\}| \geq |N_{G/e}(v) - \{u^*\}| \geq d$ , and  $|N_G(v) \cup \{v\} - \{u_2\}| \geq d + 1$  hold (note that the points of  $\{f'(w) \mid w \in V(G/e)\} = \{f(w) \mid w \in V(G) - \{u_2\}\}$  are in general position). Let  $D^* = \bigcap_{v \in N_G(u_2) - V(G_1)} D_v$ .

Hence, for proving the lemma, it suffices to show that  $D^* \cap \text{int}(\text{conv}_f(N_G(u_2))) \neq \emptyset$  holds; we can embed  $u_2$  in  $D^* \cap \text{int}(\text{conv}_f(N_G(u_2)))$  (while satisfying that the points of  $\{f(v) \mid v \in V(G)\}$  are in general position). There are the following two possible cases: (I)  $f(u_1) \in \text{int}(\text{conv}_f(N_G(u_1) - \{u_2\}))$  or  $u_1 \in V(G_1)$ , (II) otherwise.

(I) In this case, we have  $D_{u_1} = \mathfrak{R}^d$  or  $u_1 \notin N_G(u_2) - V(G_1)$ ; we do not have to consider the convexity for  $u_1$ . Since each node  $v \in N_G(u_2) - V(G_1) - \{u_1\}$  satisfies  $v \in N_{G/e}(u^*)$  and  $f(v) = f'(v) \in \text{int}(\text{conv}_{f'}(N_{G/e}(v)))$ , it follows that  $D^*$  contains the point  $f'(u^*) = f(u_1)$ . This implies that  $D^* \neq \emptyset$ . Moreover, since  $D^*$  is an open set, we can observe that  $D^*$  contains points sufficiently close to  $f(u_1)$ . This and  $u_1 \in N_G(u_2)$  indicate that  $D^* \cap \text{int}(\text{conv}_f(N_G(u_2))) \neq \emptyset$ .

$\emptyset$ .

(II) In this case,  $f(u_1)$  is a vertex of  $\text{conv}_f(N_G(u_1) \cup \{u_1\} - \{u_2\})$  and  $u_1 \in V(G) - V(G_1)$  holds, and hence  $D_{u_1} = D(u_1, \text{conv}_f(N_G(u_1) \cup \{u_1\} - \{u_2\}))$  holds (see Figure 4(d)). Let  $D' = \bigcap_{v \in N_G(u_2) - V(G_1) - \{u_1\}} D_v$  (note that  $D^* = D' \cap D_{u_1}$ ). Similarly to the arguments in (I), we can observe that  $D'$  contains the point  $f(u_1)$  and points sufficiently close to  $f(u_1)$ . From  $u_2 \in N_G(u_1)$  and the definition of  $D_{u_1}$ , we can observe that if  $D_{u_1} \cap \text{int}(\text{conv}_f(N_G(u_2))) \neq \emptyset$ , then some points sufficiently close to  $f(u_1)$  included in  $D'$  are also contained in  $D_{u_1} \cap \text{int}(\text{conv}_f(N_G(u_2)))$ . Based on this observation, for proving  $D^* \cap \text{int}(\text{conv}_f(N_G(u_2))) \neq \emptyset$ , it suffices to show that  $D_{u_1} \cap \text{int}(\text{conv}_f(N_G(u_2))) \neq \emptyset$ .

Assume by contradiction that  $D_{u_1} \cap \text{int}(\text{conv}_f(N_G(u_2))) = \emptyset$ . From the definition of  $D_{u_1} = D(u_1, \text{conv}_f(N_G(u_1) \cup \{u_1\} - \{u_2\}))$ , it follows that there exists a supporting hyperplane  $H(a, b)$  of  $\text{conv}_f(N_G(u_1) \cup \{u_1\} - \{u_2\})$  containing the point  $f(u_1)$  such that without loss of generality,  $\text{conv}_f(N_G(u_1) \cup \{u_1\} - \{u_2\}) \cup \text{conv}_f(N_G(u_2)) \subseteq H^+(a, b)$  holds. This and  $N_{G/e}(u^*) = (N_G(u_1) - \{u_2\}) \cup (N_G(u_2) - \{u_1\})$  indicate that  $H(a, b)$  is a supporting hyperplane of  $\text{conv}_{f'}(N_{G/e}(u^*) \cup \{u^*\})$  containing  $f'(u^*) (= f(u_1))$  in  $\mathfrak{R}^d$ . It follows that  $f'(u^*) \in \text{int}(\text{conv}_{f'}(N_{G/e}(u^*)))$  cannot hold and it violates the statement (ii) in Definition 4 about  $f'$ , which contradicts that  $f'$  is an SC-embedding of  $G/e$  (note that  $u^* \notin V(G_1)$  from  $\{u_1, u_2\} \subseteq V(G) - V(G_1)$ ).  $\square$

Here we show that Step 2 of algorithm EMBED3 can be implemented to run in  $O(n^3 \log n)$  time. Since the number of the contraction in Step 1 is  $O(n)$ , it suffices to show that given an SC-embedding  $f'$  of  $G/e$  with boundary  $G_1$  into  $\mathfrak{R}^3$  for an arc  $(u_1, u_2)$  with  $\{u_1, u_2\} - V(G_1) \neq \emptyset$ , we can find an SC-embedding  $f$  of  $G$  with boundary  $G_1$  in  $O(n^2 \log n)$  time. According to the proof of Lemma 9, we can observe that the time complexity of this procedure depends on that of choosing a location for  $u_2$ .

First we need to compute  $\text{conv}_f(N_G(u_2))$  and  $D_v$  for each node  $v \in N_G(u_2)$ . It is known in [5] that for a set  $P$  of points in  $\mathfrak{R}^3$ ,  $\text{conv}(P)$  can be computed in  $O(n \log n)$  time. Moreover, from Definition 10, we can observe that the time complexity of computing  $D(v, \text{conv}(P))$  depends on that of computing  $\text{conv}(P)$ . Hence it follows that  $\text{conv}_f(N_G(u_2))$  and  $D_v$  for each node  $v \in N_G(u_2)$  can be computed in  $O(n^2 \log n)$  time. Now since the number of facets representing  $\text{conv}(P)$  is  $O(n)$ , the number of hyperplanes representing  $D^* \cap \text{conv}_f(N_G(u_2))$  is  $O(n^2)$ . This implies that  $D^* \cap \text{conv}_f(N_G(u_2))$  can be computed in  $O(n^2 \log n)$  time. Moreover, it is not difficult to observe that we can find a location for  $u_2$  in  $D^* \cap \text{conv}_f(N_G(u_2))$  in  $O(n^2 \log n)$  time so that all points in  $f(V(G))$  are in general position.

Before closing this section, we remark that the following properties hold as a corollary of Theorem 1.

**Corollary 12** *Let  $G = (V, E)$  be a 4-connected graph and contain pairwise disjoint subsets  $T_1, T_2, T_3$  of  $V$  such that  $|T_i|$  is even for  $i = 1, 2, 3$ . Then, by adding at most two extra arcs to  $G$ , a 3-bisection can be obtained. In particular, if  $G$  has  $K_3$ , then by adding at most one extra arc to  $G$ , a 3-bisection can be obtained.*

**PROOF.** Let  $(u, v), (v, w), (w, x)$  be three arcs in  $G$  (note that such three arcs always exist). In  $G' = G + \{(u, w), (v, x), (u, x)\}$ ,  $\{u, v, w, x\}$  forms  $K_4$ . So by applying BISECT3 to  $G'$ , we obtain a 3-bisection  $\{V_1, V_2\}$ . It is not difficult to see that at least one of extra arcs  $(u, w), (v, x)$ , and  $(u, x)$  can be deleted while preserving the feasibility, which proves the first statement of this corollary. Then note that it turns out to be only the case of  $\{v, x\} \subseteq V_1$  and  $\{w, u\} \subseteq V_2$  that two extra arcs may be necessary. Hence, in the case where  $G$  has  $K_3$ , if we choose  $u, v, w, x$  such that  $\{u, v, w\}$  forms  $K_3$  and  $(w, x) \in E$ , then we can observe that one extra arc suffices.  $\square$

## 5 Concluding Remarks

### 5.1 Sufficient connectivity for a general $k$

We have shown that any 4-connected graph with  $K_4$  admits a 3-bisection. However, the sufficient connectivity for a graph to admit a  $k$ -bisection is still a challenging question even if  $k = 3$ . Let  $f(k)$  denote the smallest  $p$  such that any  $p$ -connected graph admits a  $k$ -bisection. From results in [12,14] and [11],  $f(1) = 2$  and  $f(2) = 3$  hold, respectively. In this paper, we have shown  $f(k) \geq 6$  for  $k \geq 3$  (see Figure 1).

As mentioned at the end of Section 2.2, from Theorem 3 and Lemma 5, we can observe that if an SC-embedding of  $G$  into  $\mathfrak{R}^k$  exists, then  $G$  admits a  $k$ -bisection. Lemma 9 implies that if

- (a)  $G$  has a subgraph  $G_1$  which plays the role as  $G'$  in Definition 4 (i),
- (b)  $|N_G(v)| \geq k + 1$  holds for each node  $v \in V(G) - V(G_1)$ , and
- (c) we can continue contracting arcs not in  $E(G_1)$  while preserving the above (b) until a graph consisting of  $G_1$  and at most one extra node is obtained,

then an SC-embedding of  $G$  with boundary  $G_1$  into  $\mathfrak{R}^k$  can be found. Hence, a sufficient condition for  $G$  to satisfy the above (a)–(c) indicates one for  $G$  to have a  $k$ -bisection.

## 5.2 The arc-version of the bisection problem

We here give a proof of Theorem 2 about the arc-version of the bisection problem; given an undirected graph  $G = (V, E)$  and  $k$  pairwise disjoint sets  $T_1, \dots, T_k$  of arcs, where each  $|T_i|$  is even, find a partition  $E_1$  and  $E_2$  of  $E$  such that the graphs induced by  $E_1$  and  $E_2$  are both connected and  $|E_1 \cap T_i| = |E_2 \cap T_i| = |T_i|/2$  holds for each  $i = 1, \dots, k$ . This can be done by using a reduction to the node-version of the problem.

Let  $G$  be a  $(k + 1)$ -edge-connected graph. If  $G$  has exactly  $k + 1$  arcs, then it has exactly two nodes and it is trivial. Assume that  $|E(G)| > k + 1$ . Let  $L(G)$  denote the line graph of  $G$ , and  $V_L(E') \subseteq V(L(G))$  denote the node set of  $L(G)$  corresponding to an arc set  $E' \subseteq E$  in  $G$ . Observe that  $\{E_1, E_2\}$  is a  $k$ -bisection of the arc set with respect to  $\{T_1, \dots, T_k\}$  in  $G$  if and only if  $\{V_L(E_1), V_L(E_2)\}$  is a  $k$ -bisection of the node set with respect to  $\{V_L(T_1), \dots, V_L(T_k)\}$  in  $L(G)$ . Moreover, if  $G$  is  $(k + 1)$ -edge-connected, then  $L(G)$  is  $(k + 1)$ -connected and has  $K_{k+1}$ .

As mentioned in Section 1 and Theorem 1, if  $k = 1, 2, 3$ , then  $L(G)$  admits a  $k$ -bisection of the node set, and hence  $G$  admits a  $k$ -bisection of the arc set. Finally, we remark that there exist instances that have no feasible partition unless  $G$  is  $(k + 1)$ -edge-connected for  $k = 1, 2, 3$ . Therefore,  $f'(k) = k + 1$  holds for  $k = 1, 2, 3$ , where  $f'(k)$  denotes the smallest  $p$  such that any  $p$ -edge-connected graph admits a  $k$ -bisection of the arc set.

### Acknowledgments:

We are very grateful to two anonymous referees for their helpful suggestions. This research was partially supported by the Scientific Grant-in-Aid from Ministry of Education, Culture, Sports, Science and Technology of Japan.

### References

- [1] B. Bozkaya, G. Laporte, E. Erkut, *A tabu search heuristic and adaptive memory procedure for political districting*, European J. Operational Research, **144** (2003), 12–26.
- [2] L. Chi-Yuan, J. Matoušek and W. Steiger, *Algorithms for ham-sandwich cuts*, Discrete Comput. Geom., **11** (1994), 433–452.

- [3] J. Chleíková, *Approximating the maximally balanced connected partition problem in graphs*, Information Processing Letters, **60** (1999), 225–230.
- [4] M. E. Dyer and A. M. Frieze, *On the complexity of partitioning graphs into connected subgraphs*, Discrete Applied Mathematics, **10** (1985), 139–153.
- [5] H. Edelsbrunner, *Algorithms in combinatorial geometry*, Springer-Verlag, Berlin, 1987.
- [6] R. C. Gonzales and P. Wintz, *Digital Image Processing*, Publisher Addison-Wesley, Reading, MA, 1977.
- [7] B. Hayes, *Machine politics*, American Scientist, **84** (1996), 522–526.
- [8] A. Kanevsky, R. Tamassia, G. Di Battista, and J. Chen, *On-line maintenance of the four-connected components of a graph*, Proc. 32th IEEE Symp. on Foundations of Computer Science, (1991), 793–801.
- [9] N. Martinov, *A recursive characterization of the 4-connected graphs*, Discrete Mathematics, **84** (1990), 105–108.
- [10] H. Nagamochi and T. Ibaraki, *A linear-time algorithm for finding a sparse  $k$ -connected spanning subgraph of a  $k$ -connected graph*, Algorithmica, **7** (1992), 583–596.
- [11] H. Nagamochi, T. Jordán, Y. Nakao, and T. Ibaraki, *Convex embeddings bisecting of 3-connected graphs*, Combinatorica, **22**(4) (2002), 537–554.
- [12] H. Suzuki, N. Takahashi, and T. Nishizeki, *A linear algorithm for bipartition of biconnected graphs*, Information Processing Letters, **33** (1990), 227–232.
- [13] D. C. Tsichritzis and P. A. Bernstein, *Operating Systems*, Academic Press, New York, 1981.
- [14] K. Wada and K. Kawaguchi, *Efficient algorithms for tripartitioning triconnected graphs and 3-edge-connected graphs*, Lecture Notes in Comput. Sci., 790, Springer, Graph-theoretic concepts in computer science, 1994, 132–143.
- [15] J. C. Williams Jr, *Political redistricting: a review*, Papers in Regional Science, **74** (1995), 12–40.