A Linear Algorithm for Resource Tripartitioning Triconnected Planar Graphs

TANVEER AWAL[1]
MD. SAIDUR RAHMAN[2]

Department of Computer Science and Engineering
Bangladesh University of Engineering and Technology
Dhaka -1000, Bangladesh
[1]tanveerawal@yahoo.com
[2]saidurrahman@cse.buet.ac.bd

Abstract. Given a connected graph $G = (V, E)$, a set $V_r \subseteq V$ of r special vertices, three distinct base vertices $u_1, u_2, u_3 \in V$ and three natural numbers r_1, r_2, r_3 such that $r_1 + r_2 + r_3 = r$, we wish to find a partition V_1, V_2, V_3 of V such that V_i contains u_i and r_i vertices from V_r, and V_i induces a connected subgraph of G for each $i, 1 \leq i \leq 3$. We call a vertex in V_r a resource vertex and the problem above of partitioning vertices of G as the resource 3-partitioning problem. In this paper, we give a linear-time algorithm for finding a resource tripartition of a 3-connected planar graph G. Our algortihm is based on a nonseparating ear decomposition of G and st-numbering of G. We also present a linear algorithm to find a nonseparating ear decomposition of a 3-connected planar graph. This algorithm has bounds on ear-length and number of ears.

Keywords: Algorithm, Nonseparating ear decomposition, Planar graph, Resource tripartitioning, Resource bipartitioning, st-numbering, Triconnected graph.

(Received December 25, 2009 / Accepted July 02,2010)

1 Introduction

Let $G = (V, E)$ be a connected graph of $|V| = n$ vertices. Among these n vertices of G, some belong to a special class of vertices that we call *resource vertices*. Let $V_r \subseteq V$ be the set of resource vertices and $|V_r| = r$. Let $u_1, u_2, u_3 \in V$ be three designated vertices and r_1, r_2, r_3 be three natural numbers such that $r_1+r_2+r_3 = r$. Our goal is to find a partition V_1, V_2, V_3 of V such that $u_1 \in V_1, u_2 \in V_2, u_3 \in V_3$, V_i contains r_i resource vertices and V_i induces a connected subgraph of G for each $i, 1 \leq i \leq 3$. We call this partitioning of vertices a *resource 3-partitioning* of G. For example, Figure 1(a) shows a connected graph G with $n = 15, r = 8$ vertices, where each resource vertex is drawn by white circle. Figure 1(b) illustrates a resource 3-partition of G for $r_1 = 3, r_2 = 2, r_3 = 3$.

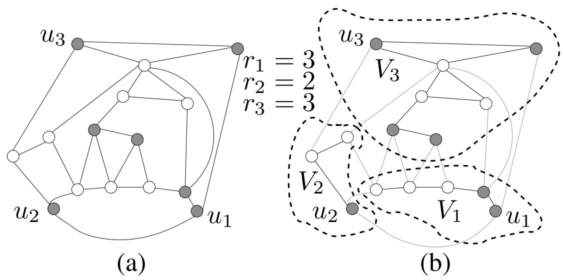


Figure 1: A resource 3-partition of a 3-connected planar graph.

The resource tripartitioning problem is a special case of resource k-partitioning problem, for $k = 3$. A *resource k-partitioning* is defined as partition $V_1, V_2, \ldots, V_k$ of V with a set $V_r \subseteq V$ of r resource vertices, base vertices $u_1, u_2, \ldots, u_k \in V$, k natural numbers $r_1, r_2, \ldots, r_k$ such that $\sum_{i=1}^{k} r_i = r$, where $u_i \in V_i, V_i$ contains r_i resource vertices and V_i induces a connected subgraph of G for each $i, 1 \leq i \leq k$.

Resource partitioning has significant applications in various areas. In computer networks, we may consider printers, routers, scanners etc. as resources. Resources need to be partitioned to balance loads on these resources and to prevent network traffic bottleneck. Furthermore, in multimedia networks, it is desired to assign a server to a specific group of clients for balancing loads among the servers. Again, in electrical power distribution systems, resource partitioning has another real-time application to serve consumers better. Here, distributed resources include a variety of energy sources like turbines, photovoltaics, fuel-cells and storage devices with various capacities. Distribution of these resources among the demand centers offers increased reliability, lower cost of power delivery and additional supply flexibility. Resource partitioning has its application in the fault-tolerant routing of communication networks [13, 22] and in computational aspects, too. For example, in grid computing we wish to divide a complex task such as computation of fractals into several subtasks and then we wish to delegate each of these subtasks to a computing element in the grid such that a computing element in the grid might not be overwhelmed with tasks from various other clients. This concept is applicable to telecommunication networks, fault tolerant systems, various producer-consumer problems and so on.

A related problem is a *k-partitioning* problem in which we are given a graph $G = (V, E)$, k distinct base vertices $u_1, u_2, \ldots, u_k \in V$, and k natural numbers $n_1, n_2, \ldots, n_k$ such that $\sum_{i=1}^{k} n_i = |V|$, we wish to find a partition $V_1, V_2, \ldots, V_k$ of the vertex set V such that $u_i \in V_i$; $|V_i| = n_i$; V_i induces a connected subgraph of G for each $i, 1 \leq i \leq k$.

The k-partitioning problem is NP-complete in general [7]. Although not every graph has a k-partition, Györi and Lovász independently proved that every k-connected graph has a k-partition for any $u_1, u_2, \ldots, u_k$ and $n_1, n_2, \ldots, n_k$ [9, 14]. However, their proofs do not yield any polynomial time algorithm for actually finding a k-partition of a k-connected graph. For the case $k = 2, 3, 4$, following algorithms have been known:

(i) There is a linear-time algorithm to find a bipartition of a biconnected graph [19, 20].

(ii) There is an $O(n^2)$ time algorithm to find a 3-partition of a triconnected graph [20].

(iii) There is a linear-time algorithm to find a 4-partition of a four connected planar graph with base vertices located on the same face of the given graph [17].

On the other hand, polynomial-time algorithms have not been known for the case $k \geq 4$. A polynomial-time algorithm for any k is claimed in [15], but is not correct [17]. If all the vertices are resource vertices then resource k-partitioning and k-partitioning problem are the same. Thus resource k-partitioning problem is also NP-complete. [18] claims the resource k-partitioning problem to be NP-hard but their claim is not correct. The following algorithms are known for finding a resource k-partition of a graph for $k = 2, 3, 4$.

(i) There are linear-time algorithms to find resource bipartitions of path-reducible graphs, series-parallel graphs and connected graphs where all resource vertices are contained in the same biconnected component [18].

(ii) There is an $O(n^2)$ time algorithm to find vertex-subset tripartitions (equivalent to resource tripartitions [18]) of triconnected and 3-edge-connected graphs [21].

(iii) There is an $O(n)$ algorithm to find a resource four-partition of a 4-connected planar graph with four base vertices located on the same face of a planar embedding [1].

But there exists no polynomial-time algorithm for resource k-partitioning of graphs for $k > 3$. In this paper, we give a linear algorithm for finding a resource tripartition of a 3-connected planar graph based on a “nonseparating ear decomposition" of the given graph. “Nonseparating ear decomposition" is a generalization of “canonical decomposition" [1]. “Canonical decomposition" is applied in convex grid drawing of planar graph [5]. “Canonical decomposition" i.e. “canonical ordering" has applications in producing straight line grid drawings with polynomial sizes for planar graphs. A “canonical decomposition", a “realizer", a “Schnyder labeling" and an “orderly spanning tree" of a plane graph play an important role in straight-line drawings, floorplanning, graph encoding etc. [2, 4, 6, 10, 12]. Miura et. al. proved that a “canonical decomposition", a “realizer", a “Schnyder labeling", an “orderly spanning tree" and an “outer triangular convex grid drawing" are notions equivalent with each other [16]. Hence

[1] T. Awal, *Resource partitioning on planar graphs*, M.Sc. Engineering Thesis, Department of Computer Science and Engineering, Bangladesh University of Engineering and Technology, Dhaka, November 2007.

"nonseparating ear decomposition" is a generalization of "canonical decomposition", a "realizer", a "Schnyder labeling", an "orderly spanning tree" and an "outer triangular convex grid drawing". Using a "nonseparating ear decomposition" of a 3-connected graph G, Cheriyan and Maheshwari finds three "independent spanning trees" rooted at a vertex r in G.

The rest of the paper is organized as follows. Section 2 gives some definitions. In section 3, we present a constructive proof for the existence of a nonseparating ear decomposition through two vertices a, b and avoiding a third vertex c of a triconnected planar graph G for any a, b, c. In section 4, we present a linear-time algorithm for finding a resource tripartitioning of a 3-connected planar graph based on a nonseparating ear decomposition of G. Finally section 5 is a conclusion.

2 Preliminaries

In this section we define several graph theoretical terms used in this paper.

Let $G(V, E)$ be a connected simple graph with vertex set $V(G)$ and edge set $E(G)$. We denote by n the number of vertices in G and by m the number of edges in G. Thus $n = |V(G)|, m = |E(G)|$. An edge joining vertices u, v is denoted by (u, v). The *degree* of a vertex v in a graph G, denoted by $d(v)$, is the number of edges incident to v in G. The *connectivity* $\kappa(G)$ of a graph G is the minimum number of vertices whose removal results in a disconnected graph or a single-vertex graph K_1. We say that G is *k-connected* if $\kappa(G) \geq k$. We call a vertex of G a *cut vertex* if its removal results in a disconnected graph or a single-vertex graph. We call two vertices of G a *separation pair* if their removal results in a disconnected graph or a single-vertex graph. A *walk*, $v_0, e_1, v_1, \ldots, v_{l-1}, e_l, v_l$, in a graph G is an alternating sequence of vertices and edges of G, beginning and ending with a vertex, in which each edge is incident to two vertices immediately preceding and following it. If the vertices $v_0, v_1, \ldots, v_l$ are distinct (except possibly v_0, v_l), then the walk is called a *path* and usually denoted either by the sequence of vertices $v_0, \ldots, v_l$ or by the sequence of edges $e_1, e_2 \ldots, e_l$. The *length* of a path is l which is one less than the number of vertices on the path. A path or walk is *open* if $v_0 \neq v_l$. A path or walk is *closed* if $v_0 = v_l$. A closed path containing at least one edge is called a *cycle*. For a path $P, V_{in}(P)$ denotes the internal vertices of P, i.e., all the vertices except the endpoints. For $W \subseteq V$, we denote by $G - W$ the graph obtained from G by deleting all vertices in W and all edges incident to them.

Let s and t be any two vertices of a connected graph G. An *st-numbering* of G is a numbering of its vertices by integers $1, 2, \ldots, n$ such that a vertex s receives number 1, a vertex t receives number n and every other vertex of G is adjacent to at least one lower-numbered vertex and at least one higher-numbered vertex. An interesting property of st-numbering of a graph is shown in the following fact.

(st1) If a graph G has an st-numbering $\pi = v_1, v_2, \ldots, v_n$, then both the subgraphs of G induced by $v_1, v_2, \ldots, v_i$ and $v_{i+1}, v_{i+2}, \ldots, v_n$ are connected for each $i, 1 \leq i \leq n$.

Not every connected graph has an st-numbering but the following Lemma [8] holds.

Lemma 2.1 *Let G be a biconnected undirected graph and (s, t) be any edge of G. Then G has an st-numbering $\pi = v_1, v_2, \ldots, v_n$ such that $v_1 = s$ and $v_n = t$, and π can be found in linear time.*

An *ear decomposition* of a biconnected graph G is a decomposition $G = P_0 \cup P_1 \cup \ldots \cup P_k$, where P_k is a path or cycle and $P_i, 0 \leq i \leq k-1$, is a path with only its two distinct end vertices in common with $P_k \cup P_{k-1} \cup \ldots \cup P_{i+1}$. An *ear* is a path. An *open ear* is a path with two distinct end vertices. We call an ear a *trivial ear* if the length of the ear is one. We call an ear a *non-trivial ear* if the length of the ear is greater than one. Let a, b, c be three vertices in G and P_i be open paths on G. We denote by G_i the subgraph of G induced by the edges of $P_0 \cup P_1 \cup \ldots \cup P_i$, by $\bar{G}_i$ the subgraph of G induced by the edges of $P_{i+1} \cup P_{i+2} \cup \ldots \cup P_k$. So $G_k = G$. Then $P_0, P_1, \ldots, P_k$ is a *nonseparating ear decomposition (nsed) through vertices a, b and avoiding vertex c* if the following five conditions hold.

(nsed1) If $(a, b) \in E(G)$, then P_k is a cycle containing the edge (a, b). Otherwise, P_k is a path in G with the vertices a, b as endpoints.

(nsed2) The first non-trivial ear has only one internal vertex and the internal vertex is c.

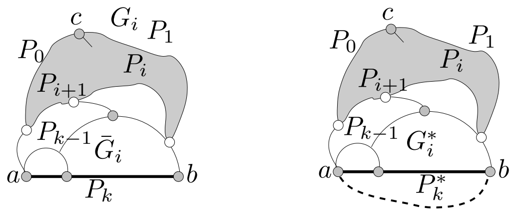
(nsed3) For each $i, 0 \leq i < k, P_i$ is a path connecting 2 distinct vertices of $\bar{G}_i$ and $V_{in}(P_i) \bigcap V(\bar{G}_i) = \phi$.

(nsed4) For each $i, 0 \leq i \leq k, G_i$ is connected.

(nsed5) For each $i, 1 \leq i \leq k$, each internal vertex of P_i has a neighbor in G_{i-1}.

Let augmented graph, $G_i^* = (V(\bar{G}_i), E(\bar{G}_i) \cup \{(a, b)\})$ and augmented $P_k, P_k^* = (V(P_k), E(P_k) \cup \{(a, b)\})$. For both of P_k^* and G_i^*, $(a, b) \notin E(G)$ and (a, b) becomes an outer edge of $\bar{G}_i$. Note that P_k^* is a cycle and

$P_k^* \cup P_{k-1} \cup \ldots \cup P_i$ is biconnected for each $i, 0 \leq i \leq k$.



(a) (b)

Figure 2: Nonseparating Ear Decomposition.

Figure 2 shows a nonseparating ear decomposition through a, b and avoiding c, P_k is drawn by thick lines. Figure 2(a) shows G_i and $\bar{G}_i$, the white vertices are contained both in G_i and $\bar{G}_i$. Figure 2(b) shows G_i, G_i^* and P_k^*, the white vertices are contained both in G_i and G_i^*.

A graph is *planar* if it can be embedded in the plane so that no two edges intersect geometrically except at a vertex to which they are both incident. A *plane graph* is a planar graph with a fixed embedding. A plane graph divides the plane into connected regions called *faces*. We regard the contour of a face as a clockwise cycle formed by the edges on the boundary of the face. We denote the contour of the outer face of graph G by $C_o(G)$. We write $C_o(G) = w_1, w_2, \ldots, w_h, w_1$ if the vertices $w_1, w_2, \ldots, w_h$ on $C_o(G)$ appear clockwise in this order, as illustrated in Figure 3. We call a vertex an *outer vertex* and an edge an *outer edge*, if the vertex and edge respectively lie on $C_o(G)$. We call a path P in a biconnected plane graph G a *chord-path* of G if P satisfies the following conditions.

(i) P connects two outer vertices $w_p, w_q, p < q$;

(ii) $\{w_p, w_q\}$ is a separation pair of G;

(iii) P lies on an inner face; and

(iv) P does not pass through any outer edge and any outer vertex other than the ends w_p and w_q .

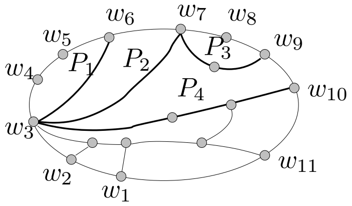


Figure 3: A plane graph with chord-paths P_1, P_2, P_3, P_4

The plane graph G in Figure 3 has four chord-paths $P_1, P_2, \ldots, P_4$ drawn by thick lines. A chord-path P is *minimal* if none of $w_{p+1}, w_{p+2}, \ldots, w_{q-1}$ is an end of a chord-path. Thus the definition of a minimal chord-path depends on which vertex is considered as the starting vertex w_1 of $C_o(G)$. P_1 and P_3 in Figure 3 are minimal, while P_2 and P_4 are not minimal. Let $\{v_1, v_2, \ldots, v_p\}$, $p \geq 3$, be a set of three or more outer vertices consecutive on $C_o(G)$ such that $d(v_1) \geq 3, d(v_2) = d(v_3) = \ldots = d(v_{p-1}) = 2$, and $d(v_p) \geq 3$. Then we call the set $\{v_2, v_3, \ldots, v_{p-1}\}$ an *outer chain* of G. The graph in Figure 3 has two outer chains $\{w_4, w_5\}$ and $\{w_8\}$. We call an outer chain $\{v_2, v_3, \ldots, v_{p-1}\}$ of G a *good outer chain* if the outer chain does not contain any vertex of $V(P_k)$. An outer chain $\{v_2, v_3, \ldots, v_{p-1}\}$ of G is a *bad outer chain* if the outer chain contains a vertex of $V(P_k)$. We call an outer edge (u, w) of G a *good outer edge* if $(u, w) \notin E(P_k^*), d(u) \geq 3$ and $d(w) \geq 3$ in G, and $u \in V(G_i)$ or $w \in V(G_i)$.

We say that a plane graph G is *internally triconnected* if G is biconnected and, for any separation pair $\{u, v\}$ of G, u and v are outer vertices and each connected component of $G - \{u, v\}$ contains an outer vertex. In other words, G is internally 3-connected if and only if it can be extended to a 3-connected graph by adding a vertex in an outer face and connecting it to all outer vertices. If a biconnected plane graph G is not internally 3-connected, then G has a separation pair $\{u, v\}$ of one of the three types illustrated in Figure 4. If an internally 3-connected plane graph G is not 3-connected, then G has a separation pair of outer vertices and hence G has a chord-path when G is not a single cycle.

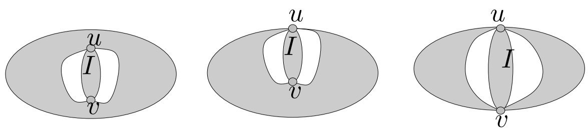


Figure 4: Biconnected graphs which are not internally 3-connected

A *resource bipartitioning* of a connected graph G of $|V| = n$ vertices can be defined as follows. Let $V_r \subseteq V$ be the set of resource vertices and $|V_r| = r$. Let $u_1, u_2 \in V$ be two designated vertices and r_1, r_2 be two natural numbers such that $r_1 + r_2 = r$. Our goal is to find a partition V_1, V_2 of V such that $u_1 \in V_1, u_2 \in V_2$, V_i contains r_i resource vertices and V_i induces a connected subgraph of G for each $i, 1 \leq i \leq 2$. The resource bipartitioning problem is a special case of resource k-partitioning problem, for $k = 2$. We have the following lemmas on resource bipartitioning.

Lemma 2.2 *A resource bipartition of a biconnected graph G can be found in linear time [21, 18].*

Lemma 2.3 *Let s, t be the two base vertices in a connected graph G. If $G \cup \{(s,t)\}$ is biconnected, then a resource bipartition of G can be found in linear time.*

Proof. We can find a resource bipartition of the biconnected graph $G \cup \{(s,t)\}$ in linear time by Lemma 2.2. As the vertices (s,t) would belong to two different partitions, clearly a resource bipartition of $G \cup \{(s,t)\}$ is a resource bipartition of G. $\mathcal{Q.E.D.}$

In section 3, we provide the constructive proof of the existence of a nonseparating ear decomposition through two vertices a, b and avoiding a third vertex c of a triconnected planar graph G for any a, b, c.

3 Nonseparating Ear Decomposition

In this section, we show a constructive proof for the existence of a nonseparating ear decomposition through two vertices a, b and avoiding a third vertex c of a triconnected planar graph G for any a, b, c.

Lemma 3.1 *Let $u_1, u_2, \ldots u_l$ be the outer facial vertices of an internally 3-connected planar graph G and $S = \{u_{p+1}, u_{p+2}, \ldots, u_{q-1}\}$ be an outer chain of G, then $G - S$ is internally triconnected.*

Proof. Let u_p, u_q be the two ends of a minimal chord-path P of G such that $p < q$. Assume for the sake of contradiction that $G - S$ is not internally triconnected. Then $G - S$ has either a cut-vertex or a separation pair $\{u, v\}$. We first consider the case where $G - S$ has a cut-vertex v. Then v must be an outer vertex of $G - S$ and $v \neq u_p, u_q$. Otherwise, G would not be internally triconnected. Then the minimal chord-path P must pass through v, as illustrated in Figure 5, contrary to the definition of a chord path.

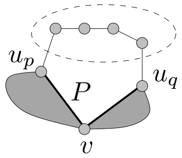


Figure 5: P passes through an outer vertex in $\bar{G}_i$

We now consider the case where $G - S$ has a separation pair $\{u, v\}$ having one of the three types shown in Figure 4. Then $\{u, v\}$ would be a separation pair of G having one of the three types illustrated in Figure 4. Hence G would not be internally triconnected, a contradiction. $\mathcal{Q.E.D.}$

Theorem 3.2 *Let a, b, c be any three vertices in a triconnected planar graph G. Then G has a nonseparating ear decomposition through a, b and avoiding c and it can be found in linear time.*

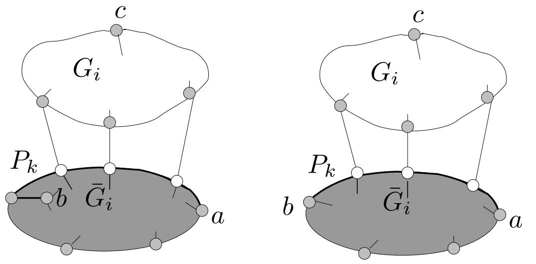
Proof. Without loss of generality we may assume that c is on the outer boundary of G. If $(a, b) \in E(G)$, then let P_k be an inner facial cycle passing through (a, b) and not passing through c. If $(a, b) \notin E(G)$ and both of a, b are outer vertices of G, then let P_k be the outer facial path from a to b not passing through c. If $(a, b) \notin E(G)$ and at least one of a, b is an inner vertex of G, then let P_k be the path from a to b that does not contain any outer edge and go through c. Since G is 3-connected, P_k exists in all of the three cases mentioned above and (nsed1) holds for P_k.

Let $e_1, e_2, \ldots, e_l$ be the neighbours of c where $l = d(c)$. We set $P_0, P_1, \ldots, P_{l-3}$ as $(c, e_1), (c, e_2), \ldots, (c, e_{l-2})$ respectively. We set $P_{l-2} = (e_{l-1}, c, e_l)$. P_{l-2} is the first non-trivial ear and it has exactly one internal vertex which is c. Hence (nsed2) holds. Since $P_0, P_1, \ldots, P_{l-2}$ are open ears and $V_{in}(P_{l-2}) \bigcap V(\bar{G}_{l-2}) = \phi$, (nsed3) holds for $P_j, 0 \leq j \leq l-2$. As for each $j, 0 \leq j \leq l-2, G_j$ is connected, (nsed4) holds for $P_j, 0 \leq j \leq l-2$. As $P_0, P_1, \ldots, P_{l-3}$ are trivial ears and the internal vertex c of P_{l-2} has at least a neighbour e_1 in G_{l-3}, (nsed5) holds for $P_j, 0 \leq j \leq l-2$. Clearly $\bar{G}_j$ or G_j^* is internally triconnected for each $j, 0 \leq j \leq l-2$.

Assume for inductive hypothesis that the ears $P_0, P_1, \ldots, P_i$ for $l-2 \leq i \leq k-2$ have been chosen appropriately so that (nsed2) holds for the first non-trivial ear P_{l-2} and (nsed3), (nsed4), (nsed5) hold for each index $j, 0 \leq j \leq i$. Furthermore G_j^* or $\bar{G}_j$ is internally triconnected for each index $j, 0 \leq j \leq i$. We now show that there is an ear P_{i+1} in $\bar{G}_i$ such that (nsed3), (nsed4), (nsed5) hold and G_j^* or $\bar{G}_j$ is internally triconnected for the index $j = i+1$. Let $u_1, u_2, \ldots u_l$ be the outer facial vertices of $\bar{G}_i$ or G_i^*. We have the following cases to consider.

Case 1: $\bar{G}_i$ or G_i^* is 3-connected.

We only consider the case where $\bar{G}_i$ is 3-connected, since the proof for the case where G_i^* is 3-connected is similar. Since $\bar{G}_i$ is 3-connected, every vertex of $\bar{G}_i$ has degree at least three. There is at least a vertex u on $C_o(\bar{G}_i)$ such that $u \in V(G_i)$ and u has a neighbour w on $C_o(\bar{G}_i)$ such that $(u, w) \notin E(P_k)$. Otherwise, at least one of a, b would be an inner vertex of G and P_k would contain an outer edge of G (see Figure 6(a)) or both of a, b would be outer vertices of G and P_k would contain an inner edge of G (see Figure 6(b)), a contradiction. Then (u, w) is a good outer edge of $\bar{G}_i$, as illustrated in Figure 7. We set $P_{i+1} = (u, w)$. As (u, w) is a trivial open ear, (nsed3) holds for P_{i+1}. As (u, w) is a good outer edge of $\bar{G}_i$, G_{i+1} remains connected and hence (nsed4) holds for P_{i+1}. As (u, w) is a trivial ear, (nsed5) also holds for P_{i+1}. Clearly $\bar{G}_{i+1}$ is internally triconnected.



(a) (b)

Figure 6: P_k is not set according to the rules in base case

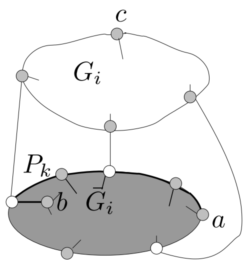


Figure 7: $\bar{G}_i$ or G_i^* is 3-connected.

Case 2: $(a, b) \notin E(G)$ **and** $\bar{G}_i$ **or** G_i^* **is not 3-connected.**
We first consider the case where $(a, b) \notin E(G)$ and $\bar{G}_i$ is not triconnected. Since $i + 1 \leq k - 1$, $\bar{G}_i$ is not a single cycle and hence there is a chord-path of $\bar{G}_i$. Let u_p, u_q be the two ends of a minimal chord-path P of $\bar{G}_i$ such that $p < q$. As $\{u_p, u_q\}$ is a separation pair of $\bar{G}_i$, then $q \geq p + 2$. We have the following two subcases to consider.
Case 2a: P_k is not on $C_o(\bar{G}_i)$.

We have the following two subcases to consider.
(i) $\bar{G}_i$ has an outer chain $\{u_{p+1}, u_{p+2}, \ldots, u_{q-1}\}$.

If $\{u_{p+1}, u_{p+2}, \ldots, u_{q-1}\}$ is a good outer chain, we choose $P_{i+1} = (u_p, u_{p+1}, u_{p+2}, \ldots, u_{q-1}, u_q)$. As $u_p \neq u_q$ and $V_{in}(P_{i+1}) \bigcap V(\bar{G}_{i+1}) = \phi$, (nsed3) holds for P_{i+1}. As G is triconnected and each internal vertex of P_{i+1} has degree two in $\bar{G}_i$, each of the internal vertices of P_{i+1} has a neighbor in G_i. So G_{i+1} is also connected. Thus (nsed4) and (nsed5) hold for P_{i+1}. From Lemma 3.1 $\bar{G}_{i+1}$ is internally triconnected.

We thus assume that $\{u_{p+1}, u_{p+2}, \ldots, u_{q-1}\}$ is a bad outer chain of $\bar{G}_i$. In this case $\bar{G}_i - \{u_p, u_q\}$ has 2 components. There is at least a vertex v in $\bar{G}_i - \{u_p, u_q\}$ in the component not containing $u \in \{u_{p+1}, u_{p+2}, \ldots, u_{q-1}\}$ such that $v \in V(G_i)$ and $d(v) \geq 3$ in $\bar{G}_i$. Otherwise, G would not be triconnected or $d(v) = 2$ in $\bar{G}_i$ and v would be contained in a good outer chain, a contradiction. The vertex v has a neighbour $w \in V(\bar{G}_i) - \{u_{p+1}, u_{p+2}, \ldots, u_{q-1}\}$ such that $d(w) \geq 3$ in $\bar{G}_i$ and $(v, w) \notin E(P_k)$. Otherwise, $d(w) = 2$ in $\bar{G}_i$ and w would be contained in a good outer chain, a contradiction. Then (v, w) is a good outer edge of $\bar{G}_i$, as illustrated in Figure 8. We set $P_{i+1} = (v, w)$. Clearly (nsed3), (nsed4), (nsed5) hold for P_{i+1}, and $\bar{G}_{i+1}$ remains internally triconnected.
(ii) $\bar{G}_i$ has no outer chain.

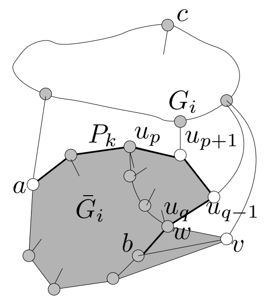


Figure 8: $\{u_{p+1}, u_{p+2}, \ldots, u_{q-1}\}$ is a good outer chain of $\bar{G}_i$.

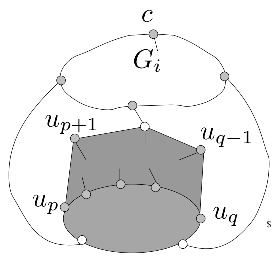


Figure 9: $\{u_{p+1}, u_{p+2}, \ldots, u_{q-1}\}$ is not an outer chain of $\bar{G}_i$

In this case every vertex in $\{u_{p+1}, u_{p+2}, \ldots, u_{q-1}\}$ has degree at least three in $\bar{G}_i$. Otherwise, $\bar{G}_i$ would have an outer chain. Furthermore, there is at least a vertex $u \in \{u_{p+1}, u_{p+2}, \ldots, u_{q-1}\}$ such that $u \in V(G_i)$. Otherwise, $\{u_p, u_q\}$ would be a separation pair of G and G would not be triconnected. If u has a neighbour w on $C_o(\bar{G}_i)$ such that $d(w) \geq 3$ in $\bar{G}_i$ and $(u, w) \notin E(P_k)$, we set $P_{i+1} = (u, w)$. Then (u, w) is a good outer edge of $\bar{G}_i$ and (nsed3), (nsed4), (nsed5) hold for P_{i+1}, and $\bar{G}_{i+1}$ remains internally triconnected. If u has no such neighbour w, then there is at least a vertex v in $\bar{G}_i - \{u_p, u_q\}$ in the component not containing $u \in \{u_{p+1}, u_{p+2}, \ldots, u_{q-1}\}$ such that $v \in V(G_i)$ and $d(v) \geq 3$ in $\bar{G}_i$. Otherwise, G would not be triconnected or $d(v) = 2$ in $\bar{G}_i$ and v would be contained in a good outer chain, a contradiction. The vertex v has a neighbour $w \in V(\bar{G}_i) - \{u_{p+1}, u_{p+2}, \ldots, u_{q-1}\}$ such that $d(w) \geq 3$ in $\bar{G}_i$ and $(v, w) \notin E(P_k)$. Otherwise, $d(w) = 2$ in $\bar{G}_i$ and w would be contained in a good outer chain, a contradiction. Then (v, w) is a good outer edge of $\bar{G}_i$. We set $P_{i+1} = (v, w)$. Clearly (nsed3), (nsed4), (nsed5) hold for P_{i+1}, and $\bar{G}_{i+1}$ remains internally triconnected.

Case 2b: P_k is on $C_o(\bar{G}_i)$ or $C_o(G_i^*)$.

If removal of P_i has left P_k on $C_o(\bar{G}_i)$, we augment $\bar{G}_i$ to G_i^* as stated before. Otherwise, P_k has already been on $C_o(\bar{G}_i)$ and $\bar{G}_i$ has already been augmented to G_i^*. Hence it is sufficient to consider only the case for G_i^*. We have the following subcases to consider.

(i) G_i^* has an outer chain $\{u_{p+1}, u_{p+2}, \dots, u_{q-1}\}$.

In this case $\{u_{p+1}, u_{p+2}, \dots, u_{q-1}\}$ is a good outer chain. Otherwise, P_k is not on $C_o(\bar{G}_i)$. We choose $P_{i+1} = (u_p, u_{p+1}, u_{p+2}, \dots, u_{q-1}, u_q)$. Clearly (nsed3), (nsed4), (nsed5) hold for P_{i+1}, and $\bar{G}_{i+1}$ remains internally triconnected.

(ii) G_i^* has no outer chain.

In this case every vertex in $\{u_{p+1}, u_{p+2}, \dots, u_{q-1}\}$ has degree at least three in G_i^*. Otherwise, G_i^* would have an outer chain. Furthermore, there is at least a vertex $u \in \{u_{p+1}, u_{p+2}, \dots, u_{q-1}\}$ such that $u \in V(G_i)$. Otherwise, $\{u_p, u_q\}$ would be a separation pair of G and G would not be triconnected. Clearly u has a neighbour $w \in \{u_p, u_{p+1}, u_{p+2}, \dots, u_{q-1}, u_q\}$ such that $(u, w) \notin E(P_k)$. Then (u, w) is a good outer edge of G_i^*. We set $P_{i+1} = (u, w)$. Clearly (nsed3), (nsed4), (nsed5) hold for P_{i+1}, and $\bar{G}_{i+1}$ remains internally triconnected.

Now it remains to consider the case where $(a, b) \notin E(G)$ and G_i^* is not triconnected. In this case, we choose P_{i+1} similar to the subcase (2b) with the exception that we do not need any augmentation.

Case 3: $(a, b) \in E(G)$ **and** $\bar{G}_i$ **is not 3-connected.**

In this case we choose P_{i+1} similar to the case where $(a, b) \notin E(G)$ and G_i^* is not triconnected.

Thus the existence of a nonseparating ear decomposition of G through a, b and avoiding c for any a, b, c is proven. An algorithm for finding a nonseparating ear decomposition based on the proof above can be implemented. We have to keep track of outer chains, minimal chord-paths and candidate degree three vertices of $\bar{G}_i$. Each face is traversed at most a constant number of times. So run time is linear. Hence the Theorem 3.2 follows. $\mathcal{Q.E.D.}$

We call the algorithm obtained from this constructive proof for the existence of a nonseparating ear decomposition of G through a, b and avoiding c for any a, b, c Algorithm *Find_Decomposition*.

Lemma 3.3 *Let a, b, c be three vertices in a triconnected planar graph G and n, m denote the number of vertices and the number of edges in G respectively. Then the nonseparating ear decomposition of G through a, b and avoiding c produced by Algorithm Find_Decomposition has the following properties.*

(a) if $(a, b) \in E(G)$, then

- *(i) length of any ear is at least one and at most the length of the longest facial cycle of G.*
- *(ii) the number of ears is $m - n + 1$.*

(b) if $(a, b) \notin E(G)$, then

- *(i) length of any ear is at least one and at most the larger one of the length of P_k and one less than the length of the longest facial cycle of G.*
- *(ii) the number of ears is $m - n + 2$.*

Proof. (a)(i) From the constructive proof of Theorem 3.2, it can be found that each ear $P_i, i \neq k$, is either a single edge or a path with all of its internal vertices belonging to an outer chain of $\bar{G}_{i-1}$. So the length of a non-trivial ear P_i can be at most 1 less than that of the longest facial cycle. But as $(a, b) \in E(G)$, the ear P_k is set as an inner facial cycle passing through the edge (a, b) and not passing through c. Therefore, P_k may have the length of the longest facial cycle. Hence length of any ear is at least one and at most the length of the longest facial cycle of G.

(a)(ii) We employ an induction on n. A graph must have at least $n = 4$ vertices and $m = 6$ edges to be 3-connected. A nonseparating ear decomposition of the graph in Figure 10(i) is as follows. $P_0 = c, a.P_1 = d, c, b$ and $P_{k=2} = a, d, b, a$. So we have $3 = 6-4+1 = m - n + 1$ ears for any a, b, c of G.

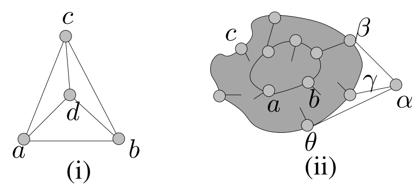


Figure 10: (i) A triconnected graph with four vertices and six edges (ii) G^α with n' vertices and m' edges.

Assume that $n \geq 5$ and the result is true for all triconnected planar graphs having n vertices. Without loss of generality we may assume that c is on the outer boundary of G. We now add a vertex α on the outer face of G. Let $G^\alpha = G \bigcup \{\alpha\}$. To make G^α 3-connected without losing planarity, $d(\alpha)$ number of edges are added from α to its $d(\alpha)$ number of neighbours on $C_o(G)$, where $3 \leq d(\alpha) \leq |C_o(G)|$. G^α has $n' = n + 1$ vertices and $m' = m + d(\alpha)$ edges. To find a nonseparating ear decomposition of G^α (see Figure 10(ii)) we use the decomposition as in G until the vertex β or γ or θ is contained in G_i. Then the following ears are chosen in the decomposition of

G^{α}. $P_x = \beta, \alpha . P_{x+1} = \gamma, \alpha, \theta$. Then we take the ears as in the decomposition in G. Hence there are $d(\alpha) - 1$ new ears more than those in the decomposition of G. So total number of ears in the nonseparating ear decomposition of G_{α} is $m - n + 1 + d(\alpha) - 1 = m + d(\alpha) - (n+1) + 1 = m^{'} - n^{'} + 1$. Hence the induction holds.

(b)(i) Now as $(a, b) \notin E(G)$, then the ear P_k is a path from a to b. Clearly length of any ear is at least one and at most maximum of the length of P_k and one less than the length of the longest facial cycle of G.

(b)(ii) Consider the graph $G_i^{'} = (V(G_i), E(G_i) \bigcup (a, b))$. From Lemma 3.3(a)(ii) $G_i^{'}$ has $m^{'} - n + 1$ ears, where $|E(G_i^{'})| = m^{'} = m + 1$. Hence the number of ears is $m - n + 2$. $\mathcal{Q.E.D.}$

In section 4, we provide an algorithm to find a resource 3-partition of a graph G having a nonseparating ear decomposition through two vertices a, b and avoiding a third vertex c for any a, b, c.

4 Resource Tripartition

In this section we give an algorithm to find a resource 3-partition of a planar graph G having a nonseparating ear decomposition through two vertices a, b and avoiding a third vertex c for any a, b, c.

Let H_i be the subgraph induced by $V_{in}(P_0) \cup V_{in}(P_1) \cup \ldots \cup V_{in}(P_i)$ and $\bar{H}_i$ be the subgraph induced by $V - V_{in}(P_0) \cup V_{in}(P_1) \cup \ldots \cup V_{in}(P_i)$. We have the following lemma.

Lemma 4.1 *Let $P_0, P_1, \ldots, P_q$ be the non-trivial ears of a nonseparating ear decomposition of a planar graph G through a, b and avoiding c. Then for any $W \subset V_{in}(P_i)$, $H_i - W$ is connected for $i, 0 \leq i \leq q$.*

Algorithm *Resource_Tripartition*
Input: A planar graph $G = (V, E)$ which has a nonseparating ear decomposition through two vertices a, b and avoiding a third vertex c for any a, b, c, three designated distinct vertices u_1, u_2, u_3 and three natural numbers r_1, r_2, r_3 such that $\sum_{i=1}^{3} r_i = r$.
Output: A resource 3-partition of G.
begin
Find a nonseparating ear decomposition $P_0, P_1, \ldots, P_k$ through u_1, u_2 and avoiding u_3 of G;
Let $P_0, P_1, \ldots, P_q$ be the non-trivial ears of this nonseparating ear decomposition of G;
Let i be the minimum integer such that $V_{in}(P_0) \cup V_{in}(P_1) \cup \ldots \cup V_{in}(P_i)$ contains at least r_3 resource vertices, where each P_i is a non-trivial ear, $0 \leq i \leq q$;
Let e be the excess number of resource vertices in $V_{in}(P_0) \cup V_{in}(P_1) \cup \ldots \cup V_{in}(P_i)$ over r_3;
There are the following two cases: (1) $e = 0$, and (2) $e \geq 1$;
Case 1: $e = 0$.
{ In this case, H_i contains r_3 resource vertices, and $\bar{H}_i$ contains $r_1 + r_2$ resource vertices. }
Let $V_3 = H_i$;
Find a resource bipartition V_1, V_2 of the biconnected graph $\bar{H}_i \cup \{(u_1, u_2)\}$ such that $u_1 \in V_1, u_2 \in V_2$, V_1 contains r_1 resource vertices and V_2 contains r_2 resource vertices, and both V_1, V_2 induce connected subgraphs;
{ We can find a resource bipartition of $\bar{H}_i \cup \{(u_1, u_2)\}$ in linear time by Lemma 2.3}
return V_1, V_2, V_3 as a resource 3-partition of G.
Case 2: $e \geq 1$.
{ In this case, H_i contains $r_3 + e$ resource vertices, and $\bar{H}_i = \bar{H}_{i-1} - V_{in}(P_i)$ contains $r_1 + r_2 - e$ resource vertices. Since $e \geq 1$,$V_{in}(P_i)$ contains at least two resource vertices, $|V_{in}(P_i)| \geq 2$ and hence $V_{in}(P_i)$ is an outer chain of $\bar{H}_{i-1}$. }
Let $C_o(\bar{H}_{i-1}) = w_1, w_2, \ldots, w_h, w_1$ where $w_1 = u_1$;
Assume that $V_{in}(P_i) = \{w_{p+1}, w_{p+2}, \ldots, w_{q-1}\}$ is an outer chain of $\bar{H}_{i-1}$;
Find an st-numbering $v_1, v_2, \ldots, v_z$ of $\bar{H}_i \cup \{(u_1, u_2)\}$ such that $s = v_1 = u_1$ and $t = v_z = u_2$;
Let $w_p = v_{p'}$ and $w_q = v_{q'}$;
Assume that $p^{'} < q^{'}$, otherwise, interchange the roles of u_1 and u_2;
Let $v_1, v_2, \ldots, v_{p'}$ contain x resource vertices;
There are the following three subcases: (a) $r_1 \leq x$, (b) $x + e \leq r_1$, and (c) $x < r_1 < x + e$;
Subcase 2a: $r_1 \leq x$.
{ In this subcase, the last $e^{'}$ vertices containing e resource vertices in the outer chain $V_{in}(P_i)$ are added to $\bar{H}_i$ as the deficient e resource vertices. }
Let $V_1 = \{v_1, v_2, \ldots, v_{n_1}\}$ be the first n_1 vertices containing r_1 resource vertices in the st-numbering of $\bar{H}_i \cup \{(u_1, u_2)\}$;
Let $V_2^{'} = \{v_{n_1+1}, v_{n_1+2}, \ldots, v_z\}$ be the remaining vertices containing $r_2 - e$ resource vertices in $\bar{H}_i$, where $w_q = v_{q'} \in V_2^{'}$;
{ By the fact (st1) of an st-numbering both V_1 and $V_2^{'}$ induce connected graphs. }
Let $W = \{w_{q-1}, w_{q-2}, \ldots, w_{q-e'}\}$ be the set of the last $e^{'}$ vertices containing e resource vertices in $V_{in}(P_i)$;
Let $V_2 = V_2^{'} \cup W$;
{ Since w_{q-1} is adjacent to $w_q \in V_2^{'}, V_2$ induces a connected graph with r_2 resource vertices. }

Let $V_3 = H_i - W$;
{ V_3 is connected by Lemma 4.1, and has r_3 resource vertices. }
return V_1, V_2, V_3 as a resource 3-partition of G.
Subcase 2b: $x + e \leq r_1$.
{In this subcase, the first $e^{'}$ vertices containing e resource vertices in $V_{in}(P_i)$ are added to $\bar{H}_i$ as the deficient e resource vertices. }
Let $V_1^{'} = \{v_1, v_2, \ldots, v_{n_1}\}$ be the first n_1 vertices containing $r_1 - e$ resource vertices in the st-numbering of $\bar{H}_i \cup \{(u_1, u_2)\}$, where $w_p = v_{p'} \in V_1^{'}$;
Let $V_2 = \{v_{n_1+1}, v_{n_1+2}, \ldots, v_z\}$ be the remaining vertices containing r_2 resource vertices in $\bar{H}_i$;
{ By the fact (st1) of an st-numbering both $V_1^{'}$ and V_2 induce connected graphs. }
Let $W = \{w_{p+1}, w_{p+2}, \ldots, w_{p+e'}\}$ be the set of the first $e^{'}$ vertices containing e resource vertices in $V_{in}(P_i)$;
Let $V_1 = V_1^{'} \cup W$;
{ Since w_{p+1} is adjacent to $w_p \in V_1^{'}$, V_1 induces a connected graph with r_1 resource vertices. }
Let $V_3 = H_i - W$;
{ V_3 is connected by Lemma 4.1, and has r_3 resource vertices. }
return V_1, V_2, V_3 as a resource 3-partition of G.
Subcase 2c: $x < r_1 < x + e$.
{ In this subcase, $e \geq 2$; the first b vertices containing $r_1 - x$ resource vertices and the last c vertices containing $e-(r_1-x)$ resource vertices in $V_{in}(P_i)$ are added to $\bar{H}_i$ as the deficient e resource vertices. }
Let $W = \{w_{p+1}, w_{p+2}, \ldots, w_{p+b}\}$ be the set of the first b vertices containing $r_1 - x$ resource vertices in $V_{in}(P_i)$;
Let $W^{'} = \{w_{q-1}, w_{q-2}, \ldots, w_{q-c}\}$ be the set of the last c vertices containing $e - (r_1 - x)$ resource vertices in $V_{in}(P_i)$;
{ Since $|W| + |W^{'}| = b + c < |V_{in}(P_i)|$, $W \cap W^{'} = \phi$, $|W \cup W^{'}| = b + c$ and $W \cup W'$ contains e resource vertices. }
Let $V_1 = \{v_1, v_2, \ldots, v_{p'}\} \cup W$;
Let $V_2 = \{v_{p'+1}, v_{p'+2}, \ldots, v_z\} \cup W^{'}$;
{ V_1 and V_2 contains r_1 and r_2 resource vertices respectively, $w_p = v_{p'} \in V_1$, $w_q = v_{q'} \in V_2$, and both V_1 and V_2 induce connected subgraphs. }
Let $V_3 = H_i - W \cup W^{'}$;
{ V_3 is connected by Lemma 4.1, and has r_3 resource vertices. }
return V_1, V_2, V_3 as a resource 3-partition of G.
end;

Since st-numbering can be obtained in $O(n)$ time by Lemma 2.1, the running time of the above algorithm is $O(n)$ if a nonseparating ear decomposition of G through u_1, u_2 and avoiding u_3 can be found in linear time. Thus we have the following theorem.

Theorem 4.2 *A planar graph G having a nonseparating ear decomposition through two vertices a, b and avoiding a third vertex c for any a, b, c has a resource 3-partition. Furthermore, if a nonseparating ear decomposition of G through two vertices a, b and avoiding a third vertex c for any a, b, c can be found in linear time, a resource 3-partition of G can be found in linear time.*

Hence from Theorem 3.2, we obtain a linear algorithm to find a resource 3-partition of a 3-conneced planar graph G by using the nonseparating ear decomposition of G through a, b and avoiding c described in section 3.

5 Conclusion

In this paper, we present a linear-time algorithm for finding a resource tripartition of a planar graph for which a nonseparating ear decomposition through two vertices a, b and avoiding a third vertex c for any a, b, c can be found in linear time. We also present a linear algorithm for constructing a nonseparating ear decomposition of a triconnected planar graph. The interesting features of the nonseparating ear decomposition produced by our algorithm regarding the number of ears and bounds on length of ear can have significant applications. Using our algorithm for finding a nonseparating ear decomposition, we obtain a linear algorithm to find resource tripartitions of triconnected planar graphs. Applying our algorithm for finding a nonseparating ear decomposition with an algorithm in [3], we can also achieve a linear algorithm to find three "independent spanning trees" in 3-connected planar graphs rooted at a vertex r. However, the following problems related to resource partitioning are still open.

(a) Developing algorithms for finding resource k-partitions of graphs for $k \geq 4$.

(b) Developing algorithms to find resource k-partitions of graphs for $k \geq 2$ where resources are specified for the partitions.

Acknowledgement

Jou et. al. presented a linear algorithm for finding a nonseparating ear decomposition of a triconnected planar graph [11] but their description of the algorithm is ambiguous and does not have any proof. However, their work served as an inspiration to our research. We thank the anonymous referees for their useful comments.

References

[1] Battista, G. D., Tamassia, R., and Vismara, L. Output sensitive reporting of disjoint paths. *Algorithmica*, 23:302–340, 1999.

[2] Chen, H. L., Liao, C. C., Lu, H. I., and Yen, H. C. Some applications of orderly spanning trees in graph drawing. In *Proc. Graph Drawing 2002 (GD 2002), LNCS*, volume 2528, pages 332–343, 2002.

[3] Cheriyan, J. and Maheshwari, S. N. Finding nonseparating induced cycles and independent spanning trees in 3-connected graphs. *Journal of Algorithms*, 9:507–537, 1988.

[4] Chiang, Y. T., Lin, C. C., and Lu, H. I. Orderly spanning trees with applications to graph encoding and graph drawing. In *Proc.of 12th Annual ACM-SIAM symp. on Discrete Algorithms*, pages 506–515, 2001.

[5] Chrobak, M. and Kant, G. Convex grid drawings of 3-connected planar graphs. *International Journal of Computational Geometry and Appl.*, 7:211–223, 1997.

[6] Chrobak, M. and Payne, T. A linear-time algorithm for drawing planar graphs on a grid. *Inform. Process. Lett.*, 54:241–246, 1995.

[7] Dyer, M. E. and Frieze, A. On the complexity of partitioning graphs into connected subgraphs. *Discrete Applied Mathematics*, 10:139–153, 1985.

[8] Even, S. and Tarjan, R. Computing an *st*-numbering. *Theoretical Computer Science*, 2:339–344, 1976.

[9] Györi, E. On division of connected subgraphs. In *Proc. of 5th Hungarian Combinational Coll.*, pages 485–494, 1978.

[10] He, X. Grid embedding of 4-connected plane graphs. *Discrete and Computational Geometry*, 17:339–358, 1997.

[11] Jou, L., Suzuki, H., and Nishizeki, T. A linear algorithm for finding a nonseperating ear decomposition of triconnected planar graphs. In *Tech. Rept. of Info. Proc. Soc. of Japan AL 40-3*, 1994.

[12] Kant, G. Drawing planar graphs using the canonical ordering. *Algorithmica*, 16:4–32, 1996.

[13] Karim, M. R., Nahiduzzaman, K. M., and Rahman, M. S. A linear-time algorithm for k-partitioning doughnut graphs. *INFOCOMP Journal of Computer Science*, 8(1):8–13, 2009.

[14] Lovász, L. A homology theory for spanning trees of a graph. *Acta Math, Acad. Sci. Hunger.*, 30:241–251, 1977.

[15] Ma, J. and Ma, S. An $o(k^2n^2)$ algorithm to find a k-partition in a k-connected graph. *J. Comput. Sci. Tech.*, 9:86–91, 1994.

[16] Miura, K., Azuma, M., and Nishizeki, T. Canonical decomposition, realizer, schnyder labeling and orderly spanning trees of plane graphs. In *10th Annual International Conference, COCOON 2004*, pages 309–318, 2004.

[17] Nakano, S., Rahman, M. S., and Nishizeki, T. A linear-time algorithm for four-partitioning four-connected planar graphs. *Inform. Process. Lett.*, 62:315–322, 1997.

[18] Shams, Z., Ferdous, S., Sultana, K. Z., and Rahman, M. S. On resource bipartitioning problem. In *Proc. of 4th ICECE*, pages 308–311, 2006.

[19] Suzuki, H., Takahashi, N., and Nishizeki, T. A linear algorithm for bipartition of biconnected graphs. *Inform. Process. Lett.*, 33:227–232, 1990.

[20] Suzuki, H., Takahashi, N., Nishizeki, T., Miyano, H., and S.Ueno. An algorithm for tripartitioning 3-connected graphs. *J. Inform. Process. Soc. Japan*, 31:584–592, 1990.

[21] Wada, K. and Kawaguchi, K. Efficient algoritms for triconnected graphs and 3-edge-connected graphs. In *Proc. of 19th International Workshop on Graph-Theoretic Concepts in Computer Science(WG), LNCS*, volume 790, pages 132–143, 1994.

[22] Wada, K., Takaki, A., and Kawaguchi, K. Efficient algorithms for a mixed k-partition problem of graphs without specifying bases. In *Proc.of 20th International Workshop on Graph Theoretic Concepts in Computer Science (WG), Lecture Notes in Computer Science*, volume 903, pages 319–330, 1995.