Partitioning the Edges of a Planar Graph into Two Partial K-Trees

Ehab S. Elmallah and Charles J. Colbourn

Department of Computing Science
University of Alberta,
Edmonton, Canada
and
Department of Combinatorics and Optimization
University of Waterloo,
Waterloo, Canada.

ABSTRACT

In this paper we prove two results on partitioning the edges of a planar graph into two partial k-trees, for fixed values of k. Interest in this class of partitioning problems arises since many intractable graph and network problems admit polynomial time solutions on k-trees and their subgraphs (partial k-trees).

The first result shows that every planar graph is a union of two partial 3-trees. Furthermore, such a partitioning can be computed in linear time. Second, we show a recursive procedure to construct an infinite family of planar graphs in which every member does not admit a partitioning into a partial 1-tree (forest) and a partial 2-tree (series-parallel graph).

1. Introduction

The classes of k-trees, $k \geq 1$, have been introduced in [BP71] as generalizations of trees as follows. The complete graph on k vertices, denoted K_k, is a $k-tree$. Furthermore, if G is a $k-tree$ then so is the graph obtained from G by adjoining a new vertex, and making it adjacent to every vertex in a complete subgraph on k vertices of G. Hence, trees are 1-trees. Moreover, k-trees are all triangulated graphs (e.g. see [Go80]).

A *partial* k-tree is a subgraph of a k-tree. Examples of well known families of such graphs include: *outerplanar* graphs ($\subset$ partial 2-trees), $series-parallel$ graphs (partial 2-trees) (e.g. [WC83]), *Halin* graphs ($\subset$ partial 3-trees) (e.g. [EC86]) and $\Delta - Y - reducible$ graphs ($\subset$ partial 4-trees) [EC85]. Note that, for a given k, the class of partial k-trees is exactly the graphs of *tree-width* at most k [RS86].

The class of planar graphs is not contained in the class of partial k-trees for any fixed k. In particular we note that, for any given k, a 2-dimensional $(k+i \times k+i)$ *grid* graph H_{k+i}, $i \geq 1$, is not a partial k-tree. The above remark follows since every k-tree G on n vertices, $n > k$, contains a complete subgraph on k vertices whose removal splits G into components, each of size at most $n/2$ [GRE84]. Hence, if G' is a partial k-tree then it contains a subset of k vertices whose removal splits G' in the above way. However, no subset of k vertices in H_{k+i} satisfies this latter condition. It then follows that H_{k+i} is not a partial k-tree.

Our interest in partitioning the edges of a planar graph into partial k-trees arises in the following ways. First, we recall that a number of problems have been shown to be NP-complete on planar graphs (e.g. see [GJ79]). Examples of such problems include: determining the *chromatic number* of a planar graph, finding a minimum cost *Steiner tree* in planar networks and determining whether a planar graph is Hamiltonian. In addition, many other problems remain open and challenging on planar graphs.

On the other hand, the classes of partial k-trees (for $k = 1, 2, \cdots$) form a hierarchy of graphs having unified paradigms for solving a variety of hard problems in polynomial times (e.g. see [AP84b], [Jo85] and [RS86] and the references therein). Hence, one may combine existing polynomial time algorithms for partial k-trees with suitable decomposition schemes for planar graphs to devise approximate algorithms for planar graphs.

The second motivation arises from a number of existing results and open problems mentioned in the context of studying other graph theoretic concepts. In particular, recall that the *arboricity* of a graph G is the minimum number of spanning forests into which $E(G)$ can be decomposed. Similarly, the *outerthickness* of G is the minimum number of *outerplanar* graphs into which $E(G)$ can be decomposed.

The first result of interest is due to [Tu61] and independently [Na61] on the arboricity of a graph G. A generalization of that result appears in [Ed65]. The result proves that the arboricity of G equals $\max \left\lceil (|E(G')|)/(|V(G')| - 1) \right\rceil$, where the maximum is taken over all induced subgraphs G' of G. This latter result together with the inequality $|E(G')| \le 3|V(G')| - 6$ for any planar graph G' implies that 3 spanning forests suffice to cover the edges of any planar graph. Hence, it becomes natural to investigate cases where two partial k-trees suffice to cover the edges of a planar graph.

It is then interesting to note that the outerthickness of a graph G determines an upper bound on the minimum number of partial 2-trees necessary to cover $E(G)$. This latter observation follows since every outerplanar graph is a partial 2-tree. The problem of determining the outerthickness of planar graphs has been suggested, among other problems, in the work of Chartrand et al. [CGH71] on graph properties unifying various graph theoretic concepts. In this latter paper, the following conjecture has been mentioned.

Conjecture [CGH71]: *The Outerthickness of any planar graph is at most 2.*

In fact, a second result of Tutte on the Hamiltonicity of 4-connected planar graphs [Tu56] leads to a rather interesting consequence: the outerthickness of a 4-connected planar graph equals 2. To explain this, fix an embedding of any given 4-connected planar graph G in the plane. Denote one of its Hamiltonian cycles by C. Now, C splits the plane into two regions which we denote by R_{in} and R_{out}. Furthermore, denote the set of chords that lies in R_{in} (or R_{out}) by E_{in} (respectively, E_{out}). The result now follows since each of the following two graphs $G_{in} = (V(G), E(C) \bigcup E_{in})$ and $G_{out} = (V(G), E(C) \bigcup E_{out})$ is an outerplanar graph. In addition, any edge of $E(G)$ belongs to at least one such subgraph. The above observation implies that every 4-connected planar graph is a union of two partial 2-trees. The work presented in this paper does not resolve the above conjecture; however, it narrows down two of its sides further.

The rest of this paper is organized as follows. Section 2 introduces some basic definitions required throughout this paper. In sections 3 we prove that every planar graph is coverable by two partial 3-trees of particular structure. Moreover, such a covering can be computed in linear time using a linear time planarity testing algorithm such as the one developed in [HT74]. An overview of the partitioning algorithm is presented in section 4. Thirdly, we show in section 5 that the above result can not be further strengthened to a covering by a forest (partial 1-tree) and a partial 2-tree. The proof shows an infinite family of planar graphs for which such a covering does not exist. Finally, we draw some conclusions in section 6.

2. Definitions and Notations

Throughout this paper a graph $G=(V(G),E(G))$ is considered to be finite and loopless. The degree of a vertex v in a graph G is denoted $deg_G(v)$, its set of neighbours is $N_G(v)$ and its set of incident edges is $\Gamma_G(v)$. In addition, if $X \subseteq V(G)$ then $N_G(X)=(\bigcup_{v\in X} N_G(v))\backslash X$. The minimum degree in G is denoted δ_G. The subgraph induced by a subset E' of edges (V' of vertices) is denoted $G[E']$ (respectively, $G[V']$). A $k-clique$ is a complete graph on k vertices. Subscripts of a variable and qualifiers of a certain graph are at times omitted when no confusion can arise.

A $k-leaf$ of a k-tree G is a vertex whose neighbours induce a k-clique. The following properties are easy consequences of the above definition (for other properties see also [Go80] and [AP84a]):

Lemma 1.

1) *If G is a k-tree on n vertices and m edges then* $m = kn - k(k+1)/2$.

2) *Every k-tree that is not a complete graph has at least two nonadjacent k-leaves.*

The classes of partial k-trees, for $k=1,2,\cdots$, form a hierarchy of graphs since any $(k-i)$-tree, $k>i\geq 1$, is a partial k-tree. A *leaf* of a partial k-tree G is a vertex x, $deg_G(x)\leq k$, which is a k-leaf in some embedding of G in a k-tree G', $G\subseteq G'$. By Lemma 1 (2), every partial k-tree that is not a complete graph has at least two nonadjacent leaves.

A *complete elimination* of a vertex v from G is the elimination of v and its incident edges and the addition of the necessary edges to complete $N(v)$ to a complete subgraph; if $deg_G(v)\leq k$ then the graph obtained by eliminating v in this way is denoted $P_k(G,v)$. The composition of two complete eliminations $P_k(P_k(G,v_1),v_2)$ is denoted $P_k(G,<v_1,v_2>)$. A *k-complete elimination sequence (k-CES)* of a graph G is an ordering of $V(G)$ such that $deg_G(v_1)\leq k$ and for any i, $2\leq i<n$, the degree of v_i in $P_k(G,<v_1,\cdots,v_{i-1}>)$ is at most k. Thus, a graph is a partial k-tree if and only if it has a $k-CES$.

The following results can be easily derived from the above definitions; for completeness we sketch a proof.

Lemma 2.

i) *Given a k-tree G and a complete subgraph H of G,* $H \cong K_k$, *there exists an ordering S of* $V\backslash V(H)$ *such that* $P_k(G,S)=H$.

ii) *Let* G_1 *and* G_2 *be two k-trees. Let* H_1 *and* H_2 *be two k-cliques in* G_1 *and* G_2*, respectively. Then the graph G obtained from* G_1 *and* G_2 *by identifying* $V(H_1)$ *and* $V(H_2)$ *pairwise is a k-tree.*

Proof. To show (i) observe that if $G \not\equiv H$ then, by Lemma 1 (2), G has at least one k-leaf x such that $\Gamma(x) \cap E(H) = \emptyset$. Hence, the required sequence S can always be constructed. To show (ii) let S_i, $i = 1, 2$, be an ordering of $V(G_i) \backslash V(H_i)$ such that $P_k(G_i, S_i) = H_i$. Such sequences exist by part (i). Then, $S = < S_1, S_2 >$ is a prefix of a k-complete elimination sequence that reduces G to H_1 (or H_2). Hence, G is a k-tree. □

Similarly, we have the following result for partial k-trees:

Corollary 1.

i) *Given a partial k-tree G and a complete subgraph H of G on at most k vertices, there exists an ordering S of* $V \backslash V(H)$ *such that* $P_k(G, S) = H$.

ii) *Let* G_1 *and* G_2 *be two partial k-trees. Let* H_1 *and* H_2 *be two complete subgraphs on at most k vertices in* G_1 *and* G_2*, respectively. Then the graph G obtained from* G_1 *and* G_2 *by identifying* $V(H_1)$ *and* $V(H_2)$ *pairwise is a partial k-tree.*

We say that G is a partial (k_1, k_2)-tree if the edges of G can be covered by two subgraphs: one is a partial k_1-tree and the other is a partial k_2-tree.

3. Covering Planar Graphs by Two Partial 3-Trees

In this section we show that every planar graph is a partial (3,3)-tree. The strategy is to partition G into a number of partial 3-trees, each has a particular structure, called an *IO-graph* hereafter. A partitioning of $E(G)$ into two partial 3-trees is then formed by taking a disjoint union of a family of the obtained IO-subgraphs. Specifically, we have:

Definition 1. *A 2-connected planar graph H is an IO-graph if H is an outerplanar graph or H has an embedding in the plane such that:*
(1) the removal of some independent subset of vertices, say $V_I(H)$*, of* $V(H)$ *leaves a 2-connected outerplanar subgraph* H' *and*
(2) the exterior face of H'*,* $ext(H')$*, is the same as the exterior face of H,* $ext(H)$*, relative to that particular embedding.*

Accordingly, if C is the unique Hamilton cycle of H' then each edge of H is either an edge of C, a chord of C or it has one vertex in $V_I(H)$ and the other in $V_C(H)$ (for example, see Figure 1).

The following lemma is an easy consequence of the above definition, and for completeness we include a proof.

Lemma 3. *Every IO-graph* H *is a partial 3-tree.*

Proof. The lemma follows easily if H is an outerplanar graph. So, assume that $V_I(H) \neq \emptyset$. The proof proceeds by induction on $|V(H)|$. The case when $H \subseteq K_4$ follows immediately. Suppose it holds inductively for all IO-graphs having less than n vertices, $n \geq 3$. Let H be an IO-graph on exactly n vertices. By definition of IO-graphs $\delta_H \geq 2$. If H has a vertex v of degree 2 then $P_2(H, v)$ is an IO-graph. Otherwise, $\delta_H \geq 3$. In this case, one may verify that H has at least one vertex, say v in $V_I(H)$ such that v has at least 3 consecutive neighbours, say (u_1, u_2, u_3), occurring on the Hamilton cycle of $H \backslash V_I(H)$ in that order, where $deg_H(u_2) = 3$. Again, the graph $P_3(H, u_2)$ is an IO-graph. In each of the above cases the reduced graph has fewer than n

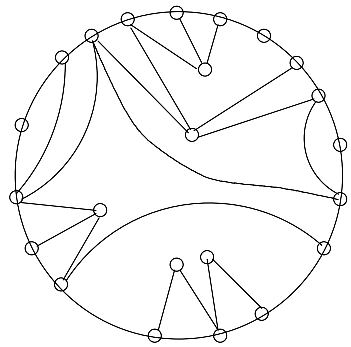

Figure 1. An IO-graph.

vertices and the lemma follows by induction. □

We note that planar partial 3-trees has been recently characterized using 2 forbidden minors [EC88]. A more general result appears in [APC86] where the class of partial 3-trees has been characterized using four forbidden minors. Using such characterization results one may derive Lemma 3 easily.

Theorem 1. *Let G be a planar graph. Then $E(G)$ can be partitioned into E_1 and E_2 such that $G[E_1]$ and $G[E_2]$ are partial 3-trees consisting of disjoint unions of trees and IO-graphs.*

Proof. Fix an embedding of G in the plane. Partition $V(G)$ into levels recursively as follows. First, let V_0 be the set of vertices lying on the exterior face of G, $ext(G)$. Subsequently, the i*th* subset V_i in the partition exists if $V \backslash \{V_0, \cdots, V_{i-1}\} \neq \emptyset$; it corresponds to the set of vertices lying on the exterior face of the possibly disconnected graph $G_i = G \backslash (V_0 \cup \cdots \cup V_{i-1})$. Denote by l the index of the last subset obtained in that way. Thus, $V(G) = \bigcup_{i=0}^{l} V_i$.

If $V_0 = V(G)$ then let $H_0 = G$. Otherwise, $l > 0$. Let H_i, $l > i \geq 0$, be the subgraph of G whose set of edges $E(H_i)$ equals $\{(x, y) | x \in V_i$ and $y \in V_i$ *or* $V_{i+1}\}$. Also, if $l > 0$ set $E(H_l) = \{(x, y) | x, y \in V_l\}$. Each component of H_i that does not contain a cut-vertex is either an edge or an IO-graph. By Lemma 3, each such 2-connected component is a partial 3-tree. By Corollary 1 (2) each H_i is a partial 3-tree. Note that $V(H_i) \cap V(H_{i+2}) = \emptyset$ for $0 \leq i \leq l-2$. Now, set $E_1 = \bigcup_{0 \leq i \leq l} E(H_i)$, i is even, and $E_2 = \bigcup_{0 \leq i \leq l} E(H_i)$, i is odd. The theorem then follows since each of the two subgraphs induced by E_i, $i = 1, 2$, is a union of a number of vertex-disjoint partial 3-trees. □

4. The Partitioning Algorithm

In this section we use the constructive proof of Theorem 1 to devise a linear time algorithm for computing such a partitioning. We use a *planar adjacency list* $A[1 \cdots n]$ ($n = |V(G)|$) which represents some embedding of G in the plane. Each location $A[i]$ contains a *pointer* to a *doubly linked circular list* containing $N_G(v_i)$. For simplicity, we associate the list pointed at by the variable $A[i]$ with the name $A[i]$. More importantly, the vertices of $N_G(v_i)$ appear in the list $A[i]$ in the same order in which they appear around v according to a particular embedding of G.

That is, for a neighbour v_j of v_i in the list $A[i]$, the *"next"* pointer points to a vertex v_{j+1} such that the edge (v_i, v_{j+1}) appears after (v_i, v_j) in a counter-clockwise scan of the edges incident to v_i according to that particular embedding of G. Conversely, the *"previous"* pointer is associated with the clockwise direction (see for example Figure 2). This representation can be constructed in linear time by the Hopcroft-Tarjan planarity algorithm [HT74].

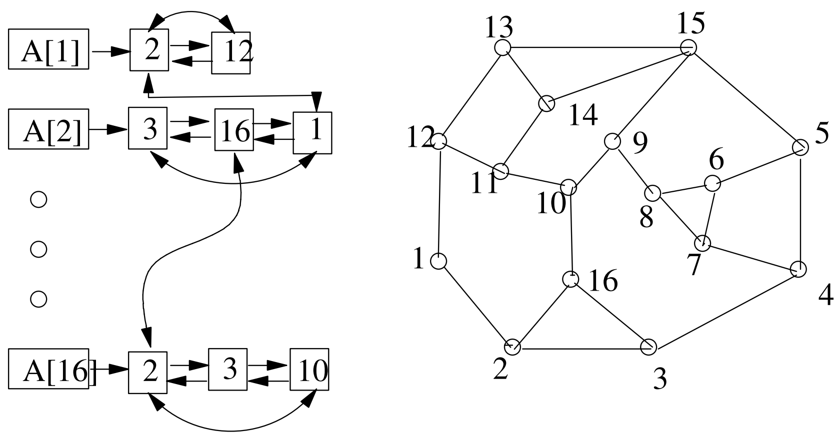

Figure 2. A planar graph and one of its planar adjacency list representations.

Furthermore, we assume that the two occurrences of an edge, say (v_0, v_1), in the two lists $A[0]$ and $A[1]$ are linked to each other. Using this link we can jump from one list to another in constant time. An edge (v_0, v_1) is said to be *traversed in the direction* $v_0 \rightarrow v_1$ if the traversal is accomplished by visiting the cell containing v_1 in the list of $A[0]$ in the data structure.

Now, we are ready to describe a procedure, called *IFB*, which identifies an edge-boundary of a face f containing an edge (v_0, v_1) in $O(|E(f)|)$ time.

Procedure IFB $(\mathbf{v_0}, \mathbf{v_1})$:

Traverse the edge (v_0, v_1) in the direction $v_0 \rightarrow v_1$. Jump to the list $A[1]$ and let v_2 be the next *vertex to* v_0 *on the list* $A[1]$. *Traverse the edge* (v_1, v_2) *in the direction* $v_1 \rightarrow v_2$. *Repeat the above steps until finally the edge* (v_0, v_1) *is traversed again in the direction* $v_0 \rightarrow v_1$.

End of IFB.

Using the above procedure we next show how to classify $V(G)$ into levels $V_0, V_1, \cdots, V_l$ in linear time, taking any arbitrary face of G as being the outer face.

Lemma 4. *Let G be a planar connected graph on n vertices. Then $V(G)$ can be partitioned into levels as described above in $O(n)$ time.*

Proof. Use the Hopcroft-Tarjan [HT74] algorithm to compute a planar adjacency list of G. In addition, use the array $Level[1 \cdots n]$ to record the level numbers of the vertices identified so far. Pick an arbitrary edge (v_0, v_1) of G and use the *IFB* procedure to identify the set of vertices V_0 defining the boundary of a face f containing that edge. Now, f is taken as the outer face of G.

Suppose that the subset V_i, $0 \le i < l$, has been computed so far. We now show how to identify V_{i+1}. First, observe that $V_0 \cup \cdots \cup V_i$ can be deleted from the planar adjacency list $A[1 \cdots n]$ to yield another planar adjacency list $A_{i+1}[1 \cdots n]$ for the subgraph $G_{i+1} = G \backslash V_0 \cup \cdots \cup V_i$. Hence, the *IFB* procedure can be applied to $A_{i+1}[1 \cdots n]$.

Second, observe that any vertex x, $x \in V_{i+1}$, can be reached from a vertex x' (possibly $x' = x$), $x' \in N_G(V_i) \cap V_{i+1}$, by a path consisting of vertices of V_{i+1}. The above two observations together with the correctness of procedure *IFB* are the basic ingredients to establish the correctness of the the following steps to compute V_{i+1}:

1) Set all vertices in V_i to be *unvisited*.

2) Repeat until every vertex in V_i has been *visited*.

   2.1) Let v_l be the next *unvisited* vertex in V_i.

   2.2) Jump to $A_i[l]$. Find the *next* vertex v_{l+1} to v_l on the list $A_i[l]$ whose level number has not been determined yet. By the definition of V_{i+1} we know that $v_{l+1} \in V_{i+1}$. Assign level $i+1$ to v_{l+1}. If no such a vertex exists go to statement 2.5.

   2.3) Otherwise, jump to $A_i[l+1]$. Find a vertex v_{l+2} *next* to v_l on that list (thus, v_l, v_{l+1} and v_{l+2} share a face of G) whose level number has not been determined yet. If no such a vertex exists go to statement 2.5.

   2.4) Otherwise, by the properties of $A_i[1 \cdots n]$, $v_{l+2} \in V_{i+1}$. Call $IFB(v_{l+1}, v_{l+2})$ and assign level $i+1$ to all vertices visited by the *IFB* procedure.

   2.5) Flag the vertex v_l as being *visited.*

   2.6) End.

One may verify that the time required to identify V_{i+1} is proportional to $\sum_{x \in V_i} deg(x) + |V_{i+1}|$. Hence, computing the level numbers of $V(G)$ requires time proportional to $2|E(G)| + n$. That is, the overall computation can be done in $O(n)$ time. □

Having computed the level numbers of $V(G)$ in the array $Level[1 \cdots n]$, one can assign each edge of G to E_1 or E_2 in constant time. Hence,

Lemma 5. *Let G be an n vertex planar graph. Then a partitioning of $E(G)$ into two partial 3-trees can be computed in $O(n)$ time.*

5. Planar Graphs not Coverable by partial 1- and 2-Trees

In this section we show an infinite family of triangulated 3-connected planar graphs each of which is not a partial (1,2)-tree. The construction is carried recursively using the following operation. Suppose G and H are two triangulated planar graphs embedded in the plane. Denote by $\mathbf{I}(G \leftarrow H)$ the planar graph obtained by *inserting* a copy of H into every face of G. The *insertion* is done by identifying the three vertices lying on the exterior face of H with the three vertices lying on the face of G in which H is to be inserted. For example, if G is triangulated planar graph embedded in the plane then $I(G \leftarrow K_4)$ is obtained by adding a new vertex x_f to each face f of G and adjoining it to each of the three vertices in $V(f)$. Now we are ready to prove the following theorem:

Theorem 2. *There exists an infinite family of 3-connected triangulated planar graphs that are not partial (1,2)-trees.*

Proof. Let G_1 be a 5-connected triangulated planar graph. Thus, $|V(G_1)| \geq 12$. Fix an embedding of G_1 in the plane. Let $G_2 = \mathbf{I}(G_1 \leftarrow K_4)$, $G_3 = \mathbf{I}(G_2 \leftarrow K_4)$, $G_4 = \mathbf{I}(G_3 \leftarrow P_6)$, where P_6 is the platonic graph on six vertices, illustrated in Figure 3 (b). In addition, if A is a triangle of G_1 (a face of G_1) then denote the subgraph of G_i, $1 \leq i \leq 4$, embedded inside A and including the edges of A by A_i. That is, $A_1 \cong A$, $A_2 \cong \mathbf{I}(A_1 \leftarrow K_4)$, $A_3 \cong \mathbf{I}(A_2 \leftarrow K_4)$ and $A_4 \cong \mathbf{I}(A_3 \leftarrow P_6)$.

Our objective is to show that G_4 is not a partial (1,2)-tree. An infinite family of such graphs can then be constructed from the infinite set of 5-connected planar triangulated graphs. To derive a contradiction, suppose that G_4 can be covered by a partial 2-tree S_4 and a partial 1-tree T_4. Consequently, any graph G_i, $1 \leq i \leq 3$, is covered by the partial 2-tree $S_i = (V(G_i), E(G_i) \cap E(S_4))$ and the partial 1-tree $T_i = (V(G_i), E(G_i) \cap E(T_4))$. Let us denote the restriction of S_i, $1 \leq i \leq 4$, to the subgraph A_i contained inside a triangle A of G_1 by $S_i|A_i$. That is, $S_i|A_i = (V(A_i), E(A_i) \cap E(S_i))$.

Now, each $G_i, 1 \leq i \leq 4$, is a triangulated planar graph (hence, $|E(G_i)| = 3|V(G_i)| - 6$) and consequently $2|V(G_i)| - 5 \leq |E(S_i)| \leq 2|V(G_i)| - 3$ and $|V(G_i)| - 3 \leq |E(T_i)| \leq |V(G_i)| - 1$. Hence, we may need to add at most two edges to complete $S_i, 1 \leq i \leq 4$, to a 2-tree. Denote such a possible set of edges by $M(S_i)$. Note that $M(S_i)$ is not necessarily unique. We now prove the following claims in order.

1) S_1 contains at least 5 triangular faces of G_1.

   For convenience, we call a triangle A of G_1 *defective in* G_i, $i = 2$ or 3, if the restriction $S_i|A_i$ is a partial 2-tree but not a 2-tree. In contrast, it is *good* if it is not defective in G_2 and G_3.

2) G_1 has at least one good triangle.

3) If A is a good triangle then either the restriction $S_4|A_4$ contains a homeomorph from K_4 or $T_4|A_4$ contains a cycle.

The theorem then follows since claim (3) implies that either S_4 is not a partial 2-tree or T_4 is not a forest, contradicting the assumption.

Proof of claim (1). Consider a possible set $M(S_1)$. If $|M(S_1)| = 0$ then S_1 is a 2-tree having $|V(G_1)| - 2$ triangles (≥ 10) and the claim holds. Otherwise, $1 \leq |M(S_1)| \leq 2$. Note that $M(S_1)$ may contain edges not in $E(G_1)$. Let $|V(G_1)| - 2 - k$, $k \geq 1$, denote the number of triangles in S_1. We show that $k \leq 5$. If $|M(S_1)| = 1$, say $M(S_1) = \{e\}$, then e can not be shared between three or more triangles in $S_1 + e$. Otherwise, S_1 would contain a subgraph isomorphic to $K_{2,3}$ and consequently, G_1 contains such a subgraph, contradicting the connectivity of G_1. Thus, $|M(S_1)| = 1$ implies $k \leq 2$.

The remaining case when $|M(S_1)| = 2$, say $M(S_1) = \{e_1, e_2\}$ is treated similarly. Here, there may be at most 2 triangles in the 2-tree $S_1 + e_1 + e_2$ containing any single edge e_1 or e_2. Moreover, there may be at most one triangle in $S_1 + e_1 + e_2$ containing both e_1 and e_2. Hence, $k \leq 5$ and the claim follows.

Proof of claim (2). By claim (1), G_1 has a set $\mathbf{A} = \{A^{(l)} | l = 1, 2, \cdots, 5\}$ of triangles. As mentioned above, $|M(S_i)| \leq 2$ for every $i, 1 \leq i \leq 4$. Now, for each defective triangle $A_i^{(l)}$, $i > 1$, $A^{(l)} \in \mathbf{A}$, one can add an edge (x, y), $x, y \in V(A_i^{(l)})$, to the set of edges $E(S_i)$ such that the resulting graph is a partial 2-tree. Hence, at most 2 distinct triangles in $\mathbf{A}$ are defective in G_i, $i = 2$ *or* 3. Thus, at most 4 triangles in $\mathbf{A}$ are not *good* and one is *good* in G_1, G_2 and G_3.

Proof of claim (3). Suppose A is a good triangle. One may verify that at least one of the 9 internal triangular faces of A_3, say (a, b, c), has the following two properties (refer to Figure 3 (a)):

(1) one edge of A, say (a, b), belongs to S_3. The other two edges belong to T_3.

(2) if P_{ac} (or P_{bc}) is an (a, c)-path (respectively, (b, c)-path) through the vertices of the copy of P_6 inserted in the triangle (a, b, c) (hence, it is vertex-disjoint from $V(A_3) \backslash \{a, b, c\}$) then the subgraph $S_3 + P_{ac}$ (respectively, $S_3 + P_{bc}$) contains a subgraph homeomorphic from K_4.

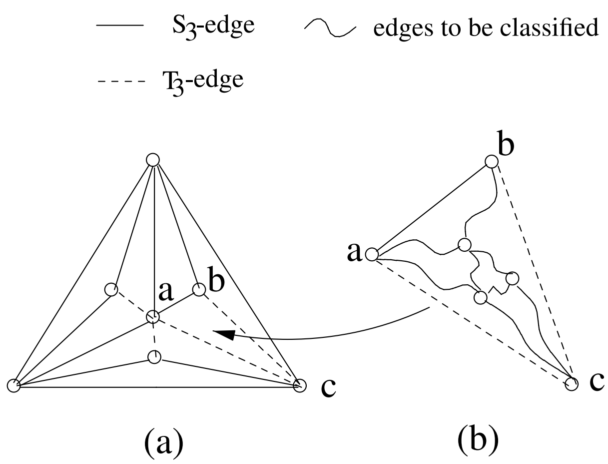

Figure 3. (a) A good triangle A. (b) A copy of P_6 in (a, b, c).

Now, let H be the copy of P_6 that has been inserted in the face (a, b, c) of G_3 to obtain G_4, as illustrated in Figure 3 (b). Here, $(a, b) \in S_4$ (the thick edge) and (b, c), $(a, c) \in T_4$ (the light edges). By exhausting a few cases, one may verify that any further partitioning of the remaining edges of H (the splines) into a partial 2-tree and a partial 1-tree results in a partial 2-tree which contains an (a, c)-path or a (b, c)-path or a partial 1-tree that forms a cycle with the other edges of T_3. In any case, a contradiction arises. This completes the proof of the theorem. □

6. Concluding Remarks

In this paper we showed that planar graphs are all partial (3,3)-trees but not necessarily partial (1,2)-trees. A linear time algorithm to partition the edges of any given planar graph to two partial 3-trees is presented. The algorithm uses the planarity testing algorithm of [HT74]. A next interesting step in this direction is to determine whether there exists an efficient algorithm to compute a partial (1,2)-tree cover of the edges of a given planar graph whenever possible.

On the other hand, deciding whether planar graphs are partial (2,2)-trees or (2,3)-trees seem to be more challenging problems. Moreover, resolving any such problem seems to be a reasonable step to solve the outerthickness conjecture of planar graphs suggested by [CGH71] as mentioned in the introduction.

Acknowledgements

The first author would like to thank W. R. Pulleyblank for many interesting discussions throughout this work. Research of the second author is supported by NSERC Canada grant A0579.

References

[AP84a] Arnborg S. and Proskurowski A., ‘‘Characterization and Recognition of Partial K-trees’’, *Proceedings of the 16th Southeastern Conference on Combinatorics, Graph Theory and Computing, Congressus Numerantium, Vol. 47, 1985, pp. 69-75.*

[AP84b] Arnborg S. and Proskurowski A., ‘‘Linear Time Algorithms for NP-Hard Problems on Graphs Embedded in k-trees’’, *TRITA-NA-8404, Royal Institute of Technology, S-100 44 Stockholm, Sweden, 1984.*

[APC86] Arnborg S., Proskurowski A. and Corneil D., ‘‘Forbidden Minors Characterization of Partial 3-trees’’. *Technical Report CIS-TR-86-07, Dept. of Computer and Information Science, University of Oregon.*

[BP71] Beineke L. W. and Pippert R. E., ‘‘Properties and Characterizations of k-Trees’’, *Mathematika, Vol. 18, 1971, pp. 141-151.*

[CGH71] Chartrand G., Geller D. P. and Hedetniemi S., ‘‘Graphs with Forbidden Subgraphs’’, *J. Comb. Theory, Vol. 10, 1971, pp. 12-41.*

[Ed65] Edmonds J., ‘‘Minimum Partition of a Matroid into Independent Subsets’’, *J. Res. Nat. Bur. Stand., Vol. 69B, 1965, pp. 67-72.*

[EC85] Elmallah E. S. and Colbourn C. J., ‘‘Reliability of Δ-Y Networks’’, *Proceedings of the 16th Southeastern Conference on Combinatorics, Graph Theory and Computing, Congressus Numerantium, Vol. 48, 1985, pp. 49-54.*

[EC86] Elmallah E. S. and Colbourn C. J., ‘‘Partial k-Tree Algorithms’’, *to appear in the Proceedings of the 250th Anniversary Conference on Graph Theory, March 1986, Indiana Purdue Fort Wayne University.*

[EC88] Elmallah E. S. and Colbourn C. J., ‘‘On Two Dual Classes of Planar Graphs’’, *To appear in Discrete Mathematics.*

[GJ79] Garey M. R. and Johnson D. S., *Computers and Intractability: A Guide to the Theory of NP-completeness,* W. H. Freeman, San Francisco, 1979.

[GRE84] Gilbert J. R., Rose D. J. and Edenbrandt A., "A Separator Theorem for Chordal Graphs", *SIAM J. Alg. Disc. Meth., Vol. 5, No. 3, 1984, pp. 306-313.*

[Go80] Golumbic M. C., *Algorithmic Graph Theory and Perfect Graphs,* Academic press, Inc., New York, 1980.

[HT74] Hopcroft J. E. and Tarjan R., ‘‘Efficient Planarity Testing’’, *J. Assoc. Comp. Mach., Vol. 21, 1974, pp. 549-568.*

[Jo85] Johnson D., ‘‘The NP-Completeness Column: an Ongoing Guide’’, *Journal of Algorithms, Vol. 6, 1985, pp. 434-451.*

[Na61] Nash-Williams C. St. J., ‘‘Edge-Disjoint Spanning Trees of Finite Graphs’’, *J. London Math. Soc., Vol. 36, 1961, pp. 445-450.*

[RS86] Robertson N. and Seymour P. D., ‘‘Graph Minors II. Algorithmic Aspects of Tree-Width’’, *Journal of Algorithms, Vol. 7, 1986, pp. 309-322.*

[Tu56] Tutte W. T., ‘‘A Theorem on Planar Graphs’’, *Trans. Amer. Math. Soc., Vol. 82, 1956, pp. 99-116.*

[Tu61] Tutte W. T., ‘‘On the Problem of Decomposing a Graph into n Connected Factors’’, *J. London Math. Soc., Vol. 36, 1961, pp. 221-230.*

[WC83] Wald J. A. and Colbourn C. J., ‘‘Steiner Trees, Partial 2-Trees and Minimum IFI Networks’’, *Networks, Vol. 13, 1983, pp. 159-167.*