PLANARITY IN LINEAR TIME

Class notes

Robin Thomas*
School of Mathematics
Georgia Institute of Technology
Atlanta, Georgia 30332, USA
thomas@math.gatech.edu

ABSTRACT

I give a self-contained exposition of a linear-time planarity algorithm of Shih and Hsu.

20 December 1995
Revised 6 June 1997

---

* Partially supported by NSF under Grant No. DMS-9303761 and by ONR under Grant No. N00014-93-1-0325.

1. INTRODUCTION

There are four linear-time planarity algorithms that I am aware of [1, 3, 4, 5, 6], of which the one in [5] (thereafter referred to as the S&H algorithm) seems to be the simplest. The purpose of this note is to give a self-contained presentation of this algorithm. In section 2 we prove a structural result that is the basis for the correctness of the S&H algorithm, and illustrate it by deducing Kuratowski's theorem from it. The structural result immediately gives a quadratic planarity algorithm. In Section 3 we show how to implement it to run in linear time and linear space for 3–connected graphs on a RAM machine.

2. PLANAR GRAPHS

By a *graph* we mean a finite, undirected, simple graph. This is without loss of generality, because a multigraph is planar if and only if its underlying simple graph is planar. A graph is *planar* if it can be drawn in the plane without crossings. The S&H algorithm proceeds as follows. First it suitably enumerates the vertices of an input graph G as $w_1, w_2, \ldots, w_n$, and then iterates n times the following step. For $i = 1, 2, \ldots, n$, let H_i be the subgraph of G induced by $\{w_1, w_2, \ldots, w_i\}$. Given a "frame" of H_i (defined later) it tries to extend it to a frame of H_{i+1}, and failing that it produces a Kuratowski subgraph of G as a certificate of non-planarity. Each iteration takes time proportional to the length of a certain path that is constructed during this process; moreover, each edge of G is in at most two such paths, which implies linearity of the algorithm.

Let us now be more precise. The graph with no vertices is called the *null graph* and is regarded as being connected. *Paths* and *circuits* have no "repeated" vertices. Let G be a graph. If X is a vertex or a set of vertices, we denote by $G\backslash X$ the graph obtained from G by deleting X. If G_1 and G_2 are subgraphs of G we denote by $G_1 \cup G_2$ the subgraph of G with vertex-set $V(G_1) \cup V(G_2)$, edge-set $E(G_1) \cup E(G_2)$ and the obvious incidences. The graph $G_1 \cap G_2$ is defined analogously. A vertex v of G is a *cut-vertex* of G if the edge-set of G can be partitioned into two non-empty sets E_1 and E_2 in such a way that v is the only vertex of G incident with edges in both E_1 and E_2. A graph G is a *block* if it has no cut-vertex. A subgraph H of a graph G is a *block of G* if H is a block and is a maximal subgraph of G with this property. We denote the set of blocks of a graph G by $\mathcal{B}(G)$. Let S be a subgraph of a graph G, let $X \subseteq V(G)$, and let v be a vertex of S. We say that v is *X-exposed* for S in G if there exists a path in G with one end v, the other end in X, and disjoint from $V(S) - \{v\}$. Thus every vertex of $V(S) \cap X$ is X-exposed for S in G.

Let G be a graph, and let H be an induced subgraph of G. We say that $\mathcal{C} = (C_B : B \in \mathcal{B}(H))$ is a *frame for H in G* if for every $B \in \mathcal{B}(H)$

(i) either B has at most one edge, in which case $B = C_B$, or C_B is a circuit of B,

(ii) B has a planar drawing in which C_B bounds the infinite region, and

(iii) every vertex of B that is $(V(G) - V(H))$-exposed for B in G belongs to $V(C_B)$.

(In fact, (ii) implies (i), but (i) is included for clarity.) If H is a block then $\mathcal{B}(H) = \{H\}$ and $\mathcal{C} = (C_H)$; otherwise we define $C_H = \bigcup\{C_B : B \in \mathcal{B}(H)\}$. To motivate the next steps let us make the following observation.

(2.1) *Let H be an induced subgraph of a planar graph G and let $G \backslash V(H)$ be connected or null. Then H has a frame in G.*

Proof. Exercise. $\square$

Let H be an induced subgraph of G, and let $\mathcal{C} = (C_B : B \in \mathcal{B}(H))$ be a frame for H in G. A (possibly null) path P is called a *perimeter path for H and $\mathcal{C}$* if P is a subgraph of C_H. If $E(P) = \emptyset$ we define the *span of P* and the *complementary path to P* to be itself. Otherwise, let $B_1, B_2, \ldots, B_k \in \mathcal{B}(H)$ be all the blocks of H that contain an edge of P. We call $B_1 \cup B_2 \cup \cdots \cup B_k$ the *span* of P in H. For $i = 1, 2, \ldots, k$, $P \cap C_{B_i}$ is a path; if B_i has no more than one edge let $P'_i = P_i$, and otherwise let P'_i be the subpath of C_{B_i} with the same ends as P_i and edge-disjoint from it. We say that $P'_1 \cup P'_2 \cup \cdots \cup P'_k$ is the *complementary path* to P. Let $X, Y \subseteq V(H)$ be sets. A perimeter path P is called an *(X,Y)-path for H and $\mathcal{C}$* if

(i) both ends of P belong to X,

(ii) every vertex of the span S of P in H that is X-exposed for S in H belongs to P,

(iii) every vertex of S that is Y-exposed for S in H belongs to the complementary path to P, and

(iv) no component of $H\backslash V(S)$ contains a neighbor of a vertex of S, a member of X and a member of Y.

A subgraph L of H is called an *(X, Y)-link for H and $\mathcal{C}$* if the intersection of L with every component of H is an (X, Y)-path for H and $\mathcal{C}$. The concept of an (X, Y)-link is a tool for extending frames, as the next lemma explains.

(2.2) *Let H' be an induced subgraph of a graph G, let w be a vertex of H', let $H = H'\backslash w$, and let $\mathcal{C}$ be a frame for H in G. Let X be the set of vertices of H that are adjacent to w, and let Y be the set of vertices of H that are adjacent to a vertex in $V(G) - V(H')$. If there is an (X, Y)-link for H and $\mathcal{C}$, then H' has a frame in G.*

Proof. Let $\mathcal{C} = (C_B : B \in \mathcal{B}(H))$, and let B be a block of H'. If B is a block of H, then we put $C'_B = C_B$; otherwise $w \in V(B)$ and we proceed as follows. If $V(B) = \{w\}$ we put $C'_B = B$; otherwise there exists a component Q of H with $V(Q) \cap X \neq \emptyset$ such that B is the union of w, $V(Q) \cap X$, all edges between w and vertices in $V(Q) \cap X$ and all blocks D of Q containing at least two distinct vertices that are X-exposed for D in H. Let P be the intersection of the (X, Y)-link for H and $\mathcal{C}$ with Q; then P is an (X, Y)-path in Q. Let C'_B be obtained from the complementary path to P by adding two edges, each joining one end of P to w. It is now straightforward to verify that $\mathcal{C}' = (C'_B : B \in \mathcal{B}(H'))$ is a frame for H' in G. □

Next we introduce obstructions to the existence of frames. Let H be an induced subgraph of a graph G, let $\mathcal{C}$ be a frame for H in G and let $X, Y \subseteq V(H)$ be sets. An *(X, Y)-obstruction for H and $\mathcal{C}$* is either

(i) a quintuple (B, v_1, v_2, v_3, v_4) such that $B \in \mathcal{B}(H)$ and v_1, v_2, v_3, v_4 are distinct vertices of C_B occurring on C_B in the order listed and such that v_1 and v_3 are X-exposed for B in H, and v_2 and v_4 are Y-exposed for B in G, or

(ii) a quadruple (B, v_1, v_2, v_3) such that $B \in \mathcal{B}(H)$ and v_1, v_2, v_3 are distinct vertices of C_B such that all of them are both X- and Y-exposed for B in H, or

(iii) a quadruple (v, H_1, H_2, H_3) such that $v \in V(H)$ and H_1, H_2, H_3 are three distinct components of $H\backslash v$, each containing a member of X, a member of Y, and a neighbor of v.

A *Kuratowski graph* is a graph isomorphic to a subdivision of K_5 or $K_{3,3}$.

(2.3) *Let $G, w, H, H', \mathcal{C}, X$ and Y be as in (2.2), and assume that $G \backslash V(H)$ and $G \backslash V(H')$ are connected. If there exists an (X, Y)-obstruction for H and $\mathcal{C}$, then G contains a Kuratowski subgraph.*

Proof. The proof is straightforward and is left to the reader. □

The following lemma justifies the term obstruction.

(2.4) *Let H be an induced subgraph of a graph G, let $\mathcal{C}$ be a frame for H in G, and let X, Y be subsets of $V(H)$ such that every vertex in $X \cup Y$ is adjacent to a vertex in $V(G) - V(H)$. Then either there exists an (X, Y)-link or an (X, Y)-obstruction for H and $\mathcal{C}$.*

Proof. Let us assume that there is no (X, Y)-obstruction for H and $\mathcal{C}$. We must show that there exists an (X, Y)-link for H and $\mathcal{C}$. To this end we may assume that H is connected, for otherwise we apply the same argument to every component of H. For every subgraph B of H let X_B be the set of vertices of B that are X-exposed for B in H, and let Y_B be defined analogously. We first claim the following.

(1) *For every block B of H there exists an (X_B, Y_B)-path for B and (C_B).*

Indeed, $X_B \cup Y_B \subseteq V(C_B)$ by the third frame axiom. If $|X_B| \leq 1$ then the path with vertex-set X_B satisfies (1). We may therefore assume that $|X_B| \geq 2$. Since there is no (X, Y)-obstruction for H and $\mathcal{C}$ we deduce that there is no (X_B, Y_B)-obstruction for B and (C_B). It follows that C_B has a path P with both ends in X_B such that $X_B \subseteq V(P)$, and such that every vertex of $V(P) \cap Y_B$ is an end of P. Then P is an (X_B, Y_B)-path for B and (C_B). This proves (1).

To prove the lemma we proceed by induction on $|V(H)|$. If H is a block then $X_H = X$, $Y_H = Y$ and the theorem follows from (1). We may therefore assume that H has a cut vertex u. Let $Z_1, Z_2, \ldots, Z_k$, where $k \geq 2$, be the vertex-sets of the components of $H \backslash u$, and for $i = 1, 2, \ldots, k$ let H_i be the subgraph of H induced by $Z_i \cup \{u\}$. If more than two of the sets $Z_1, Z_2, \ldots, Z_k$, say Z_1, Z_2 and Z_3 contain vertices of both X and Y, then $(u, H_1 \backslash u, H_2 \backslash u, H_3 \backslash u)$ is an (X, Y)-obstruction for H and $\mathcal{C}$, a contradiction. Thus we may assume that for $i = 3, 4, \ldots, k$, Z_i is disjoint from one of X or Y.

By the induction hypothesis there exists, for $i = 1, 2$, either an (X_{H_i}, Y_{H_i})-link or an (X_{H_i}, Y_{H_i})-obstruction for H_i and $\mathcal{C}_i$, where $\mathcal{C}_i$ is the restriction of $\mathcal{C}$ to H_i. Since an (X_{H_i}, Y_{H_i})-obstruction for H_i and $\mathcal{C}_i$ is easily converted to an (X, Y)-obstruction for H and $\mathcal{C}$, we may assume that, for $i = 1, 2$, there is an (X_{H_i}, Y_{H_i})-link P_i for H_i and $\mathcal{C}_i$, which, in fact, is an (X_{H_i}, Y_{H_i})-path for H_i and $\mathcal{C}_i$, because H_i is connected. Notice that $X_{H_i} \subseteq X \cup \{u\}$ for all $i = 1, 2, \ldots, k$. If for some $i \in \{1, 2\}$, u is not an end of P_i, then P_i is an (X, Y)-path for H and $\mathcal{C}$. We may therefore assume that u is an end of both P_1 and P_2. Then $P_1 \cup P_2$ is an (X, Y)-path for H and $\mathcal{C}$, as desired. □

(2.5) Kuratowski's theorem. *If a graph G is not planar, then it contains a Kuratowski subgraph.*

Proof. Since G is not planar, it has a component that is not planar, and so we may assume that G is connected. Let H be a maximal induced subgraph of G that has a frame $\mathcal{C}$ and such that $G \backslash V(H)$ is connected. Since G is not planar, $H \neq G$. Let $w \in V(G) - V(H)$ be such that $G \backslash (V(H) \cup \{w\})$ is connected, let H' be the subgraph of G induced by $V(H) \cup \{w\}$, and let X, Y be as in (2.2). By (2.4) there is either a (X, Y)-link or an (X, Y)-obstruction for H and $\mathcal{C}$. By (2.2) and the maximality of H there is no (X, Y)-link for H and $\mathcal{C}$, and hence there is an (X, Y)-obstruction for H and $\mathcal{C}$. By (2.3) G has a Kuratowski subgraph. □

The proof of (2.5) can be easily converted to a quadratic algorithm to decide if a graph is planar. In the next section we will explain how to implement it more efficiently for 3-connected graphs. To that end we need a refinement of (2.4). First we prove an easy lemma.

(2.6) *Let G, w, H, $\mathcal{C}$, X and Y be as in (2.2), and let us assume that G is 3-connected. Let v be a cut-vertex of H, let B be a block of H containing v, and let K be a component of $H\backslash v$ disjoint from B and containing a neighbor of v. Then $V(K)$ contains a member of Y, and hence v is Y–exposed for B in H.*

Proof. Since $G\backslash\{v,w\}$ is connected, we deduce that $V(K)\cap Y\neq\emptyset$. It follows that v is Y–exposed for B in H. □

If B is a subgraph of a graph H and $Z\subseteq V(H)$, we say that B is *Z-active* in H if B contains at least two vertices that are Z-exposed for B in H. The following is a refinement of (2.4) for 3–connected graphs.

(2.7) *Let G, w, H, $\mathcal{C}$, X and Y be as in (2.2), assume that G is 3-connected, and that for every block B of H there exists a path P_B as in claim* (1) *of the proof of (2.4). Let L be the union of all P_B, the union taken over all blocks B of H that are X–active in H. If every component of L is a path then L is an (X,Y)–link for H and $\mathcal{C}$; otherwise there exists an (X,Y)-obstruction for H and $\mathcal{C}$.*

Proof. Let X_B, Y_B be as in the proof of (2.4). We fist prove the following claim.

(1) *For every component K of H, the graph $K\cap L$ is connected or null.*

Indeed, otherwise there exist an integer $k\geq 1$ and distinct blocks $B_0, B_1, \ldots, B_k$ of K, and distinct vertices $v_1, v_2, \ldots, v_k$ of K such that for $i=1,2,\ldots,k$, $v_i\in V(B_{i-1})\cap V(B_i)$, both B_0 and B_k are X–active in H and P_{B_0} and P_{B_k} are subpaths of different components of $K\cap L$. We may assume that the above blocks and vertices are chosen so that k is minimum. For $i=1,2,\ldots,k-1$, v_{i-1} and v_i are X–exposed for B_i in H; hence B_i is X–active in H, and thus $E(B_i)\cap E(L)\neq\emptyset$. On the other hand the minimality of k implies $E(B_i)\cap E(L)=\emptyset$, and so $k=1$. Since v_1 is X–exposed for both B_0 and B_1 in H, we deduce that $v_1\in V(P_{B_0})\cap V(P_{B_1})$, contrary to the fact that P_{B_0} and P_{B_k} are subpaths of different components of $K\cap L$. This proves (1).

(2) *If for some component K of H, $K\cap L$ is not a path, then there exists an (X,Y)–obstruction for H and $\mathcal{C}$.*

Indeed, if $K \cap L$ is not a path, then by (1) and the fact that L is clearly a forest, L has a vertex u of degree at least three. Suppose first that P_{B_1}, P_{B_2} and P_{B_3} all have end u, where B_1, B_2, B_3 are three distinct X-active blocks of H. For $i = 1, 2, 3$ let J_i be the component of $H \backslash u$ containing $B_i \backslash u$. Using (2.6) we deduce that (u, J_1, J_2, J_3) is an obstruction for H and $\mathcal{C}$, as desired. Thus we may assume that u is an interior vertex of P_{B_1}, and belongs to P_{B_2}, where B_1 and B_2 are distinct X-active blocks of H. By (2.6) $u \in Y_{B_1}$, contrary to the fact that P_{B_1} is an (X_{B_1}, Y_{B_1})-path for B_1 and (C_{B_1}). This proves (2).

(3) *Every vertex of L of degree one belongs to X.*

Indeed, let u be a vertex of L of degree one, and suppose that it does not belong to X. Let B be an X–active block of H such that u is an end of P_B. Since P_B is an (X_B, Y_B)-path for B and (C_B) we deduce that $u \in X_B$, and hence there exists a block B' with $V(B') \cap V(B) = \{u\}$ and such that $X_{B'} - \{u\} \neq \emptyset$. Thus B' is X-active, but $E(B') \cap E(P) = \emptyset$, a contradiction. This proves (3).

The result now follows easily from (2), (3) and the definition of L. □

3. A LINEAR-TIME ALGORITHM FOR 3-CONNECTED GRAPHS

Our objective in this section is to give a description of the following algorithm.

(3.1) *A planarity algorithm for 3-connected graphs.*

Input. *A 3-connected graph G.*

Output. *A valid statement that G is planar, or a Kuratowski subgraph of G (in which case G is non-planar).*

Running time. $O(|V(G)|)$.

We begin with the following "on-line" algorithm. Let C be a circuit, and let X be a set of vertices of C. By an *X-span* in C we mean a subpath P of C with both ends in X and with $X \subseteq V(P)$. We denote by $\mathrm{sp}_C(X)$ the length of a shortest X-span in C.

(3.2) *There exists an "on-line" algorithm with the following specifications.*

Input. *A circuit C, and a sequence $v_1, v_2, \ldots, v_k$ of vertices of C.*

Output. *For all $i = 1, 2, \ldots, k$, given $v_1, v_2, \ldots, v_i$, a $\{v_1, \ldots, v_i\}$-span P_i of length at most $3\mathrm{sp}_C(\{v_1, v_2, \ldots, v_i\})$.*

Running time of step i: $O(\mathrm{sp}_C(\{v_1, \ldots, v_i\}) - \mathrm{sp}_C(\{v_1, v_2, \ldots, v_{i-1}\}) + 1)$.

We leave the description of (3.2) to the reader. □

Let $w_1, w_2, \ldots, w_n$ be a post-order numbering of the vertices of G relative to a depth-first search spanning tree (T, w_n); that is, $w_1, w_2, \ldots, w_n$ is an ordering of the vertices of G and T is a spanning tree of G satisfying conditions (D1) and (D2) below. We say that a vertex v of G is a *descendant* of a vertex u of G if u is a vertex of the unique path in T with ends v and w_n. The two conditions we require are

(D1) for every edge of G, one of its ends is a descendant of the other, and

(D2) for $i = 1, 2, \ldots, n$, if w_i is a descendant of w_j, then $i < j$.

It is a well-known and easy fact that a spanning tree and a numbering of the vertices of G satisfying (D1) and (D2) can be found in linear time. For $i = 1, 2, \ldots, n$ let $H_i = G \backslash \{w_{i+1}, w_{i+2}, \ldots, w_n\}$; then $G \backslash V(H_i)$ is connected. For $i = 1, 2, \ldots, n-1$ let X_i be the set of all vertices of H_i that are adjacent to w_{i+1}, and let Y_i be the set of all vertices of H_i that are adjacent to a vertex in $\{w_{i+2}, w_{i+2}, \ldots, w_n\}$. We will refer to n, T, w_i, H_i, X_i, Y_i or its subset $(i = 1, 2, \ldots, n)$ as a *standard notation*. If B is a connected non-null subgraph of G we define the *root* of B to be the highest-numbered vertex of B. If B is an arbitrary subgraph of G, its *roots* are the roots of its components. We need the following easy lemma.

(3.3) *Let G be a graph, and let n, T, w_i, H_i, X_i, Y_i $(i = 1, 2, \ldots, n)$ be a standard notation. Then for all $i = 1, 2, \ldots, n$, the following statements hold.*

(i) *If K is a component of H_i with root r, then every vertex of K is a descendant of r.*

(ii) *If K is a component of H_i containing a vertex of X_i, then the root of K belongs to X_i.*

(iii) *Let v be a vertex of H_i other than a root, and let u be the unique neighbor of v in T such that v is a descendant of u. Then $u \in V(H_i)$. Let B be the block of H_i containing the edge uv. Then B is the only block of H_i containing v such that v is not its root.*

Proof. To prove (i) let K and r be as stated. If r is a descendant of a vertex $u \in V(G)$, then $u \notin V(H_i)$ by (D2) and the definition of r. It follows from (D1) that every vertex of K is a descendant of r, as desired.

To prove (ii) let r be the root of K and let $u \in V(K) \cap X_i$; that is, u is adjacent to w_{i+1}. Then u is a descendant of w_{i+1} by (D1), and u is a descendant of r by (i). It follows that one of r and w_{i+1} is a descendant of the other, and hence r is a descendant of w_{i+1}, because $r \in V(H_i)$. Let w_k be the next to last vertex on the unique path in T from w_{i+1} to r. Then $k \leq i+1$ by (D2). On the other hand $w_k \notin V(H_i)$ because r is the root of K, and hence $k \geq i+1$. Thus r is adjacent to $w_k = w_{i+1}$, and hence $r \in X_i$, as desired.

To prove (iii) we first notice that clearly $u \in V(H_i)$. Let B' be a block of H_i containing v, let r be the root of B' and assume that $v \neq r$. Let P be the unique path in T with ends v and r; then $u \in V(P)$. Moreover, P is a subgraph of H_i, and hence it is a subgraph of B', because $B' \cup P$ is a block. Thus $u \in V(B')$, and hence $B' = B$, as desired. □

Let v be a vertex of H_i that is not a root of H_i, and let B be as in (3.3)(iii). We call B the *base block of v in H_i*.

Let $\mathcal{C}_1$ be the unique frame for H_1 in G. The S&H algorithm finds, for every $i = 1, 2, \ldots, n-1$, either an (X_i, Y_i)-link for H_i and $\mathcal{C}_i$, or an (X_i, Y_i)-obstruction for H_i and $\mathcal{C}_i$. In the latter case it uses (2.3) to output a Kuratowski subgraph of G and stops. In the former case it uses (2.2) to construct a frame $\mathcal{C}_{i+1}$ for H_{i+1} in G, and proceeds to the next iteration. Thus all we need to do is to describe how to find either an (X_i, Y_i)-link for H_i and $\mathcal{C}_i$, or an (X_i, Y_i)-obstruction for H_i and $\mathcal{C}_i$. We need two lemmas to facilitate this.

(3.4) *Let G be a graph, let n, H_i, X_i $(i = 1, 2, \ldots, n)$ be a standard notation, let $i \in \{1, 2, \ldots, n\}$, let B be a block of H_i containing a member of X_i, and let $v \in V(B)$. Then v is X_i-exposed for B in H_i if and only if at least one of the following three conditions holds:*

(i) *$v \in X_i$, or*

(ii) *v is the root of B, or*

(iii) *v is the root of an X_i–active block of H_i.*

Proof. If (i) holds then v is clearly X_i–exposed for B in H_i. If (ii) holds then by (3.3)(ii) the path of T between v and the root of H_i shows that v is X_i–exposed for B in H_i. If (iii) holds then let B' be an X_i–active block with root v and consider a path P in B' between v and u, where $u \neq v$ is a vertex that is X_i–exposed for B' in H_i. Let P' be a path in H_i with one end u, the other end in X_i, and disjoint from $B' \backslash u$. Then $P \cup P'$ shows that v is X_i–exposed for B in H_i.

Conversely, suppose that v is X_i–exposed for B in H_i, and that (i) and (ii) do not hold. Let P be a path in H with one end v, the other end in X_i, and disjoint from $V(B) - \{v\}$. Let B' be the block of H_i containing the edge of P incident with v. Since v belongs to B and is not its root, we deduce from (3.3)(iii) that v is the root of B'. Moreover, both ends of the path $P \cap B'$ are X_i–exposed for B' in H_i, and hence B' is X_i–active, as desired. □

(3.5) *Let G be a 3–connected graph, let n, H_i, X_i $(i = 1, 2, \ldots, n)$ be a standard notation, let $i \in \{1, 2, \ldots, n\}$, let B be a block of H_i, and let $v \in V(B)$. Then v is Y_i-exposed for B in H_i if and only if at least one of the following three conditions holds:*

(i) *$v \in Y_i$, or*

(ii) *v is the root of a block of H_i, but not a root of H_i, or*

(iii) *v is the root of H_i, and it is also the root of at least two blocks of H_i.*

Proof. By (2.6) a vertex v of B is Y_i–exposed for B in H_i if and only if either $v \in Y_i$, or v is a cut-vertex of H_i. This implies (3.5). □

For the next algorithm we need to explain how frames will be stored. Let $\mathcal{C}$ be a frame for H_i in G, and let v be a vertex of $V(C_{H_i})$. If v is a root of H_i we store that information; otherwise we store pointers to its neighbors in C_B, where B is the base block of v in H_i, and information indicating which of the two neighbors (if any) is the root of B. Moreover, for every vertex v of H_i we store the number of blocks of H_i with root v.

(3.6) *Algorithm.*

Input. *A 3-connected graph G, an integer $i \in \{1, 2, \ldots, n-1\}$, a frame $\mathcal{C}$ of H_i, the set $X_i \subseteq V(C_{H_i})$ presented as a list, and a characteristic function of the set $Y_i \subseteq V(C_{H_i})$, where H_i, X_i, Y_i is a standard notation.*

Output. *Either an (X_i, Y_i)-link for H_i and $\mathcal{C}$, or an (X_i, Y_i)-obstruction for H_i and $\mathcal{C}$.*

Running time. *$O(|V(L)|)$ if the output is an (X_i, Y_i)-link L for H_i and $\mathcal{C}$, and $O(|V(G)|)$ otherwise.*

Description. For a subgraph B of H_i let X_B be the set of all vertices of B that are X_i–exposed for B in H_i, and let Y_B be defined similarly. We proceed in three steps.

Step 1. We first find, using algorithm (3.2), all the X_i-active blocks of H_i and for every X_i-active block B of H_i, an X_B-span Q_B. Indeed, we begin by putting all the vertices of X_i on a stack. Then we iteratively pop a vertex v from the stack, and if v is not the root of H_i we apply (3.2) to v and C_B, where B is the base block of v in H_i. Every time a new block is encountered we push its root on the stack. By (3.4) all X_i–active blocks of H_i will be eventually located. Moreover, the total running time of this step is $O(\Sigma|V(Q_B)|)$, the summation taken over all X_i–active blocks B of H_i.

Step 2. For every X_i-active block B of H_i we do the following. If every vertex of $V(Q_B) \cap Y_B$ is an end of Q_B, then Q_B is an (X_B, Y_B)-path for B and (C_B), and we put $P_B = Q_B$. This clearly takes time $O(|V(P_B)|)$, because by (3.5) membership in Y_B can be tested in constant time. Otherwise, similarly as in the proof of claim (1) of (2.4), we either find an (X_B, Y_B)-path P_B for B and (C_B), or an (X_B, Y_B)-obstruction for B and (C_B). This takes time $O(|V(C_B)|)$. In the former case $V(C_B) - V(Q_B) \subseteq V(P_B)$; hence

$$|V(C_B)| \leq |V(Q_B)| + |V(C_B) - V(Q_B)| \leq 3\mathrm{sp}_{C_B}(X_B) + |V(P_B)| \leq 4|V(P_B)|,$$

and so the running time is $O(|V(P_B)|)$. In the latter case the (X_B, Y_B)-obstruction for B and (C_B) is an (X_i, Y_i)-obstruction for H_i and $\mathcal{C}$, and so if we find it, we output it and stop. That can clearly be done in time $O(|V(G)|)$. We may therefore assume that the above process yields an (X_B, Y_B)-path P_B for B and (C_B) for every active block B of H_i. The total running time of this step is then $O(\Sigma|V(P_B)|)$, the summation taken over all X_i–active blocks B of H_i.

Step 3. Let L be the union of P_B, over all X_i–active blocks of B of H_i. If every component of L is a path, then L is an (X_i, Y_i)–link for H_i and $\mathcal{C}$ by (2.7); we output it and stop. This takes time $O(|V(L)|)$. Otherwise we proceed as in the proof of (2.7) and find an (X_i, Y_i)–obstruction for H_i and $\mathcal{C}$, which can clearly be done in time $O(|V(G)|)$. This completes the description of (3.6). □

We are now ready to complete the description of (3.1).

Description of (3.1). Let G have n vertices. If G has more than $3n-6$ edges, then G is non-planar. We therefore read at most $3n-5$ edges of G from the input and call the resulting graph G. Ignoring the remaining edges does not affect the planarity status of G.

We find a spanning tree T of G, and a numbering $w_1, w_2, \dots, w_n$ of the vertices of G satisfying (D1) and (D2). Let H_1 be the subgraph of G with vertex-set $\{w_1\}$ and no edges, and let $\mathcal{C}_1$ be the unique frame for H_1 in G. We initialize the data structure described prior to (3.6), and iterate steps (1)–(4) for $i = 1, 2, \dots, n-1$.

(1) We apply (3.6) to find either an (X_i, Y_i)-link L_i for H_i and $\mathcal{C}_i$, or an (X_i, Y_i)-obstruction for H_i and $\mathcal{C}_i$. In the latter case we use (2.3) to find a Kuratowski subgraph of G, we output it and stop.

(2) We use L_i as in (2.2) to construct a frame $\mathcal{C}_{i+1}$ of $H_{i+1} = G \backslash \{w_{i+2}, w_{i+3}, \dots, w_n\}$. It is worth pointing out that a block of H_{i+1} is either a block of H_i, or has root w_{i+1}. This step can be implemented to run in time $O(|V(L_i)|)$. It involves, for every component P of L_i, the use of the complementary path. However, it is not necessary to scan the entire complementary path; it suffices to update $O(k)$ pointers, where k is the number of blocks comprising the span of P. Since $k = O(|V(P)|)$, the running time of this step is $O(|V(L_i)|)$.

(3) For every vertex v of H_{i+1} we update the number of blocks with root v. This only changes for w_{i+1} and the roots of X_i–active blocks of H_i, and can be done in time $O(|V(L_i)|)$.

(4) If $i = n-1$ then we output the statement that G is planar and stop. Otherwise we increase i by one and return to (1) for the next iteration.

Let $L_1, L_2, \dots, L_k$ be all the links resulting from applications of (3.6) in step (1). Thus either $k = n - 1$, or $k < n - 1$ and the application of (3.6) during iteration $k + 1$ yielded a Kuratowski graph. We need the following claim.

(∗) *Every edge of G belongs to at most two of the graphs $L_1, L_2, \ldots, L_k$.*

Indeed, let e be an edge of G. Let us say that e is *i–singular* if there exists a block of H_i with edge-set $\{e\}$. If $e \in E(L_i)$ and e is i-singular, then e is j–singular for no integer j with $i < j \leq n$. If $e \in E(L_i)$ and e is not i–singular, then for all integers j with $i < j \leq n$, $e \notin E(C_{H_j})$, and hence $e \notin E(L_j)$. The claim (∗) follows.

The bound on the running time of the algorithm now easily follows from (∗) and the running time bound of (3.6). □

4. CONCLUSION

There are two reasons why the algorithm (3.1) is not completely satisfactory. The first one is the requirement that G be 3–connected. One way to remove this restriction would be to apply the algorithm of [2] to decompose the input graph G into "3–connected components," and then apply (3.1) to each of these components. The graph G is planar if and only if each of its 3-connected components is planar. It is also possible to formulate the S&H algorithm so it runs in linear time for all input graphs regardless of their connectivity, but a formal exposition is technically more complicated.

The second unsatisfactory feature of (3.1) is that it does not return a planar embedding if the input graph is planar. That, too, can be remedied, but only at the expense of more technical complications.

The reader can perhaps work out these technical improvements. Hopefully we will include them in a future version of these notes.

Acknowledgment

Thanks to Ion Mandoiu and Christopher Carl Heckman for carefully reading an earlier version of the manuscript and for several helpful suggestions.

References

1. K. S. Booth and G. S. Luecker, Testing the consecutive ones property, interval graphs and graph planarity using PQ-tree algorithms, *J. Comput. Syst. Sci.* **13** (1976), 335–379.
2. J. E. Hopcroft and R. E. Tarjan, Dividing a graph into triconnected components, *SIAM J. Comput.* **2** (1973), 135–158.
3. J. E. Hopcroft and R. E. Tarjan, Efficient planarity testing, *J. Assoc. Comput. Mach.* **21** (1974), 549–568.
4. A. Lempel, S. Even and I. Cederbaum, An algorithm for planarity testing of graphs, In: Theory of Graphs, P. Rosenstiehl ed., Gordon and Breach, New York, 1967, 215–232.
5. W.-K. Shih and W.-L. Hsu, A simple test for planar graphs, manuscript.
6. S. G. Williamson, Depth-first search and Kuratowski subgraphs, *J. Assoc. Comput. Mach.* **31** (1984), 681–693.