A Linear-Time Algorithm to Find Four Independent Spanning Trees in Four-Connected Planar Graphs

Kazuyuki Miura, Daishiro Takahashi,
Shin-ichi Nakano, and Takao Nishizeki

Graduate School of Information Sciences
Tohoku University, Sendai 980-8579, Japan
e-mail:{miura,daishiro,nakano,nishi}@nishizeki.ecei.tohoku.ac.jp

Abstract. Given a graph G, a designated vertex r and a natural number k, we wish to find k "independent" spanning trees of G rooted at r, that is, k spanning trees such that, for any vertex v, the k paths connecting r and v in the k trees are internally disjoint in G. In this paper we give a linear-time algorithm to find four independent spanning trees in a 4-connected planar graph rooted at any vertex.

1 Introduction

Given a graph $G = (V, E)$, a designated vertex $r \in V$ and a natural number k, we wish to find k spanning trees $T_1, T_2, \cdots, T_k$ of G such that, for any vertex v, the k paths connecting r and v in $T_1, T_2, \cdots, T_k$ are internally disjoint in G, that is, any two of them have no common intermediate vertices. Such k trees are called *k independent spanning trees of G rooted at r*. Four independent spanning trees are drawn in Fig. 1 by thick lines. Independent spanning trees have applications to fault-tolerant protocols in networks [BI96,DHSS84,IR88,OIBI96].

Given a graph $G = (V, E)$ of n vertices and m edges, and a designated vertex $r \in V$, one can find two independent spanning trees of G rooted at any vertex in linear time if G is biconnected [BTV96,IR88], and find three independent spanning trees of G rooted at any vertex in $O(mn)$ and $O(n^2)$ time if G is triconnected [BTV96,CM88]. It is

J. Hromkovič, O. Sýkora (Eds.): WG'98, LNCS 1517, pp. 310–323, 1998.

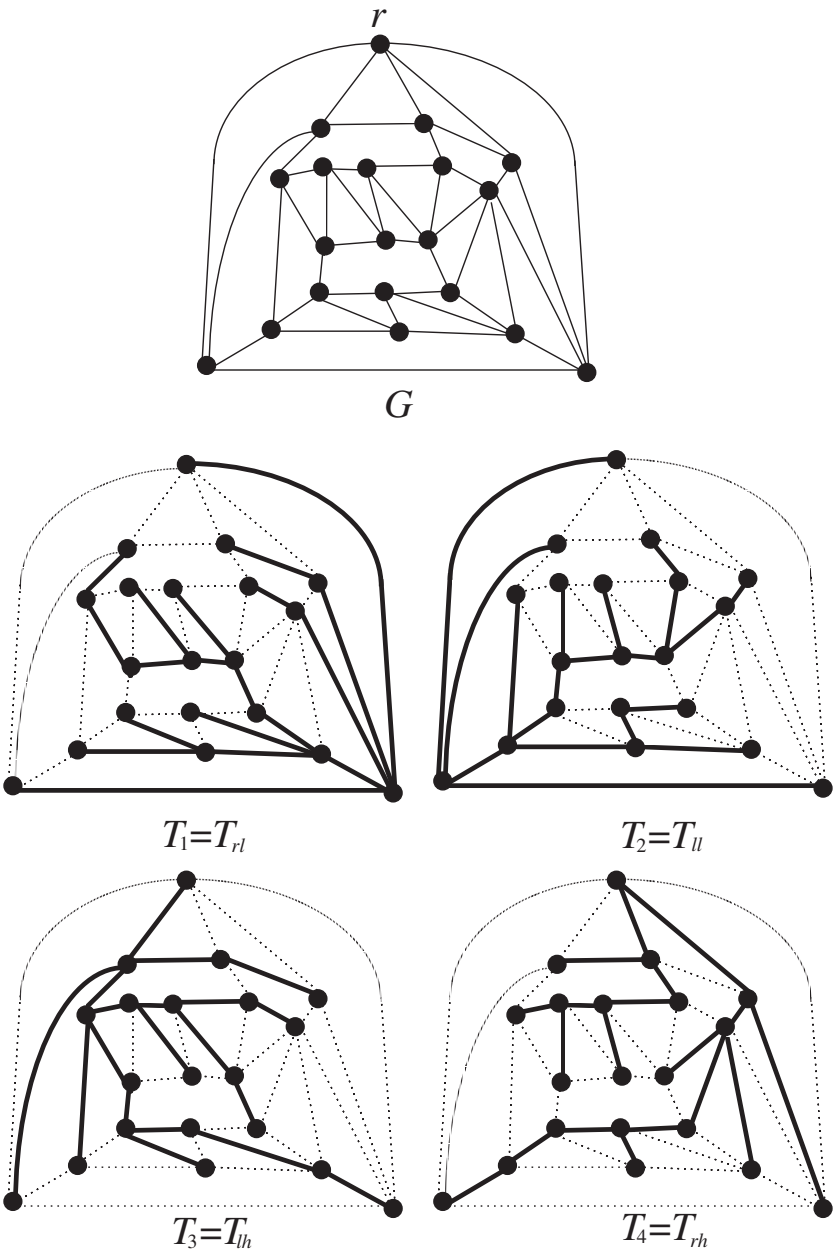

Fig. 1. Four independent spanning trees T_1, T_2, T_3 and T_4 of a graph G rooted at r.

conjectured that, for any $k \geq 1$, every k-connected graph has k independent spanning trees rooted at any vertex [KS92,ZI89]. Recently Huck has proved that every 4-connected planar graph has four independent spanning trees rooted at any vertex [H94]. The proof in [H94] yields an algorithm to actually find four independent spanning trees, but it takes time $O(n^3)$.

In this paper we give a simple linear-time algorithm to find four independent spanning trees of a 4-connected planar graph rooted at any designated vertex. Our algorithm is based on a "4-canonical decomposition" of a 4-connected planar graph [NRN97], which is a generalization of an *st*-numbering [E79], a canonical ordering [CK93] and a canonical 4-ordering [KH94].

The remainder of the paper is organized as follows. In Section 2 we introduce some definitions. In Section 3 we present our algorithm to find four independent spanning trees. Finally we put conclusion in Section 4.

2 Preliminaries

In this section we introduce some definitions.

Let $G = (V, E)$ be a connected graph with vertex set V and edge set E. Throughout the paper we denote by n the number of vertices in G, and we always assume that $n > 4$. An edge joining vertices u and v is denoted by (u, v). The *degree* of a vertex v in G, denoted by $d(v, G)$ or simply by $d(v)$, is the number of neighbors of 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. A graph G is *k-connected* if $\kappa(G) \geq k$. A *path* in a graph is an ordered list of distinct vertices $v_1, v_2, \cdots, v_l$ such that $v_{i-1}v_i$ is an edge for all i, $2 \leq i \leq l$. We say that two paths having common start and end vertices are *internally disjoint* if their intermediate vertices are disjoint. We also say that a set of paths having common start and end vertices are *internally disjoint* if every pair of paths in the set are internally disjoint.

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. The *contour* $C_o(G)$ of a biconnected plane graph G is

the clockwise (simple) cycle on the outer face. We write $C_o(G) = (w_1, w_2, \cdots, w_h)$ if the vertices $w_1, w_2, \cdots, w_h$ on $C_o(G)$ appear in this order.

3 Algorithm

In this section we give our algorithm to find four independent spanning trees of a 4-connected planar graph rooted at any designated vertex.

Given a 4-connected planar graph $G = (V, E)$ and a designated vertex $r \in V$, we first find a planar embedding of G in which r is located on $C_o(G)$. Let $G' = G - \{r\}$ be the subgraph of the plane graph G induced by $V - \{r\}$. In Fig. 2 (a) G is drawn by solid and dotted lines, and G' by solid lines. Since G is 4-connected, $d(r) \geq 4$. We may assume that all the neighbors $r_1, r_2, \cdots, r_{d(r)}$ of r in G appear on $C_o(G')$ clockwise in this order. Let $C_o(G') = (w_1, w_2, \cdots, w_h)$, $r_1 = w_1, r_2 = w_a, r_3 = w_b$ and $r_4 = w_c$, where $1 < a < b < c \leq d(r)$. We add to G' two new vertices r_b and r_t, join r_b with r_1 and r_2, and join r_t with $r_3, r_4, \cdots, r_{d(r)}$. Let G'' be the resulting plane graph, where vertices r_1, r_b, r_2, r_3, r_t and $r_{d(r)}$ appear on $C_o(G'')$ clockwise in this order. Fig. 2 (b) illustrates G''.

Let $\Pi = (W_1, W_2, \cdots, W_m)$ be a partition of the vertex set $V - \{r\}$ of G'. We denote by G_k, $1 \leq k \leq m$, the plane subgraph of G'' induced by $\{r_b\} \bigcup W_1 \bigcup W_2 \bigcup \cdots \bigcup W_k$. We denote by $\overline{G_k}$, $0 \leq k \leq m-1$, the plane subgraph of G'' induced by $W_{k+1} \bigcup W_{k+2} \bigcup \cdots \bigcup W_m \bigcup \{r_t\}$. We assume that if $1 \leq k \leq m$ and $W_k = \{u_1, u_2, \cdots, u_l\}$ then vertices $u_1, u_2, \cdots, u_l$ consecutively appear on $C_o(G_k)$ clockwise in this order. A partition $\Pi = (W_1, W_2, \cdots, W_m)$ of $V - \{r\}$ is called a *4-canonical decomposition* of G' if the following three conditions (co1)–(co3) are satisfied.

(co1) $W_1 = \{w_a, w_{a-1}, \cdots, w_1\}$ and $W_m = \{w_b, w_{b+1}, \cdots, w_c\}$;
(co2) For each k, $1 \leq k \leq m-1$, both G_k and $\overline{G_{k-1}}$ are biconnected (See Fig. 3.); and
(co3) For each k, $1 < k < m$, one of the following three conditions holds (See Fig. 3.):
  (a) $|W_k| \geq 2$, and each vertex $u \in W_k$ satisfies $d(u, G_k) = 2$ and $d(u, \overline{G_{k-1}}) \geq 3$;

(b) $|W_k| = 1$, and the vertex $u \in W_k$ satisfies $d(u, G_k) \geq 2$ and $d(u, \overline{G_{k-1}}) \geq 2$; and
(c) $|W_k| \geq 2$, and each vertex $u \in W_k$ satisfies $d(u, G_k) \geq 3$ and $d(u, \overline{G_{k-1}}) = 2$.

Fig. 2 (b) illustrates a 4-canonical decomposition of $G' = G - \{r\}$, where G' are drawn in solid lines and each set W_i is indicated by an oval drawn in a dotted line. A 4-canonical decomposition is a generalization of an "*st*-numbering" [E79], a "canonical decomposition" [CK93] and a "canonical 4-ordering" [KH94]. Although the definition of a 4-canonical decomposition above is slightly different from one in [NRN97], they are effectively equivalent each other.

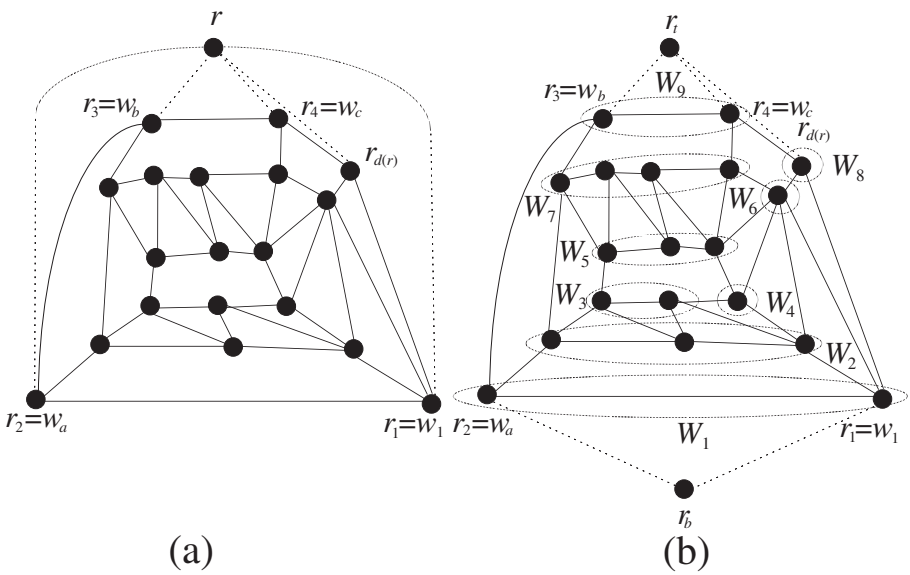

Fig. 2. (a) Four-connected plane graph G and (b) plane graph G''.

We have the following lemma.

Lemma 1. Let $G = (V, E)$ be a 4-connected plane graph, and let r be a designated vertex on $C_o(G)$. Then $G' = G - \{r\}$ has a 4-canonical decomposition Π. Furthermore Π can be found in linear time.

Proof. Similar to the proof of Lemma 3 in [NRN97]. $\mathcal{Q.E.D.}$

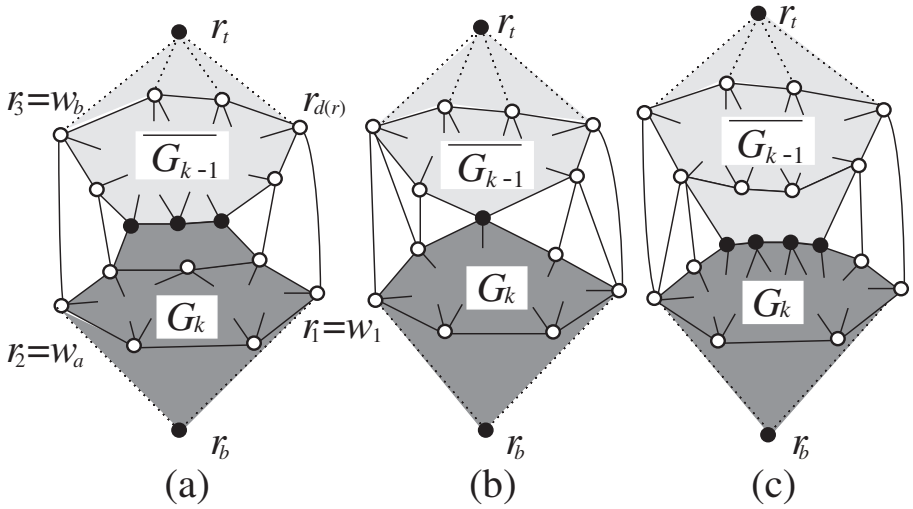

Fig. 3. Three conditions for (co3).

We need a few more definitions to describe our algorithm. For a vertex $v \in V-\{r\}$ we write $N(v) = \{v_1, v_2, \cdots, v_{d(v)}\}$ if $v_1, v_2, \cdots, v_{d(v)}$ are the neighbors of vertex v in G'' and appear around v clockwise in this order. To each vertex $v \in V - \{r\}$ we assign four edges incident to v in G''as *the left hand* $lh(v)$, *the right hand* $rh(v)$, *the left leg* $ll(v)$ and *the right leg* $rl(v)$ as follows. We will show later that such an assignment immediately yields four independent spanning trees of G. Let $v \in W_k$ for some k, $1 \leq k \leq m$, then there are the following three cases to consider.

Case 1: either (i) $1 < k < m$ and W_k satisfies Condition (a) of (co3) or (ii) $k = 1$. (See Fig. 4.)
Let $W_k = \{u_1, u_2, \cdots, u_l\}$. Let u_0 be the vertex on $C_o(G_k)$ preceding u_1, and let u_{l+1} be the vertex on $C_o(G_k)$ succeeding u_l. For each $u_i \in W_k$ we define $rl(u_i) = (u_i, u_{i+1})$, $ll(u_i) = (u_i, u_{i-1})$, $lh(u_i) = (u_i, v_1)$, and $rh(u_i) = (u_i, v_{d(u_i)-2})$ where we assume $N(u_i) = \{u_{i-1}, v_1, v_2, \cdots, v_{d(u_i)-2}, u_{i+1}\}$.

Case 2: W_k satisfies Condition (b) of (co3). (See Fig. 5.)
Let $W_k = \{u\}$, let u' be the vertex on $C_o(G_k)$ preceding u, and let u'' be the vertex on $C_o(G_k)$ succeeding u. Let $N(u) = \{u', v_1, v_2, \cdots, v_{d(u)-1}\}$, and let $u'' = v_x$ for some x, $3 \leq x \leq$

$d(u) - 1$. Then $rl(u) = (u, u'')$, $ll(u) = (u, u')$, $lh(u) = (u, v_1)$, and $rh(u) = (u, v_{x-1})$.

Case 3: either (i) $1 < k < m$ and W_k satisfies Condition (c) of (co3) or (ii) $k = m$. (See Fig. 6.)

Let $W_k = \{u_1, u_2, \cdots, u_l\}$. Let u_0 be the vertex on $\overline{C_o(G_{k-1})}$ succeeding u_1, and let u_{l+1} be the vertex on $\overline{C_o(G_{k-1})}$ preceding u_l. For each $u_i \in W_k$ we define $rl(u_i) = (u_i, v_1)$, $ll(u_i) = (u_i, v_{d(u_i)-2})$, $lh(u_i) = (u_i, u_{i-1})$, and $rh(u_i) = (u_i, u_{i+1})$ where we assume $N(u_i) = \{u_{i+1}, v_1, v_2, \cdots, v_{d(u_i)-2}, u_{i-1}\}$.

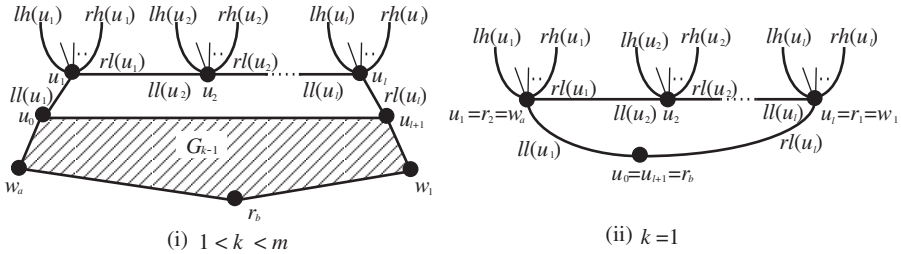

Fig. 4. Assignment for Case 1.

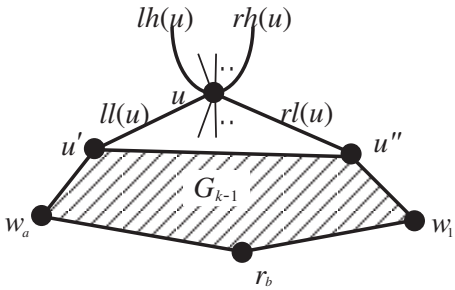

Fig. 5. Assignment for Case 2.

We are now ready to give our algorithm.

Procedure FourTrees(G, r)
begin
1 Find a planar embedding of G such that $r \in C_o(G)$;

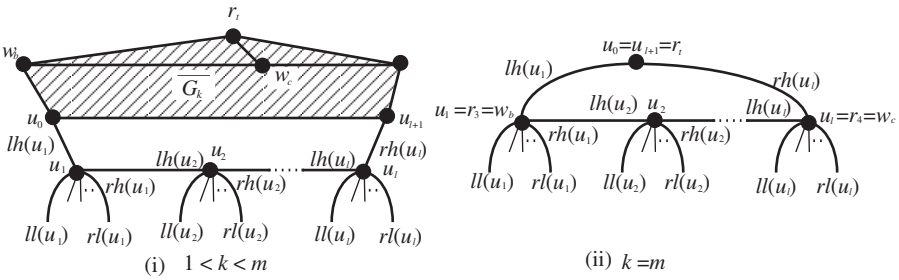

Fig. 6. Assignment for Case 3.

Find a 4-canonical decomposition $\Pi = (W_1, W_2, \cdots, W_m)$ of $G - \{r\}$;
For each vertex $v \in V - \{r\}$ find $rl(v), ll(v), rh(v)$ and $lh(v)$;
Let T_{rl} be a graph induced by the right legs of all vertices in $V - \{r\}$;
Let T_{ll} be a graph induced by the left legs of all vertices in $V - \{r\}$;
Let T_{lh} be a graph induced by the left hands of all vertices in $V - \{r\}$;
Let T_{rh} be a graph induced by the right hands of all vertices in $V - \{r\}$;
Regard vertex r_b in trees T_{rl} and T_{ll} as vertex r;
Regard vertex r_t in trees T_{lh} and T_{rh} as vertex r;
return T_{rl}, T_{ll}, T_{lh} and T_{rh} as four independent spanning trees of G.
end

We then verify the correctness of our algorithm. Assume that $G = (V, E)$ is a 4-connected planar graph with a designated vertex $r \in V$, and that Algorithm FourTrees finds a 4-canonical decomposition $\Pi = (W_1, W_2, \cdots, W_m)$ of $G - \{r\}$ and outputs T_{rl}, T_{ll}, T_{lh} and T_{rh}. We first have the following lemma.

Lemma 2. Let $1 \leq k \leq m$, and let T_{rl}^k be a graph induced by the right legs of all vertices in $G_k - \{r_b\}$. Then T_{rl}^k is a spanning tree of G_k.

Proof. We prove the claim by induction on k.

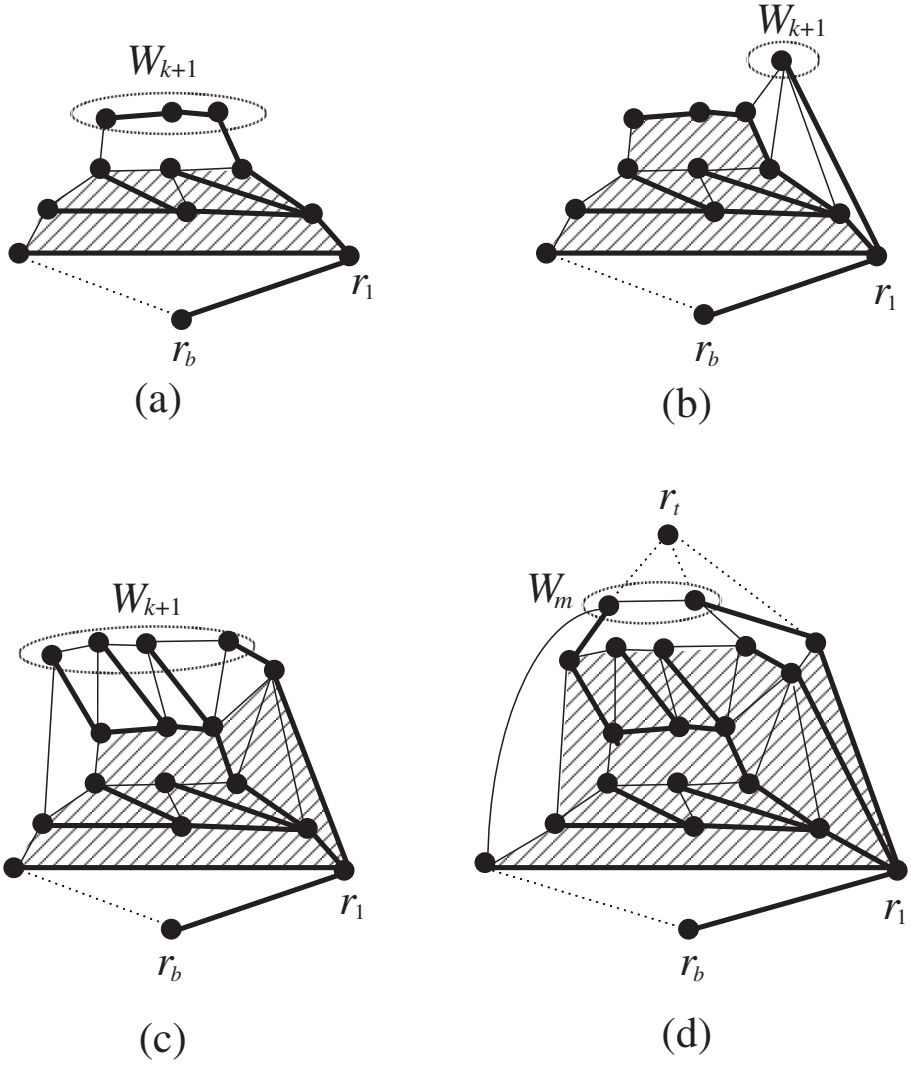

Fig. 7. The four cases for Lemma 2.

Clearly the claim holds for $k = 1$.

We assume that $1 \leq k \leq m-1$ and T_{rl}^k is a spanning tree of G_k, and we shall prove that T_{rl}^{k+1} is a spanning tree of G_{k+1}. There are the following four cases to consider.

Case 1: $k \leq m-2$ and W_{k+1} satisfies Condition (a) of (co3).
Case 2: $k \leq m-2$ and W_{k+1} satisfies Condition (b) of (co3).
Case 3: $k \leq m-2$ and W_{k+1} satisfies Condition (c) of (co3).
Case 4: $k = m-1$.

For each case T_{rl}^{k+1} is a spanning tree of G_{k+1} as shown in Fig. 7; (a) for Case 1; (b) for Case 2; (c) for Case 3; and (d) for Case 4. $\mathcal{Q.E.D.}$

We then have the following lemma.

Lemma 3. T_{rl}, T_{ll}, T_{lh} and T_{rh} are spanning trees of G.

Proof. By Lemma 3.2 T_{rl}^m is a spanning tree of G_m, and hence T_{rl} in which r_b is regarded as r is a spanning tree of G.

Similarly T_{ll}, T_{lh} and T_{rh} are spanning trees of G. $\mathcal{Q.E.D.}$

Let v be any vertex in $V - \{r\}$, and let P_{rl}, P_{ll}, P_{lh} and P_{rh} be the paths connecting r and v in T_{rl}, T_{ll}, T_{lh} and T_{rh}, respectively. For any vertex u in $V - \{r\}$ we write $rank(u) = k$ if $u \in W_k$; $rank(r)$ is undefined. If an edge (v, u) of G' is a leg of vertex v, and (v, w) of G' is a hand of v, then $rank(u) \leq rank(v) \leq rank(w)$ and $rank(u) < rank(w)$.

Lemma 4. Each of the four pairs of paths, P_{rl} and P_{lh}, P_{rl} and P_{rh}, P_{ll} and P_{lh}, P_{ll} and P_{rh}, are internally disjoint.

Proof. We prove only that P_{rl} and P_{lh} are internally disjoint. Proofs for the other pairs are similar. If $v = r_1$ then $P_{rl} = (v, r)$. If $v = r_3$ then $P_{lh} = (v, r)$. Therefor P_{rl} and P_{lh} are internally disjoint if v is r_1 or r_3. Thus we may assume that $v \neq r_1, r_3$. Let $P_{rl} = (v, v_1, v_2, \cdots, v_l, r)$, then $v_l = r_1$. Let $P_{lh} = (v, u_1, u_2, \cdots, u_{l'}, r)$, then $u_{l'} = r_3$. The definition of a right leg implies that $rank(v) \geq rank(v_1) \geq rank(v_2) \geq \cdots \geq rank(v_l)$, and the definition of a left hand implies that $rank(v) \leq rank(u_1) \leq rank(u_2) \leq \cdots \leq rank(u_{l'})$. Thus $rank(v_l) \leq \cdots \leq rank(v_2) \leq rank(v_1) \leq rank(v) \leq rank(u_1) \leq rank(u_2) \leq \cdots \leq rank(u_{l'})$. We furthermore have $rank(v_1) < rank(u_1)$. Therefore P_{rl} and P_{lh} are internally disjoint. $\mathcal{Q.E.D.}$

We next have the following lemma.

Lemma 5. Let $u \in V - \{r\}$, $ll(u) = (u, u')$, $rl(u) = (u, u'')$, and $N(u) = \{v_1, v_2, \cdots, v_{d(u)}\}$. One may assume that $u' = v_1$ and $u'' = v_s$ for some s, $1 < s \leq d(u)$. Then there exists t, $1 \leq t \leq s$, such that $rl(v_i) = (v_i, u)$ for each i, $2 \leq i \leq t-1$, and $ll(v_j) = (v_j, u)$ for each j, $t+1 \leq j \leq s-1$. (Thus either (i) $rl(v_t) = (v_t, u) \neq ll(v_t)$, (ii) $rl(v_t) \neq (v_t, u) = ll(v_t)$, or (iii) $rl(v_t) \neq (v_t, u) \neq ll(v_t)$. See Fig. 8.)

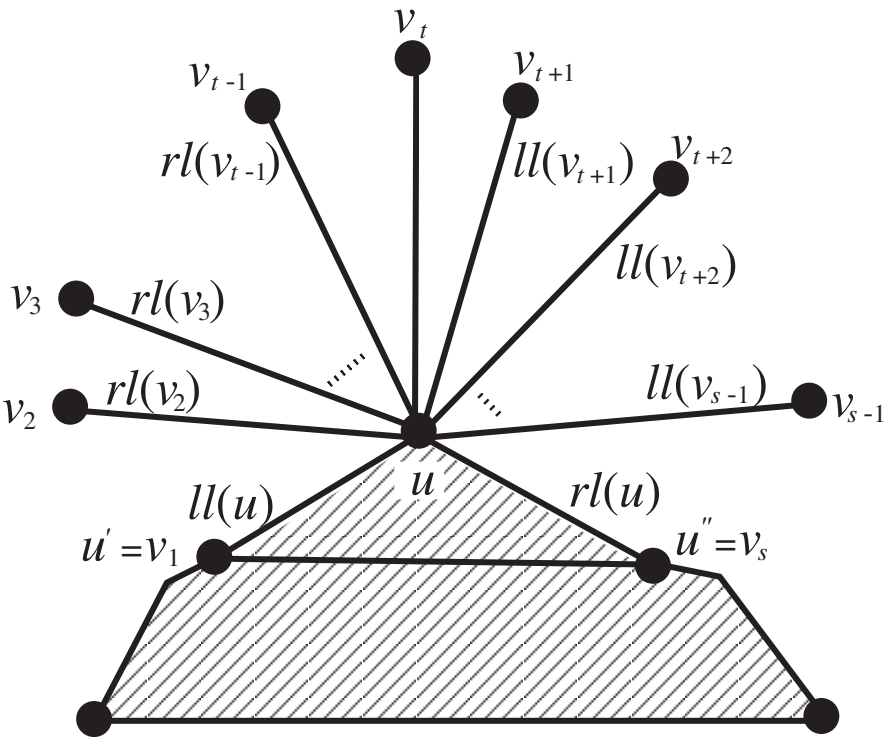

Fig. 8. Illustration for Lemma 5.

Proof. From the definitions of a 4-canonical decomposition and a right leg, one can observe that if $2 \leq i \leq s-1$ and $rl(v_i) = (v_i, u)$ then $rank(v_{i-1}) < rank(v_i)$. Similarly, if $2 \leq i \leq s-1$ and $ll(v_j) = (v_j, u)$ then $rank(v_j) > rank(v_{j+1})$.

Assume for a contradiction that the claim does not hold. Then $rl(v_i) = (v_i, u)$ and $ll(v_j) = (v_j, u)$ for some i and j, $1 \leq j < i \leq s$. Let $v_i \in W_{i'}$ and $v_j \in W_{j'}$ for some i' and j', $1 \leq i', j' \leq m$. Thus $rank(v_i) = i'$, $rank(v_j) = j'$, and both $G_{i'}$ and $G_{j'}$ are biconnected. There are the following three cases.

Case 1: $i' = j'$. In this case, $G_{i'}$ has edges (u, v_j) and (v_i, u), and all vertices in $W_{i'}$ appear on $C_o(G_{i'})$. Therefore, vertex u and the vertices in $W_{i'}$ from v_j to v_i form a cycle in $G_{i'}$, and $G_{i'}$ has at least one vertex in the proper inside of the cycle. None of the edges of G in the outside of the cycle is incident to any vertex on the cycle other than u, v_j and v_i. Hence the removal of three vertices u, v_j and v_i from G results in a disconnected graph, contrary to the 4-connectivity of G.

Case 2: $i' < j'$. Since $rl(v_i) = (v_i, u)$, v_i precedes u on $C_o(G_{i'})$. Since $ll(v_j) = (v_j, u)$, v_j succeeds u on $C_o(G_{j'})$. Since $G_{i'}$ is a subgraph of $G_{j'}$, v_i must precede v_j in $N(u)$, contrary to the assumption $j < i$.

Case 3: $i' > j'$. Similar to Case 2 above. $\mathcal{Q.E.D.}$

Lemma 5 immediately implies the following lemma.

Lemma 6. P_{rl} and P_{ll} may cross at a vertex u, but do not share a vertex u without crossing at u.

From the definitions of a left leg and a right leg one can immediately have the following lemma.

Lemma 7. Let $1 \leq k \leq m$ and $u \in W_k$. Then u is on $C_o(G_k)$. Let u' be the succeeding vertex of u on $C_o(G_k)$. Assume that the ordered set $N(u)$ starts with u'. Let $ll(u) = (u, v')$ and $rl(u) = (u, v'')$. Then v'' precedes v' in $N(u)$.

We then have the following lemma.

Lemma 8. Each of the two pairs of paths, P_{rl} and P_{ll}, P_{lh} and P_{rh}, are internally disjoint.

Proof. We prove only that P_{rl} and P_{ll} are internally disjoint. Proof for the other case is similar. Suppose for a contradiction that P_{rl} and P_{ll} share an intermediate vertex. Let w be the intermediate vertex

that is shared by P_{rl} and P_{ll} and appear last on the path P_{rl} going from r to v. Then P_{rl} and P_{ll} cross at w by Lemma 6. However, the claim in Lemma 7 holds both for $k = rank(v)$ and $u = v$ and for $k = rank(w)$ and $u = w$, and hence P_{rl} and P_{ll} do not cross at w, a contradiction. $\mathcal{Q.E.D.}$

By Lemmas 3, 4 and 8 we have the following lemma.

Lemma 9. T_{rl}, T_{ll}, T_{lh} and T_{rh} are four independent spanning trees of G rooted at r.

Clearly the running time of Algorithm FourTrees is $O(n)$. Thus we have the following theorem.

Theorem 1. Four independent spanning trees of any 4-connected plane graph rooted at any designated vertex can be found in linear time.

4 Conclusion

In this paper we give a linear-time algorithm to find four independent spanning trees of a 4-connected planar graph rooted at any designated vertex. Using four independent spanning trees, one can efficiently solve the 4-path query problem for 4-connected planar graphs.

It is remained as future work to find a linear-time algorithm for a larger class of graphs, say 4-connected graphs which are not always planar.

References

BI96. F. Bao and Y. Igarashi, *Reliable broadcasting in product networks with Byzantime faults*, Proc. 26th Annual International Symposium on Fault-Tolelant Computing (FTCS'96) (1996) 262-271.

BTV96. G. Di Battista, R. Tamassia and L.Vismara, *Output-sensitive reporting of disjoint paths*, Technical Report CS-96-25, Department of Computer Science, Brown University (1996).

CK93. M. Chrobak and G. Kant, *Convex grid drawings of 3-connected planar graphs*, Technical Report RUU-CS-93-45, Department of Computer Science, Utrecht University (1993).

CM88. J. Cheriyan and S. N. Maheshwari, *Finding nonseparating induced cycles and independent spanning trees in 3-connected graphs*, J. Algorithms, 9 (1988) 507-537.

DHSS84. D. Dolev, J. Y. Halpern, B. Simons, and R. Strong, *A new look at fault tolerant network routing*, Proc. 16th Annual ACM Symposium on Theory of Computing (1984) 526-535.

E79. S. Even, *Graph Algorithms*, Computer Science Press, Potomac (1979).

H94. A. Huck, *Independent trees in Graphs*, Graphs and Combinatorics, 10 (1994) 29-45.

IR88. A. Itai and M. Rodeh, *The multi-tree approach to reliability in distributed networks*, Information and Computation, 79 (1988) 43-59.

KH94. G. Kant and X. He, *Two algorithms for finding rectangular duals of planar graphs*, Proc. 19th Workshop on Graph-Theoretic Concepts in Computer Science (WG'93), LNCS 790 (1994) 396-410.

KS92. S. Khuller and B. Schieber, *On independent spanning trees*, Information Processing Letters, 42 (1992) 321-323.

NRN97. S. Nakano, M. S. Rahman and T. Nishizeki, *A linear time algorithm for four-partitioning four-connected planar graphs*, Information Processing Letters, 62 (1997) 315-322.

OIBI96. K. Obokata, Y. Iwasaki, F. Bao and Y. Igarashi, *Independent spanning trees of product graphs and their construction*, Proc. 22nd Workshop on Graph-Theoretic Concepts in Computer Science (WG'96), LNCS 1197 (1996) 338-351.

ZI89. A. Zehavi and A. Itai, *Three tree-paths*, J. Graph Theory, 13 (1989) 175-188.