

**Edge Coloring Planar Graphs
With Two Outerplanar Subgraphs**

Lenwood S. Heath

TR 91-34

Department of Computer Science
Virginia Polytechnic Institute and State University
Blacksburg, Virginia 24061

December 30, 1991

EDGE COLORING PLANAR GRAPHS
WITH TWO OUTERPLANAR SUBGRAPHS

Lenwood S. Heath

Department of Computer Science

Virginia Polytechnic Institute and State University

Blacksburg, VA 24061

December 28, 1991

Abstract

The standard problem of edge coloring a graph with k colors is equivalent to partitioning the edge set of the graph into k matchings. Here edge coloring is generalized by replacing matchings with outerplanar graphs. We give a polynomial-time algorithm that edge colors any planar graph with two outerplanar subgraphs. Two is clearly minimal for the class of planar graphs.

1 Introduction

The problem of edge coloring a graph $H = (V, E)$ is to assign a color to each edge so that, for each $v \in V$, all edges incident to v have distinct colors. Usually, an edge coloring using as few colors as possible is desired; the minimum number of colors that can edge color H is $\chi'(G)$, the chromatic index of H . The problem has a rich algorithmic history [33, 14, 24]. In particular, Vizing [33] shows that, if Δ is the maximum degree of a graph H , then either $\chi'(G) = \Delta$ or $\chi'(G) = \Delta + 1$. Holyer [21] shows that the problem of determining $\chi'(G)$ is NP-complete.

Another way of viewing the problem of edge coloring H with k colors is to partition E into k subsets E_1, E_2, \dots, E_k such that each E_i , when viewed as a graph, is a matching. This view suggests a generalization of the problem. Let Q be any class of graphs. (Think of Q as defining some property a graph may have.) The problem of *edge coloring H with Q subgraphs* is the problem of coloring the edges of H so that each monochromatic subgraph is isomorphic to some graph in $G \in Q$. (In keeping with terminology from [17, 32], we think of H as a *host graph* and $G \in Q$ as a *guest graph*.) Every graph has an edge coloring with Q subgraphs if and only if Q contains a graph consisting of a single edge. When Q is the class of matchings, then we have the original edge coloring problem. For the case that Q consists of all complete graphs, Holyer [22] shows that edge coloring with complete subgraphs is NP-complete.

If Q is taken to be the class of forests (acyclic graphs), then results of Nash-Williams [28, 29] and Tutte [31] show that every planar graph can be edge colored with three acyclic subgraphs. The matroid partitioning algorithm of Edmonds [11] can edge color any planar graph with three acyclic subgraphs. Schnyder [30] gives an algorithm for finding three edge-disjoint trees that partition the edge set of a triangulated planar graph; this provides a second algorithm for edge coloring an arbitrary planar graph with three acyclic subgraphs.

Chvátal and Hammer [7] show that the problem of minimally edge coloring H with subgraphs that are threshold graphs is NP-complete. Caro [3] considers the problem of

edge coloring a tree with subtrees, each isomorphic to a tree from a fixed family of trees. Jünger, Reinelt, and Pulleyblank [23] investigate the problem of edge coloring a connected graph with some fixed number of connected subgraphs.

A k -tree, $k \geq 2$, is defined inductively as follows. The complete graph on k vertices is a k -tree. If H is a k -tree and $\{v_1, v_2, \dots, v_k\}$ is any k -clique of H , then the graph obtained by adding to H a new vertex v and new edges $(v, v_1), (v, v_2), \dots, (v, v_k)$ is a k -tree. A *partial k -tree* is a graph that is a subgraph of some k -tree. Elmallah and Colbourn [12] consider the problem of edge coloring a planar graph with two partial k -trees. They show that every planar graph can be edge colored with two partial 3-trees.

In this paper, we address the problem of edge coloring a planar graph with outerplanar subgraphs. Our main result is the following

Theorem 1 *Let H be a planar graph having n vertices. Then the edges of H can be colored with two colors so that each monochromatic subgraph is outerplanar. This coloring can be found in $O(n^2 \sqrt{\log n})$ time.*

As an outerplanar graph is a partial 2-tree, this improves the result in [12]. This result also settles in the affirmative the conjecture of Chartrand, Geller, and Hedetniemi [4] that every planar graph can be edge colored with two outerplanar subgraphs. The proof of this theorem utilizes a new technique for recursively decomposing a triangulated planar graph. This technique processes the graph recursively by identifying separating 3-cycles and 4-cycles; within 4-cycles, the technique finds progressively longer diagonal paths to color. Other separating structures are likely to be relevant to other edge coloring problems. A more informal presentation of this result appears in [16].

Applications of generalized edge coloring include approximation algorithms for NP-complete graph problems that are solvable in polynomial time for graphs in Q (for examples, see [17, 32]). Colbourn [8, 9] indicates how edge colorings can be applied to estimating the reliability of a network.

A related problem is *edge-packing H with Q subgraphs*, which is the problem of finding a set of edge-disjoint subgraphs of H such that each subgraph is isomorphic to some $G \in Q$. Yap [35] surveys the well-studied problem of edge-packing complete graphs with trees. Edge-packing has been studied especially for cases in which Q consists of a single graph G . In these cases, the problem becomes an optimization problem: Finding the maximum number of edge-disjoint copies of G in H . Masuyama and Ibaraki [27] show that edge-packing with a path of length 2 is solvable in linear time. Corneil, Masuyama, and Hakimi [10] investigate edge-packing an arbitrary host graph with a single connected guest graph G ; they show that this problem is NP-complete for every G which is not a path of length 2. Heath and Vergara [17, 32] study edge-packing an arbitrary planar host graph with a single connected guest graph; they obtain a number of NP-completeness and algorithmic results.

The analogous generalization for vertex coloring has also been studied. The problem of *vertex coloring with Q subgraphs* is to partition V into subsets V_1, V_2, \dots, V_k such that the subgraph induced by each V_i is isomorphic to some graph in Q . If Q is the class of isolated vertices, the usual vertex coloring problem is obtained. If Q contains just a graph with two vertices and a single edge, then the vertex coloring problem is equivalent to finding a perfect matching. Kirkpatrick and Hell [25] show that vertex coloring with Q subgraphs is NP-complete provided that Q consists of a single graph having a connected component of at least three vertices.

The related optimization problem for vertex coloring is Maximum G Matching (finding the maximum number of vertex-disjoint copies of G in H). Hell and Kirkpatrick [25, 18, 26, 19, 20] have a number of complexity results for this and related problems. In fact, it is solvable in polynomial time when the guest graph is an edge by matching, but is NP-complete for other connected guest graphs, even if H is planar. Kirkpatrick and Hell [25] prove NP-completeness for this problem in general (reduction from 3-DIMENSIONAL MATCHING), while Berman et al. [1] prove NP-completeness for planar hosts (reduction from PLANAR 3-SAT).

There is another generalization of edge coloring, book embedding, that looks much like the problem of edge coloring with outerplanar subgraphs [2, 6]. A *book embedding* of H orders V on a circle, draws each edge as a chord of the circle, and colors E such that no two chords of the same color intersect. Each monochromatic subgraph is outerplanar and, in fact, the book embedding gives an outerplanar embedding of it. The minimum number of colors required to book embed H is its *pagenumber*. Yannakakis [34] shows that the pagenumber of the class of planar graphs is four. Edge coloring with outerplanar subgraphs does not impose any order on V or other restrictions on the outerplanar subgraphs, so, as we show, two outerplanar subgraphs will suffice for edge coloring planar graphs.

2 The Structure of Outerplanar Graphs

In this section, we establish some terminology and notation and review some properties of outerplanar graphs.

We consider only undirected graphs without loops or multiple edges. If H is a graph, $V(H)$ denotes the set of vertices of H , while $E(H)$ denotes the set of edges of H . The edge incident to $u, v \in V(H)$ is written as either (u, v) or (v, u) and equals the set $\{u, v\}$. For a planar graph H , we assume a fixed planar embedding of H ; in particular, there is a unique unbounded face for H , called the *exterior* face. An *interior* face is one that is not exterior. Any subgraph of H inherits the planar embedding induced by the embedding of H . A *triangle* is a face of H having exactly three incident edges and vertices. A planar graph H is *triangulated* if all its faces are triangles. An *outerplanar* graph is a planar graph that can be embedded in the plane in such a way that all vertices are incident to the exterior face.

A path or a cycle is denoted by a sequence of vertices, e.g., u_1, u_2, \dots, u_k . All paths and cycles are *simple*, i.e., all vertices in the sequence are distinct. A *k-path*, $k \geq 0$, is a path of length k (i.e., having $k + 1$ vertices). A *k-cycle*, $k \geq 3$, is a cycle of length k (i.e., having k vertices). A cycle in a planar graph *separates* the plane into two connected regions, the

exterior and the *interior* of the cycle. If there are vertices or edges of the graph in the interior of a cycle, then the cycle is a *separating cycle*. A path u_1, u_2, \dots, u_k is a *diagonal path* of a separating cycle C if u_1 and u_k are on C and u_2, \dots, u_{k-1} are interior to C ; edges on a diagonal path are *diagonal edges*. By convention, the vertices of a cycle are given in a clockwise order.

For any outerplanar graph H (along with a fixed outerplanar embedding), we can place each edge in $\mathbf{E}(H)$ into exactly one of three subsets:

- $\mathbf{B}(H)$ *bounding edges*: edges that are on a cycle in H and incident to the exterior face;
- $\mathbf{C}(H)$ *chords*: edges that are on a cycle in H and not incident to the exterior face;
- $\mathbf{X}(H)$ *cut edges*: edges that are not on any cycle in H ; the removal of any cut edge from H increases the number of connected components by one.

Note that the classification of an edge into one of the three classes does not depend on the planar embedding of H : An edge (u, v) in an outerplanar H is a chord if and only if u and v are in the same biconnected component of $H - (u, v)$.

Let H be an arbitrary graph. Obtain a new graph H' by the operation of *edge subdivision*: select any edge $(u, v) \in \mathbf{E}(H)$ and replace it with the path u, z, v , where z is a new vertex. A graph obtained from H by zero or more applications of the edge subdivision operation is called an *edge subdivision* of H . A well-known characterization of outerplanar graphs is due to Chartrand and Harary [5]:

Proposition 2 [5] *A graph is outerplanar if and only if it contains no subgraph that is an edge subdivision of K_4 or $K_{2,3}$.*

3 The Algorithm COLOREDGE

This section is devoted to a constructive proof of Theorem 1 via an algorithm called COLOREDGE. In the first subsection, we sketch the principle ideas behind the algorithm. In the succeeding three subsections, we elaborate on the three recursive subroutines that constitute COLOREDGE. In the final subsection, we analyze the time complexity of COLOREDGE.

3.1 Overview of COLOREDGE

Without loss of generality, we may assume that H is triangulated. At any time during the execution of COLOREDGE, each edge has a color assigned via the *coloring function*

$$\Gamma : \mathbf{E}(H) \rightarrow \{\perp, \mathbf{blue}, \mathbf{red}\}.$$

An edge assigned the color \perp is said to be *uncolored*; all other edges are *colored*. (In all figures, the waffle pattern is **red** and the plaid pattern is **blue**. A solid edge is ambiguous; it may or may not be colored.) If $\gamma \in \{\mathbf{blue}, \mathbf{red}\}$, then $\bar{\gamma}$ is the *opposite color*:

$$\bar{\gamma} = \begin{cases} \mathbf{red} & \text{if } \gamma = \mathbf{blue} \\ \mathbf{blue} & \text{if } \gamma = \mathbf{red}. \end{cases}$$

At any time during the execution of COLOREDGE, $\mathbf{blue}(H)$ is the subgraph of H induced by the edges that are currently colored **blue**; similarly, $\mathbf{red}(H)$ is the subgraph of H induced by the edges that are currently colored **red**. Once COLOREDGE has colored an edge **blue** or **red**, its color never changes. Initially,

$$\Gamma(u, v) = \perp \quad \forall (u, v) \in \mathbf{E}(H).$$

(We simplify notation by writing $\Gamma(u, v)$, rather than $\Gamma((u, v))$.) When COLOREDGE completes execution,

$$\Gamma(\mathbf{E}(H)) \subseteq \{\mathbf{blue}, \mathbf{red}\},$$

$\text{blue}(H)$ is outerplanar, and $\text{red}(H)$ is outerplanar.

At any time during the execution of COLOREDGE and after the assignment of a color **blue** or **red** to (u, v) , $\Gamma(u, v)(H)$ is, of course, outerplanar. Furthermore, (u, v) is unambiguously a bounding edge, a chord, or a cut edge of $\Gamma(u, v)(H)$. Hence, we will often say that a colored edge (u, v) is a cut edge (or a bounding edge or a chord) to mean that it is in $\mathbf{X}(\Gamma(u, v)(H))$ (or $\mathbf{B}(\Gamma(u, v)(H))$ or $\mathbf{C}(\Gamma(u, v)(H))$).

From Proposition 2, it suffices to color the edges of H such that neither $\text{red}(H)$ nor $\text{blue}(H)$ contains a subgraph that is an edge subdivision of K_4 or $K_{2,3}$. Note that any edge subdivision of K_4 , other than K_4 itself, contains an edge subdivision of $K_{2,3}$. Therefore, we may concentrate on avoiding edge subdivisions of $K_{2,3}$ and treat K_4 as a special case.

In a planar embedding, a K_4 appears as three vertices in a cycle with a fourth vertex, adjacent to the first three, in the interior of the cycle. A $K_{2,3}$ appears as a separating 4-cycle with a diagonal path of length 2 in the interior. COLOREDGE identifies all separating 3-cycles, all separating 4-cycles without diagonal 1-paths, and some larger separating cycles called *halls* (defined in the next paragraph); it colors the edges in their interiors recursively. To this purpose, the algorithm has three recursive subroutines: THREECYCLE, FOURCYCLE, and HALL.

A *hall* M is a separating cycle which is specified by two paths $u = u_0, u_1, \dots, u_s = v$, $s \geq 2$, and $u = v_0, v_1, \dots, v_t = v$, $t \geq 2$. The first path is the *left wall* of M , and the second path is the *right wall*. u is the *top* of M . v is the *bottom* of M . Except for $u_0 = v_0$ and $u_s = v_t$, the two walls are vertex-disjoint. The clockwise order of M is the left wall in order from the bottom followed by the right wall in order from the top:

$$v = u_s, u_{s-1}, \dots, u_0 = u = v_0, v_1, \dots, v_{t-1}, v_t = v.$$

When any of the three subroutines is called, the edges of the enclosing separating cycle have been colored, and all edges interior to the cycle remain to be colored. Henceforth, we only use separating cycle to refer to a cycle, all of whose edges are colored and none

of whose interior edges are colored; for convenience of exposition, we also call a colored 3-cycle, even without interior edges, a separating cycle. FOURCYCLE is never called if the 4-cycle has a diagonal 1-path interior to it, forming two separating 3-cycles; instead, two calls are made to THREECYCLE.

The subroutines need a bit more information than just the colors of the outermost edges. Consider the classification of a **red** edge (u, v) at some intermediate stage of coloring H . If $(u, v) \in \mathbf{C}(\mathbf{red}(H))$ is currently a chord, then a new **red** path must not be created from u to v . If $(u, v) \in \mathbf{B}(\mathbf{red}(H))$ is currently a bounding edge, then one **red** path, at most, may be created from u to v . If $(u, v) \in \mathbf{X}(\mathbf{red}(H))$ is currently a cut edge, then two **red** paths, at most, may be created from u to v . When first colored, the edge (u, v) belongs to at most two currently-colored separating cycles; unless (u, v) is on the exterior face, it belongs to *exactly* two separating cycles. Whenever a **red** path may be created, it may be created in the interior of either of the separating cycles. The subroutines must know where **red** (or **blue**) paths are allowed and within which separating cycles the paths are allowed. If a path of the same color as the edge is allowed within a separating cycle, that edge is said to be *open* inside the cycle; otherwise, that edge is *closed* inside the cycle. A chord is closed inside both its separating cycles. A cut edge is open inside both its separating cycles. A bounding edge is open inside one separating cycle and closed inside the other; one flexibility that COLOREDDGE has is that it may assign either separating cycle of a bounding edge to be the one inside which the edge is open, before the recursive calls for its two separating cycles are made. We use the notation

$$\text{Open}(e_1, e_2, \dots, e_n; A)$$

to mean that each of the edges e_1, e_2, \dots, e_n is an edge of cycle A and that each is open inside A .

To illustrate the concept of openness, we prove that a particular family of triangulated planar graphs can be edge colored with two outerplanar subgraphs. If H is a triangulated planar graph, a *stellation operation* on H produces a triangulated planar graph H' that

is gotten by choosing an arbitrary interior face of H , say bounded by the 3-cycle u, v, w , placing a new vertex z inside that face, and adding new edges (u, z) , (v, z) , and (w, z) . A *stellation* of H is any triangulated planar graph that can be obtained from H by the application of zero or more stellation operations. The family $ST(K_3)$ of triangulated planar graphs consists of all stellations of K_3 . (See [2, 15].)

Theorem 3 *Every graph in $ST(K_3)$ can be edge colored with two outerplanar subgraphs in such a way that each of its interior faces has an edge open inside the face.*

Proof: The proof is by induction on the number of stellation operations that are required to obtain an $H' \in ST(K_3)$ from K_3 . If $H' = K_3$, then color two edges of H' **blue** and the third edge **red**; any edge can be chosen to be open inside the one interior face. Now suppose H' is obtained from $H \in ST(K_3)$ by the addition of the vertex z inside the triangle of H bounded by u, v, w . By the inductive hypothesis, H can be edge colored with two outerplanar subgraphs in such a way that each of its interior faces has an edge open inside the face. In particular, an edge of u, v, w is open inside the triangle. Without loss of generality, (u, v) is open inside the triangle, and $\Gamma(u, v) = \mathbf{red}$. Extend the coloring function as follows:

$$\begin{aligned}\Gamma(u, z) &= \mathbf{red} \\ \Gamma(v, z) &= \mathbf{red} \\ \Gamma(w, z) &= \mathbf{blue}.\end{aligned}$$

Now (u, z) and (v, z) are bounding edges (of $\mathbf{red}(H')$), while (w, z) is a cut edge (of $\mathbf{blue}(H')$). The faces of H' inherit the open edges of H except for the three new faces, for which the following hold:

$$\begin{aligned}\text{Open}((u, z); u, v, z) \\ \text{Open}((w, z); v, w, z) \\ \text{Open}((w, z); w, u, z).\end{aligned}$$

By induction, the theorem follows. □

As a typical member of $ST(K_3)$ is rife with copies of K_4 , the above proof indicates that the presence of K_4 in a planar graph is not a particular difficulty for edge coloring with outerplanar subgraphs.

To justify the correctness of FOURCYCLE and HALL, there is a need to extend the concept of openness to 2-paths. A 2-path x, y, z on a cycle A is *open* inside A , denoted $\text{Open}(x, y, z; A)$, if $\Gamma(x, y) = \Gamma(y, z)$ and if y is a cutpoint of $\Gamma(x, y)(H)$ whose removal disconnects x and z . Hence, a path of color $\Gamma(x, y)$ may be added between x and z without destroying the outerplanarity of $\Gamma(x, y)(H)$.

Each of the three subroutines has a list of requirements that its inputs must satisfy. As each of the three subroutines is recursive, we call these requirements the *inductive hypotheses* for the subroutine. The correctness of COLOREDGE follows from proofs that the input of each subroutine always satisfies its inductive hypotheses and that no subroutine ever colors an edge in a way that creates a non-outerplanar $\text{blue}(H)$ or $\text{red}(H)$.

COLOREDGE begins by triangulating the planar graph. The exterior 3-cycle is colored arbitrarily, and a call to THREECYCLE completes the coloring. A complete presentation of each of the three subroutines and of their inductive hypotheses follows in the next three subsections.

3.2 THREECYCLE

The call $\text{THREECYCLE}(u, v, w)$ passes the cycle u, v, w , together with its interior edges, to THREECYCLE. Without loss of generality, we may assume that u, v, w has non-empty interior; if u, v, w has empty interior, then THREECYCLE has no coloring to do and just returns. The inductive hypotheses for THREECYCLE are:

IH 1.1 All three edges of the cycle u, v, w are colored;

IH 1.2 $\text{Open}((u, v); u, v, w)$.

Note that because of IH 1.2, the order u, v, w matters. By symmetry of **blue** and **red**, without loss of generality, we may assume that $\Gamma(u, v) = \text{red}$. **THREECYCLE** has essentially two cases.

In the first case, there is a vertex x interior to u, v, w and adjacent to each of u, v , and w , forming a K_4 (Figure 1). The coloring function Γ is extended as follows:

$$\begin{aligned}\Gamma(u, x) &= \text{red} \\ \Gamma(v, x) &= \text{red} \\ \Gamma(w, x) &= \text{blue}\end{aligned}$$

(see Figure 2). Now (u, x) and (v, x) are bounding edges, while (w, x) is a cut edge. These three recursive calls to **THREECYCLE** finish up:

$$\begin{aligned}\text{THREECYCLE}(x, u, v) \\ \text{THREECYCLE}(x, w, u) \\ \text{THREECYCLE}(w, x, v).\end{aligned}$$

It is evident that IH 1.1 is satisfied for these three calls. Since (x, u) is a bounding edge and $(x, w) = (w, x)$ is a cut edge, IH 1.2 is also satisfied.

In the second case, there is no interior vertex x to form a K_4 . Identify vertex z , interior to cycle u, v, w and adjacent to u and w so that cycle u, w, z encloses as many vertices as possible (Figure 3). The coloring function Γ is extended as follows:

$$\begin{aligned}\Gamma(u, z) &= \text{red} \\ \Gamma(w, z) &= \text{blue}\end{aligned}$$

(see Figure 4). One call to **THREECYCLE** and one to **FOURCYCLE** finish up:

THREECYCLE(u, z, w)

FOURCYCLE(z, u, v, w).

Since (u, z) is a cut edge, IH 1.2 is satisfied for the call to THREECYCLE. Refer to Subsection 3.3 for inductive hypotheses IH 2.1, IH 2.2, and IH 2.3 of FOURCYCLE. It is evident that IH 2.1 and IH 2.2 are satisfied for the call to FOURCYCLE. Since $(w, z) \in X(\text{blue}(H))$ and $\text{Open}((u, v); z, u, v, w)$, IH 2.3 is satisfied for the call to FOURCYCLE.

3.3 FOURCYCLE

The call FOURCYCLE(x, u, y, v) passes the cycle x, u, y, v , together with its interior edges, to FOURCYCLE. The inductive hypotheses for FOURCYCLE are:

IH 2.1 All four edges of the cycle x, u, y, v are colored;

IH 2.2 Neither (x, y) nor (u, v) is an interior edge of the cycle x, u, y, v ;

IH 2.3 $\text{Open}((u, y); x, u, y, v)$, $\text{Open}(y, v, x; x, u, y, v)$, and (u, y) and (v, x) are oppositely colored.

For an illustration, see Figure 5. We think of the path u, x, v as being on the left of the cycle and the path u, y, v as being on the right. By symmetry, we may assume that $\Gamma(u, y) = \text{red}$ and $\Gamma(v, x) = \text{blue}$. FOURCYCLE has two main cases, depending on whether the cycle x, u, y, v has any diagonal 2-paths.

3.3.1 Diagonal 2-paths

In the first main case, there is either a diagonal 2-path u, w, v or a diagonal 2-path x, z, y . If both path u, w, v and path x, z, y exist, then $w = z$ (by planarity), and FOURCYCLE takes u, w, v as the diagonal 2-path in the following description. Otherwise, without loss

of generality, we may assume that there are one or more diagonal 2-paths between u and v . See Figure 6. For purposes of orientation, think of x as the left, u as the top, y as the right, and v as the bottom. This orientation defines a left-to-right order on the diagonal 2-paths between u and v .

The goal in this case is to color the edges of the 2-paths so as to avoid a monochromatic $K_{2,3}$. Let the 2-paths be $u, z_i, v, 1 \leq i \leq r$, ordered left to right. For notational convenience, let $z_0 = x$ and $z_{r+1} = y$. If any of the *horizontal* edges

$$(z_0, z_1), (z_1, z_2), \dots, (z_{r-1}, z_r), (z_r, z_{r+1})$$

are in $\mathbf{E}(H)$, FOURCYCLE colors them now also. The coloring function Γ is extended as follows (a horizontal edge (z_i, z_{i+1}) is colored only if present, of course):

$$\begin{aligned} \Gamma(u, z_i) &= \text{red} \\ \Gamma(v, z_i) &= \text{blue} \\ \Gamma(z_0, z_1) &= \text{blue} \\ \Gamma(z_i, z_{i+1}) &= \text{red} \end{aligned}$$

where $1 \leq i \leq r$. See Figure 7. The diagonal 2-paths, together with any horizontal edges, partition the interior of the 4-cycle x, u, y, v into 3-cycles and separating 4-cycles. The flexibility that FOURCYCLE has is to assign the open edges for these 3-cycles and 4-cycles. The general assignment strategy is that (u, z_i) is open inside the separating cycle on its left if (z_{i-1}, z_i) is absent and (v, z_i) is open inside the separating cycle on its right if $i \neq 0$. More specifically, if (z_0, z_1) is present, then z_0, z_1, v is a **blue** cycle, and $\text{Open}((z_0, z_1); z_0, z_1, v)$; otherwise, $\text{Open}((v, z_0), (u, z_1); z_0, u, z_1, v)$. If (z_i, z_{i+1}) is present and $i > 0$, then z_{i+1}, z_i, u is a **red** cycle, and $\text{Open}((z_{i+1}, z_i); z_{i+1}, z_i, u)$; if (z_0, z_1) is present, then, since the edge (u, z_1) is **red**, $\text{Open}((z_1, z_0); z_1, z_0, v)$.

Now we give recursive calls for three subcases, which depend on the presence or absence of specific horizontal edges. Verification of the inductive hypotheses for each call is straightforward and is left to the reader.

Subcase 1: (z_i, z_{i+1}) , $0 \leq i \leq r$, is absent. The recursive call

FOURCYCLE (z_i, u, z_{i+1}, v)

completes the coloring of edges interior to the cycle z_i, u, z_{i+1}, v .

Subcase 2: (z_0, z_1) is present. The recursive calls

THREECYCLE (u, z_1, z_0)

THREECYCLE (z_0, z_1, v)

complete the coloring of edges interior to the cycles u, z_1, z_0 and z_0, z_1, v .

Subcase 3: (z_i, z_{i+1}) , $1 \leq i \leq r$, is present. The recursive calls

THREECYCLE (z_{i+1}, z_i, u)

THREECYCLE (v, z_i, z_{i+1})

complete the coloring of edges interior to the cycles u, z_{i+1}, z_i and z_i, z_{i+1}, v .

3.3.2 No Diagonal 2-paths

In the second main case, there are no diagonal 2-paths. Let the shortest diagonal path from u to v be a k -path, $k > 2$. Consider all diagonal k -paths from u to v (see Figure 8 where $k = 4$). Let W be the set of all vertices on these diagonal k -paths, including u and v . Let L , the set of all *level edges*, be the set of all edges on these diagonal k -paths. There is a well-defined distance function

$$d_u : W \rightarrow \{0, 1, \dots, k\}$$

where $d_u(w)$ is the distance from u to w along any diagonal k -path containing w . Partition W into $k + 1$ subsets

$$W_i = \{w \mid d_u(w) = i\}$$

where $0 \leq i \leq k$. Note that the planar embedding of H induces a left-to-right order $w_i[1], w_i[2], \dots, w_i[r_i]$ on each W_i , where $r_i = |W_i|$. By defining

$$\lambda_i = w_i[1]$$

where $0 \leq i \leq k$, we establish more convenient notation for specifying vertices and edges on the leftmost diagonal k -path. Similarly, by defining

$$\rho_i = w_i[r_i]$$

where $0 \leq i \leq k$, we establish more convenient notation for specifying vertices and edges on the rightmost diagonal k -path. Partition L into k levels

$$L_i = \{(w, z) \mid w \in W_{i-1}, z \in W_i\}$$

where $1 \leq i \leq k$. Let P be the set of *horizontal edges*:

$$P = \{(w, z) \mid w, z \in W, d_u(w) = d_u(z)\}.$$

Define $k - 1$ disjoint subsets of P which cover P :

$$P_i = \{(w_i[j], w_i[j + 1]) \in \mathbf{E}(H) \mid 1 \leq j \leq r_i - 1\}$$

where $1 \leq i \leq k - 1$. P_i is therefore the set of horizontal edges in level i . Finally, let

$$\begin{aligned} S_{\text{left}} &= \{(x, \lambda_i) \in \mathbf{E}(H) \mid 1 \leq i \leq k - 1\} \\ S_{\text{right}} &= \{(\rho_i, y) \in \mathbf{E}(H) \mid 1 \leq i \leq k - 1\} \\ S &= S_{\text{left}} \cup S_{\text{right}} \end{aligned}$$

be the *slanting edges*.

The set of edges that FOURCYCLE colors before making recursive calls is

$$T = L \cup P \cup S.$$

An alternate definition for T is the set of edges interior to the cycle x, u, y, v induced by the vertex set $W \cup \{x, u, y, v\}$. The edges of the original 4-cycle x, u, y, v together with T partition the interior into 3-cycles, separating 4-cycles, and halls. The uncolored interior edges are then colored recursively.

The coloring for the case k even is a direct generalization of the coloring for the diagonal 2-path case. Let k be even. Extend the coloring function as follows:

$$\begin{aligned} \Gamma(w_{i-1}[p], w_i[q]) &= \begin{cases} \text{red} & \text{if } i \text{ is odd and } (w_{i-1}[p], w_i[q]) \in L_i \\ \text{blue} & \text{if } i \text{ is even and } (w_{i-1}[p], w_i[q]) \in L_i \end{cases} \\ \Gamma(w_i[j], w_i[j+1]) &= \begin{cases} \text{red} & \text{if } i \text{ is odd and } (w_i[j], w_i[j+1]) \in P_i \\ \text{blue} & \text{if } i \text{ is even and } (w_i[j], w_i[j+1]) \in P_i \end{cases} \\ \Gamma(x, \lambda_i) &= \text{blue} \quad \text{for } (x, \lambda_i) \in S_{\text{left}} \\ \Gamma(\rho_i, y) &= \text{red} \quad \text{for } (\rho_i, y) \in S_{\text{right}} \end{aligned}$$

See Figure 9, where $k = 4$. As alternate levels are colored **red** and **blue**, this coloring strategy is called *alternate red/blue coloring*.

If $(\rho_1, y) \in S_{\text{right}}$, then the openness of (u, y) inside the cycle x, u, y, v justifies the creation of the **red** path u, ρ_1, y and hence of the **red** cycle u, y, ρ_1 . Similarly, if $(x, \lambda_{k-1}) \in S_{\text{left}}$, then the openness of (v, x) inside the cycle x, u, y, v justifies the creation of the **blue** path v, λ_{k-1}, x and hence of the **blue** cycle v, x, λ_{k-1} . In all other cases, it is evident that **blue**(H) and **red**(H) remain outerplanar. A way to visualize this is as follows. The edges of $L \cup P$ are alternately colored **blue** and **red** by levels, thus creating disconnected, horizontal **blue** and **red** outerplanar subgraphs, called *ropes*. Coloring the left slanting edges **blue** ties some of the **blue** ropes together at one end but cannot create an edge subdivision of $K_{2,3}$. Similarly, coloring the right slanting edges **red** ties some of the **red** ropes together at one end but cannot create an edge subdivision of $K_{2,3}$.

If k is odd, then alternate **red/blue** coloring must be modified so that L_1 is colored **red** and L_k is colored **blue**. This is accomplished by choosing an arbitrary intermediate

level L_i , $2 \leq i \leq k-1$, called the *distinguished level*, to color in a distinctive manner. For definiteness, choose L_{k-1} to be the distinguished level. The coloring function is first extended to all edges in $T - (L_{k-1} \cup P_{k-2} \cup P_{k-1})$ as follows:

$$\begin{aligned} \Gamma(w_{i-1}[p], w_i[q]) &= \begin{cases} \mathbf{red} & \text{if } i \text{ is odd, } i < k-1, \text{ and } (w_{i-1}[p], w_i[q]) \in L_i \\ \mathbf{blue} & \text{if } i \text{ is even, } i < k-1, \text{ and } (w_{i-1}[p], w_i[q]) \in L_i \end{cases} \\ \Gamma(w_i[j], w_i[j+1]) &= \begin{cases} \mathbf{red} & \text{if } i \text{ is odd, } i < k-2, \text{ and } (w_i[j], w_i[j+1]) \in P_i \\ \mathbf{blue} & \text{if } i \text{ is even, } i < k-2, \text{ and } (w_i[j], w_i[j+1]) \in P_i \end{cases} \\ \Gamma(w_{k-1}[p], v) &= \mathbf{blue} \quad \text{for } 1 \leq p \leq r_{k-1} \\ \Gamma(x, \lambda_i) &= \mathbf{blue} \quad \text{for } (x, \lambda_i) \in S_{\text{left}} \\ \Gamma(\rho_i, y) &= \mathbf{red} \quad \text{for } (\rho_i, y) \in S_{\text{right}} - \{(\rho_{k-1}, y)\} \\ \Gamma(\rho_{k-1}, y) &= \mathbf{blue} \quad \text{if } (\rho_{k-1}, y) \in S_{\text{right}} \end{aligned}$$

Observe that the coloring of (ρ_{k-1}, y) with **blue** instead of **red** is the rationale for

$$\text{Open}(y, v, x; x, u, y, v)$$

in IH 2.3.

It remains to color the graph

$$Z = (W_{k-2} \cup W_{k-1}, L_{k-1} \cup P_{k-2} \cup P_{k-1})$$

Z has the property that if the vertex sets W_{k-2} and W_{k-1} are placed on two horizontal lines in their normal left-to-right order, then all the edges of Z can be drawn as nonintersecting line segments. This drawing of Z induces a left-to-right order on L_{k-1} . Note that, except for slanting edges, any colored edges incident to W_{k-2} are **red** and any colored edges incident to W_{k-1} are **blue**.

The coloring of Z proceeds as follows. If $r_{k-1} = 1$, then

$$\begin{aligned} \Gamma(w_{k-2}[j], w_{k-2}[j+1]) &= \mathbf{red} \quad \text{for } (w_{k-2}[j], w_{k-2}[j+1]) \in P_{k-2} \\ \Gamma(w_{k-2}[1], \lambda_{k-1}) &= \mathbf{red} \\ \Gamma(w_{k-2}[j], \lambda_{k-1}) &= \mathbf{blue}, \end{aligned}$$

where $2 \leq j \leq r_{k-2}$. If $r_{k-1} > 1$ and $r_{k-2} = 1$, then

$$\begin{aligned}\Gamma(w_{k-1}[j], w_{k-1}[j+1]) &= \text{blue} \quad \text{for } (w_{k-1}[j], w_{k-1}[j+1]) \in P_{k-1} \\ \Gamma(\lambda_{k-2}, w_{k-1}[j]) &= \text{red},\end{aligned}$$

where $1 \leq j \leq r_{k-1}$. If $r_{k-1} > 1$ and $r_{k-2} > 1$, then the coloring proceeds left-to-right. Let the left-to-right order of edges in L_{k-1} be

$$(\lambda_{k-2}, \lambda_{k-1}) = e_1, e_2, \dots, e_{m-1}, e_m = (\rho_{k-2}, \rho_{k-1}).$$

Initially assign

$$\Gamma(\lambda_{k-2}, \lambda_{k-1}) = \text{red};$$

remaining edges in L_{k-1} are colored in the order e_2, \dots, e_m . There are three cases for the coloring of e_i , depending on whether e_{i-1} shares an endpoint with e_i . If e_{i-1} and e_i share a level $k-1$ endpoint, then assign $\Gamma(e_i) = \text{blue}$. If e_{i-1} and e_i share a level $k-2$ endpoint, then assign $\Gamma(e_i) = \text{red}$. Otherwise, $e_{i-1} = (w_{k-2}[p], w_{k-1}[q])$ and $e_i = (w_{k-2}[p+1], w_{k-1}[q+1])$; the color assignment for e_i is $\Gamma(e_i) = \overline{\Gamma(e_{i-1})}$. If $\Gamma(e_{i-1}) = \text{blue}$ and $(w_{k-2}[p], w_{k-2}[p+1]) \in P_{k-2}$, then assign $\Gamma(w_{k-2}[p], w_{k-2}[p+1]) = \text{blue}$. This assignment makes $(w_{k-2}[p], w_{k-2}[p+1])$ a cut edge. If $\Gamma(e_{i-1}) = \text{red}$ and $(w_{k-1}[q], w_{k-1}[q+1]) \in P_{k-1}$, then assign $\Gamma(w_{k-1}[q], w_{k-1}[q+1]) = \text{red}$. This assignment makes $(w_{k-1}[q], w_{k-1}[q+1])$ a cut edge. Any uncolored edges in P_{k-2} are assigned **red**, and any uncolored edges in P_{k-1} are assigned **blue**. Clearly, this coloring can be accomplished in linear time. The reader may easily verify the following claim:

Claim 4 *Each e_i , $1 \leq i \leq m$, is a cut edge.*

Thus $\text{blue}(H)$ and $\text{red}(H)$ remain outerplanar. Note that this coloring guarantees that $\Gamma(\lambda_{k-2}, \lambda_{k-1}) = \text{red}$.

The currently uncolored interior edges are colored by recursive calls to **THREECYCLE**, **FOURCYCLE**, and **HALL**. The remaining flexibility is the assignment of open edges to the

separating cycles. For clarity of exposition, we assign open edges in two phases. Starting with none of the edges of T having been colored, the first phase colors the edges of $L \cup P$, identifies the open edges for those separating cycles consisting only of those edges, and performs the requisite recursive calls. The second phase colors the edges of S , identifies the open edges for the separating cycles that contain one or more edges of S , and performs the requisite recursive calls.

Phase 1

In Phase 1, color only the edges in $L \cup P$ according to the previously mentioned extension for Γ . Note that any monochromatic cycle of edges from $L \cup P$ must be a 3-cycle. Any colored edge that is not in a monochromatic 3-cycle is necessarily a cut edge. Now consider in turn the 3-cycles, separating 4-cycles, and larger separating cycles of edges from $L \cup P$.

3-cycles. Every 3-cycle contains one horizontal edge and two level edges. Let

$$A = w_i[j], w_i[j + 1], w_n[p],$$

$|n - i| = 1$, be an arbitrary 3-cycle. To pass this cycle to THREECYCLE, at least one of its three edges must be open inside A . If A is monochromatic, then FOURCYCLE chooses the horizontal edge to be open:

$$\text{Open}((w_i[j], w_i[j + 1]); A).$$

If A is not monochromatic, then FOURCYCLE chooses

$$\text{Open}((w_i[j], w_n[p]); A),$$

since $(w_i[j], w_n[p])$ is a cut edge.

4-cycles. There are two kinds of separating 4-cycle possible, one of which looks like a diamond (consisting of four level edges) and the other of which looks like a square (consisting of two level edges and two horizontal edges). In the diamond case, the 4-cycle consists of four level edges on two consecutive levels. The 4-cycle has the form

$$A = w_{i-1}[p], w_{i-2}[q], w_{i-1}[p + 1], w_i[q'],$$

where $(w_{i-1}[p], w_{i-1}[p+1]) \notin P_{i-1}$. To pass this cycle to FOURCYCLE, two opposite edges in the cycle must be oppositely colored and open. By alternate **red/blue** coloring, at least one pair of opposite edges must have opposite colors. This is true even if two of the edges are in the distinguished level, as no monochromatic separating 4-cycles are ever created in $L \cup P$. In addition, $\text{Open}(w_{i-1}[p], w_{i-2}[q], w_{i-1}[p+1]; A)$, so IH 2.3 is satisfied.

In the square case, the 4-cycle consists of two level edges and two horizontal edges. The 4-cycle has the form

$$A = w_{i-1}[p], w_{i-1}[p+1], w_i[q+1], w_i[q],$$

where $(w_{i-1}[p], w_i[q+1]) \notin L_i$ and $(w_{i-1}[p+1], w_i[q]) \notin L_i$. Of necessity, the colors of the horizontal edges satisfy

$$\Gamma(w_{i-1}[p], w_{i-1}[p+1]) = \overline{\Gamma(w_i[q], w_i[q+1])}.$$

If both horizontal edges are cut edges, then A satisfies IH 2.3, and FOURCYCLE calls FOURCYCLE(A) recursively. The colors of the level edges satisfy

$$\Gamma(w_{i-1}[p], w_i[q]) = \overline{\Gamma(w_{i-1}[p+1], w_i[q+1])},$$

if and only if L_i is the distinguished level; in this case,

$$\text{Open}((w_{i-1}[p], w_i[q]), (w_{i-1}[p+1], w_i[q+1]); A),$$

IH 2.3 is satisfied, and FOURCYCLE calls FOURCYCLE(A) recursively.

If L_i is not the distinguished level, then the edge colors satisfy

$$\begin{aligned} \overline{\Gamma(w_{i-1}[p], w_{i-1}[p+1])} &= \Gamma(w_{i-1}[p], w_i[q]) \\ &= \Gamma(w_{i-1}[p+1], w_i[q+1]) \\ &= \Gamma(w_i[q], w_i[q+1]), \end{aligned}$$

and $(w_{i-1}[p], w_{i-1}[p+1])$ is in a monochromatic 3-cycle and cannot be open inside A . Therefore, it is not possible for A to satisfy IH 2.3. If A has no diagonal 2-paths, then the

recursive call $\text{HALL}(A)$ successfully colors its interior edges. In the case that A has one or more diagonal 2-paths, a slight modification of the original diagonal 2-path coloring for FOURCYCLE (Subsection 3.3.1) suffices to color the interior edges of A . By symmetry, we may assume that one or more diagonal 2-paths connect $w_{i-1}[p]$ and $w_i[q+1]$ and that $\Gamma(w_i[q], w_i[q+1]) = \mathbf{blue}$. Then, in the terminology of Subsection 3.3.1,

$$\begin{aligned} u &= w_{i-1}[p] \\ v &= w_i[q+1] \\ z_0 &= w_i[q] \\ z_{r+1} &= w_{i-1}[p+1]. \end{aligned}$$

Since (v, z_0) is a **blue** cut edge and $\text{Open}((z_{r+1}, v); A)$, we are justified in assigning $\Gamma(z_r, z_{r+1}) = \mathbf{blue}$, while keeping the remaining color assignments of the argument in Subsection 3.3.1. (This is, in essence, treating the *path* z_{r+1}, v, z_0 as being open.)

Larger cycles. Any remaining cycle A formed from edges in $L \cup P$ contains an even number of level edges greater than 4 (forming left and right “sides”), together with zero, one, or two horizontal edges. If a level edge is a cut edge, then it may be open inside A . Otherwise the level edge is a bounding edge in a monochromatic 3-cycle; since it has not been closed to be open inside the monochromatic 3-cycle, the edge is available to be open inside A . If there are two horizontal edges in A , then one of these edges may not be open inside A , but only one of them needs to be open to satisfy IH 3.2.

Phase 2

Once all the cycles of $L \cup P$ have been processed, the status of the edges in the leftmost and rightmost interior path remains relevant to the completion of the coloring of x, u, y, v . By left/right symmetry, it suffices to consider the status of an edge $e = (\lambda_{i-1}, \lambda_i)$ on the left. If e is a cut edge, then it is open inside the separating cycle on its left. Otherwise, e is a bounding edge in a monochromatic 3-cycle, and, since e is a level edge, it is not required to be open inside the monochromatic 3-cycle on its right, so it may be open inside the

separating cycle on its left. This means that now, in Phase 2, when FOURCYCLE colors the slanting edges, the remaining cycles formed (by edges from $L \cup S$) have all their level edges open. This is fortunate, for it is possible that, for example, the coloring of (x, λ_{i-1}) and (x, λ_i) may create a monochromatic 3-cycle. We now consider, in turn, the 3-cycles, 4-cycles, and larger cycles formed by edges from $L \cup S \cup \{(x, u), (u, y), (y, v), (v, x)\}$.

3-cycles. Suppose that k is even, i.e., there is no distinguished level. First consider a 3-cycle on the left $A = x, \lambda_{i-1}, \lambda_i$. By the coloring of S_{left} ,

$$\Gamma(x, \lambda_{i-1}) = \Gamma(x, \lambda_i) = \mathbf{blue}.$$

If $\Gamma(\lambda_{i-1}, \lambda_i) = \mathbf{red}$, then $\text{Open}((\lambda_{i-1}, \lambda_i); A)$. Otherwise, A is monochromatic, and $i \geq 2$. But now $\Gamma(\lambda_{i-2}, \lambda_{i-1}) = \mathbf{red}$, so $\text{Open}((x, \lambda_{i-1}); A)$. Second consider a 3-cycle on the right $A = y, \rho_i, \rho_{i-1}$. By the coloring of S_{right} ,

$$\Gamma(y, \rho_{i-1}) = \Gamma(y, \rho_i) = \mathbf{red}.$$

If $\Gamma(\rho_{i-1}, \rho_i) = \mathbf{blue}$, then $\text{Open}((\rho_{i-1}, \rho_i); A)$. Otherwise, A is monochromatic, and $i \leq k - 1$. But now $\Gamma(\rho_i, \rho_i) = \mathbf{blue}$, so $\text{Open}((y, \rho_i); A)$.

Now suppose that k is odd, i.e., L_{k-1} is distinguished. The arguments in the preceding paragraph still hold, except in the 3-cycle y, v, ρ_{k-1} , $\Gamma(\rho_{k-1}, y) = \mathbf{blue}$, and the edge (ρ_{k-1}, y) is available to be open inside the 3-cycle.

4-cycles. Consider a 4-cycle on the left side. It has the form $A = x, \lambda_i, \lambda_{i+1}, \lambda_{i+2}$, $0 \leq i \leq k - 2$. By construction, there is no diagonal 2-path between λ_i and λ_{i+2} . FOURCYCLE calls $\text{HALL}(A)$ with top λ_i and bottom λ_{i+2} ; we must justify the openness of two opposite edges of A . If $i = 0$, then $\Gamma(\lambda_i, \lambda_{i+1}) = \mathbf{red}$ and $\text{Open}((\lambda_i, \lambda_{i+1}); A)$. If, in addition, $k \geq 4$, then $\text{Open}((x, \lambda_{i+2}); A)$, since $\Gamma(x, \lambda_{i+2}) = \mathbf{blue}$ and $\Gamma(\lambda_{i+2}, \lambda_{i+3}) = \mathbf{red}$. Now suppose that $i > 0$. Then $\text{Open}((\lambda_{i+1}, \lambda_{i+2}); A)$ and $\text{Open}((x, \lambda_i); A)$ and $\text{Open}((x, \lambda_{i+1}); A)$.

Larger cycles. This is much like the case for 4-cycles. A k -cycle, $k > 4$, has the form

$$A = x, \lambda_i, \lambda_{i+1}, \dots, \lambda_{i+k-2}.$$

FOURCYCLE calls HALL(A) with top λ_i and bottom λ_{i+k-2} . The justification follows the same pattern as that for the 4-cycle case, except only one edge on the left wall may be closed.

This completes the strategy for assigning open edges.

Verification of the inductive hypotheses for THREECYCLE, FOURCYCLE, and HALL (see Subsection 3.4) is straightforward and left to the reader.

For example, Figure 10 shows a 6-cycle with vertex u as its upper end and vertex z as its lower end.

3.4 HALL

The call HALL($u_1, u_2, \dots, u_s; v_1, v_2, \dots, v_t$) passes the hall M with left wall u_1, u_2, \dots, u_s and right wall v_1, v_2, \dots, v_t , together with its interior edges, to HALL. The inductive hypotheses for HALL are a generalization of those for FOURCYCLE. The additional complexity of these inductive hypotheses constitute the requirements for the successful coloring of the slanting edges, as given later. The inductive hypotheses for HALL are:

- IH 3.1 All edges of M are colored;
- IH 3.2 All edges of M are open except perhaps (u_0, u_1) , and the 2-path through the bottom is also open: $\text{Open}(u_{s-1}, v, v_{t-1}; M)$;
- IH 3.3 The shortest diagonal path from u to v has length $\geq \max\{s, t, 3\}$;
- IH 3.4 No more than two consecutive edges of the left wall or of the right wall are the same color;
- IH 3.5 The two left vertices u_i and u_j , $1 \leq i < j \leq s$, are not connected by a **red** path, unless $j = i + 1$ and $\Gamma(u_i, u_j) = \mathbf{red}$, or $j = i + 2$ and $\Gamma(u_i, u_{i+1}) = \Gamma(u_{i+1}, u_j) = \mathbf{red}$;

- IH 3.6 The two right vertices v_i and v_j , $1 \leq i < j \leq t$, are not connected by a **blue** path, unless $j = i + 1$ and $\Gamma(u_i, u_j) = \mathbf{blue}$, or $j = i + 2$ and $\Gamma(u_i, u_{i+1}) = \Gamma(u_{i+1}, u_j) = \mathbf{blue}$;
- IH 3.7 For all i, j , $1 \leq i < j \leq s$, there is no path interior to M from u_i to u_j of length $\leq j - i$;
- IH 3.8 For all i, j , $1 \leq i < j \leq t$, there is no path interior to M from v_i to v_j of length $\leq j - i$.

HALL is the recursive procedure to color the edges interior to a hall. Note that the hall in Figure 11 has two open edges. This is true whenever HALL is called, though it is not guaranteed that the two open edges are of different colors. Also, HALL assumes that the edges of its left wall and the edges of its right wall are alternate **red/blue** colored.

Let the shortest diagonal path from u to v be a k -path, $k > 2$. Consider all diagonal k -paths from u to v . Let W be the set of all vertices on these diagonal k -paths, including u and v . Let L be the set of all diagonal edges on these diagonal k -paths. There is a well-defined distance function

$$d_u : W \rightarrow \{0, 1, \dots, k\}$$

where $d_u(w)$ is the distance from u to w along any diagonal k -path containing w . Partition W into $k + 1$ subsets

$$W_i = \{w \mid d_u(w) = i\}$$

where $0 \leq i \leq k$. Note that the planar embedding of H induces a left-to-right order $w_i[1], w_i[2], \dots, w_i[r_i]$ on each W_i , where $r_i = |W_i|$. By defining

$$\lambda_i = w_i[1]$$

where $0 \leq i \leq k$, we establish more convenient notation for specifying vertices and edges on the leftmost diagonal k -path. Similarly, by defining

$$\rho_i = w_i[r_i]$$

where $0 \leq i \leq k$, we establish more convenient notation for specifying vertices and edges on the rightmost diagonal k -path. Partition L into k levels

$$L_i = \{(w, z) \mid w \in W_{i-1}, z \in W_i\}$$

where $1 \leq i \leq k$. Let P be the set of *horizontal edges*:

$$P = \{(w, z) \mid w, z \in W, d_u(w) = d_u(z)\}.$$

Define $k - 1$ disjoint subsets of P which cover P :

$$P_i = \{(w_i[j], w_i[j + 1]) \in \mathbf{E}(H) \mid 1 \leq j \leq r_i - 1\}$$

where $1 \leq i \leq k - 1$. Finally, let

$$S_{\text{left}} = \{(u_j, \lambda_i) \in \mathbf{E}(H) \mid 1 \leq i \leq k - 1\}$$

$$S_{\text{right}} = \{(\rho_i, v_j) \in \mathbf{E}(H) \mid 1 \leq i \leq k - 1\}$$

$$S = S_{\text{left}} \cup S_{\text{right}}$$

be the *slanting edges*. By IH 3.8, no λ_i is incident to more than two edges in S_{left} ; furthermore, if $(u_m, \lambda_i), (u_n, \lambda_i) \in S_{\text{left}}$ are distinct edges, then $|m - n| = 1$. A similar statement holds regarding ρ_i and S_{right} .

HALL proceeds in much the same way as FOURCYCLE. If $\Gamma(v_0, v_1) = \overline{\Gamma(u_{s-1}, u_s)}$, then the coloring of the level and horizontal edges is identical to that of FOURCYCLE, as are the recursive calls for colored cycles of edges from $L \cup P$. If $\Gamma(v_0, v_1) = \Gamma(u_{s-1}, u_s)$, then, without loss of generality, assume that $\Gamma(v_0, v_1) = \mathbf{red}$. Because (u_0, u_1) and (v_{t-1}, v_t) may be closed, we need L_1 and L_k both colored **red**. This aim can be accomplished with alternate **blue/red** coloring, without a distinguished level, if k is odd. If k is even, the aim is accomplished by choosing a distinguished level to color using the same algorithm as in Section 3.3.2. For definiteness, we choose L_{k-1} , as before, to be the distinguished level.

The interesting problem is coloring the slanting edges from the left and right walls to the outermost diagonal paths. Following the strategy of FOURCYCLE, HALL ties **blue** ropes only on the left, and ties **red** ropes only on the right.

Consider first the left slanting edges. Assign colors

$$\begin{aligned}\Gamma(u_1, \lambda_1) &= \text{blue} && \text{if } (u_1, \lambda_1) \in S_{\text{left}} \\ \Gamma(u_{s-1}, \lambda_{k-1}) &= \text{red} && \text{if } (u_{s-1}, \lambda_{k-1}) \in S_{\text{left}}.\end{aligned}$$

Order the remaining edges in S_{left} from top to bottom e_1, e_2, \dots, e_n . Color these edges as follows. First,

$$\Gamma(e_1) = \begin{cases} \text{red} & \text{if } (u_1, \lambda_1) \in S_{\text{left}} \text{ and } \lambda_1 \in e_1 \\ \text{blue} & \text{otherwise.} \end{cases}$$

For the remaining edges $e_i, i > 1$,

$$\Gamma(e_i) = \begin{cases} \text{blue} & \text{if } |e_{i-1} \cap e_i| = 0 \\ \text{red} & \text{if } |e_{i-1} \cap e_i| = 1 \text{ and } e_{i-1} \cap e_i \subseteq S_{\text{left}} \\ \text{blue} & \text{otherwise.} \end{cases}$$

Once the edges are colored, the recursive calls that complete the coloring for HALL are analogous to those for FOURCYCLE.

As an example, consider Figure 11, where there is a single diagonal 7-path inside a hall, and the slanting edges triangulate the interior of the hall. The general rule is that the edges from the left wall to the diagonal path should be **blue** (and those from the right wall should be **red**). The exception is when two **blue** ropes would be tied from the right. In Figure 12, such a tie is prevented by coloring one left slanting edge **red**. As a side-effect, this **red** edge connects a **red** rope on the left of the hall to one on the right of the hall; however, no unwanted tie (i.e., one on the left) is created in the **red** subgraph. Similarly, a single right slanting edge is colored **blue** in such a way that a tie of two **red** ropes on the left is prevented. As a result, a **blue** rope on the left of the hall is connected to a **blue** rope on the right of the hall. However, no unwanted tie is created in the **blue** subgraph.

3.5 Time Complexity

The algorithm is easily shown to run in polynomial time. The time complexity is dominated by the task of finding shortest diagonal paths in FOURCYCLE and HALL. Frederickson

[13] gives an algorithm that solves the single-source shortest-path problem in a planar graph H in time $O(|\mathbf{V}(H)|\sqrt{\log |\mathbf{V}(H)|})$. Each call to FOURCYCLE or HALL invokes Frederickson's algorithm once and colors at least one previously uncolored edge. Since the number of edges in a planar graph H is linear in $|\mathbf{V}(H)|$, the time complexity of the algorithm is $O(|\mathbf{V}(H)|\sqrt{\log |\mathbf{V}(H)|})$. This completes the proof of Theorem 1.

4 Open Problems

As an edge coloring of an arbitrary planar graph with outerplanar subgraphs with a minimum number of colors can be found in polynomial time, the remaining algorithmic question is what is the best time complexity for minimal edge coloring. A method for avoiding the large number of all-pairs shortest-paths calculations in COLOREDGE is called for.

To the best of our knowledge, the COLOREDGE approach to decomposing a planar graph has not been used before. This decomposition or similar ones are likely to contribute to solving other edge coloring on planar graphs.

By counting edges, we can compare the result of Theorem 1 with the result of Nash-Williams [28, 29] and Tutte [31] mentioned in Section 1: Every planar graph can be edge colored with three acyclic subgraphs. A triangulated planar graph with $n \geq 3$ vertices has $3n - 6$ edges. A (maximal) acyclic graph with n vertices has $n - 1$ edges. Thus, an edge coloring by three acyclic subgraphs is only $3(n - 1) - (3n - 6) = 3$ edges shy of producing three maximal acyclic subgraphs. A maximal outerplanar graph with n vertices has $2n - 3$ edges. Thus, an edge coloring by two outerplanar subgraphs is a substantial $2(2n - 3) - (3n - 6) = n$ edges shy of producing two maximal outerplanar subgraphs. This suggest that there may be a subclass of outerplanar graphs with significantly fewer than $2n - 3$ edges which may edge color planar graphs with two colors. Or, since $(2n - 3) + (n - 1) > 3n - 6$, it may be possible to edge color a planar graph with one outerplanar subgraph and one acyclic subgraph.

References

- [1] F. BERMAN, D. JOHNSON, T. LEIGHTON, P. W. SHOR, AND L. SNYDER, *Generalized planar matching*, Journal of Algorithms, 11 (1990), pp. 153–184.
- [2] F. BERNHART AND B. KAINEN, *The book thickness of a graph*, Journal of Combinatorial Theory B, 27 (1979), pp. 320–331.
- [3] Y. CARO, *The decomposition of trees into subtrees*, Journal of Graph Theory, 8 (1984), pp. 471–479.
- [4] G. CHARTRAND, D. GELLER, AND S. HEDETNIEMI, *Graphs with forbidden subgraphs*, Journal of Combinatorial Theory B, 10 (1971), pp. 12–41.
- [5] G. CHARTRAND AND F. HARARY, *Planar permutation graphs*, Annales de l'Institut Henri Poincaré B, 3 (1967), pp. 433–438.
- [6] F. R. K. CHUNG, F. T. LEIGHTON, AND A. L. ROSENBERG, *Embedding graphs in books: A layout problem with applications to VLSI design*, SIAM Journal on Algebraic and Discrete Methods, 8 (1987), pp. 33–58.
- [7] V. CHVÁTAL AND P. L. HAMMER, *Aggregation of inequalities in integer programming*, Annals of Discrete Mathematics, 1 (1977), pp. 145–162.
- [8] C. J. COLBOURN, *The Combinatorics of Network Reliability*, Oxford University Press, 1987.
- [9] ———, *Edge-packings of graphs and network reliability*, Discrete Mathematics, 72 (1988), pp. 49–61.
- [10] D. G. CORNEIL, S. MASUYAMA, AND S. L. HAKIMI, *Edge-free packings of graphs*. Submitted for publication, 1991.

- [11] J. EDMONDS, *Minimum partition of a matroid into independent subsets*, Journal of Research of the NBS, 69B (1965), pp. 67–72.
- [12] E. S. ELMALLAH AND C. J. COLBOURN, *Partitioning the edges of a planar graph into two partial k -trees*, Congressus Numerantium, 66 (1988), pp. 69–80.
- [13] G. N. FREDERICKSON, *Fast algorithms for shortest paths in planar graphs, with applications*, SIAM Journal on Computing, 16 (1987), pp. 1004–1022.
- [14] H. N. GABOW AND O. KARIV, *Algorithms for edge coloring bipartite graphs*, in Proceedings of the 10th Annual ACM Symposium on Theory of Computing, 1978, pp. 184–192.
- [15] L. S. HEATH, *Embedding planar graphs in seven pages*, in Proceedings of the 25th Annual IEEE Symposium on Foundations of Computer Science, 1984, pp. 74–83.
- [16] —, *Edge coloring planar graphs with two outerplanar subgraphs*, in Proceedings of the Second Annual ACM-SIAM Symposium on Discrete Algorithms, 1991, pp. 195–202.
- [17] L. S. HEATH AND J. P. C. VERGARA, *Edge-packing in planar graphs*. In preparation, 1991.
- [18] P. HELL AND D. G. KIRKPATRICK, *On generalized matching problems*, Inf. Process. Lett., 12 (1981), pp. 33–35.
- [19] —, *Packings by cliques and by finite families of graphs*, Discrete Mathematics, 49 (1984), pp. 45–59.
- [20] —, *Packings by complete bipartite graphs*, SIAM Journal on Algebraic and Discrete Methods, 7 (1986), pp. 199–209.
- [21] I. HOLYER, *The NP-completeness of edge-coloring*, SIAM Journal on Computing, 10 (1981), pp. 718–720.

- [22] —, *The NP-completeness of some edge-partition problems*, SIAM Journal on Computing, 10 (1981), pp. 713–717.
- [23] M. JÜNGER, G. REINELT, AND W. R. PULLEYBLANK, *On partitioning the edges of graphs into connected subgraphs*, Journal of Graph Theory, 9 (1985), pp. 539–549.
- [24] H. J. KARLOFF AND D. B. SHMOYS, *Efficient parallel algorithms for edge coloring problems*, Journal of Algorithms, 8 (1987), pp. 39–52.
- [25] D. G. KIRKPATRICK AND P. HELL, *On the completeness of a generalized matching problem*, in Proceedings of the 10th Annual ACM Symposium on Theory of Computing, 1978, pp. 240–245.
- [26] —, *On the complexity of general graph factor problems*, SIAM Journal on Computing, 12 (1983), pp. 601–609.
- [27] S. MASUYAMA AND T. IBARAKI, *Chain packing in graphs*. To appear in *Algorithmica*, 1990.
- [28] C. S. A. NASH-WILLIAMS, *Edge-disjoint spanning trees of finite graphs*, Journal of the London Mathematical Society, 36 (1961), pp. 445–450.
- [29] —, *Decomposition of finite graphs into forests*, Journal of the London Mathematical Society, 39 (1964), p. 12.
- [30] W. SCHNYDER, *Embedding planar graphs on the grid*, in Proceedings of the First Annual ACM-SIAM Symposium on Discrete Algorithms, 1990, pp. 138–148.
- [31] W. T. TUTTE, *On the problem of decomposing a graph into n connected factors*, Journal of the London Mathematical Society, 36 (1961), pp. 221–230.
- [32] J. P. C. VERGARA AND L. S. HEATH, *Edge-packing by isomorphic subgraphs*, Tech. Report TR 91-3, Virginia Polytechnic Institute and State University, Department of Computer Science, 1991.

- [33] V. G. VIZING, *On an estimate of the chromatic class of a p -graph*, Diskret. Analiz., 3 (1964), pp. 25–30.
- [34] M. YANNAKAKIS, *Four pages are necessary and sufficient for planar graphs*, in Proceedings of the 18th Annual ACM Symposium on Theory of Computing, 1986, pp. 104–108.
- [35] H. P. YAP, *Packing of graphs—a survey*, Discrete Mathematics, 72 (1988), pp. 395–404.

List of Figures

1	K_4	34
2	K_4 colored.	34
3	3-cycle without K_4	34
4	Figure 3 colored.	34
5	4-cycle.	35
6	4-cycle with diagonal 2-paths.	35
7	Figure 6 colored.	35
8	4-cycle with diagonal 4-paths.	35
9	Figure 8 colored.	36
10	A hall.	36
11	A hall with diagonal.	36
12	Figure 11 colored.	36

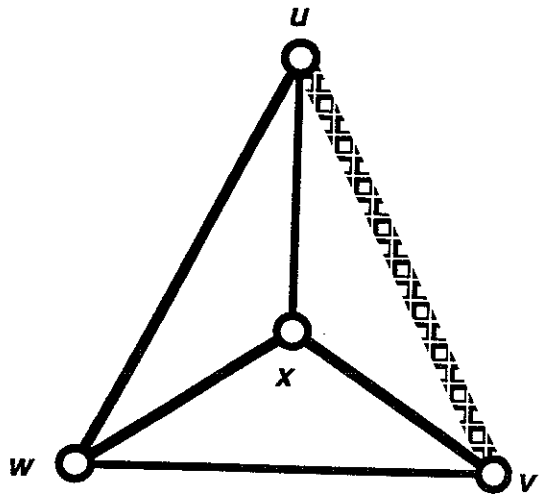


Figure 1.

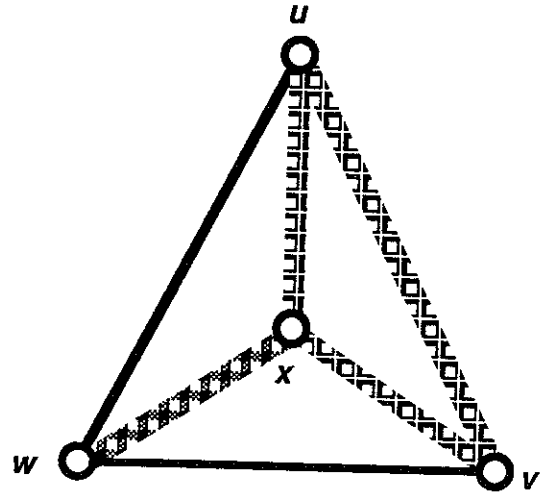


Figure 2.

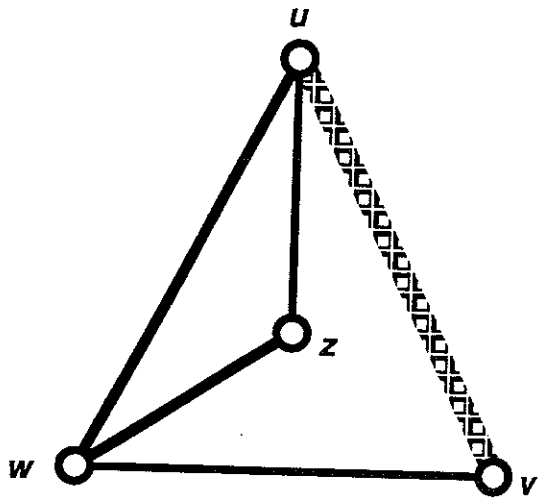


Figure 3.

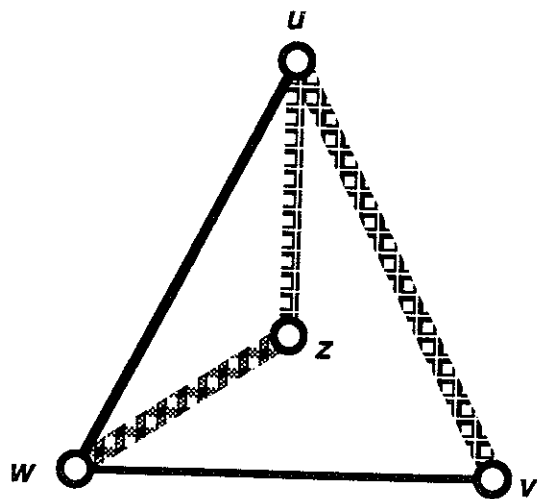


Figure 4.

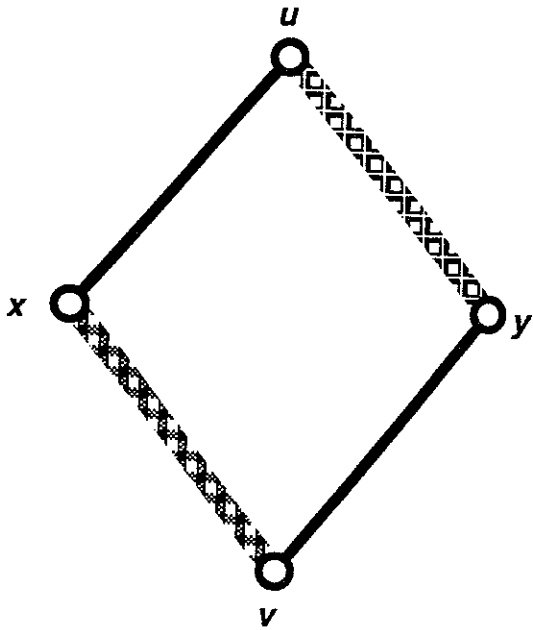


Figure 5.

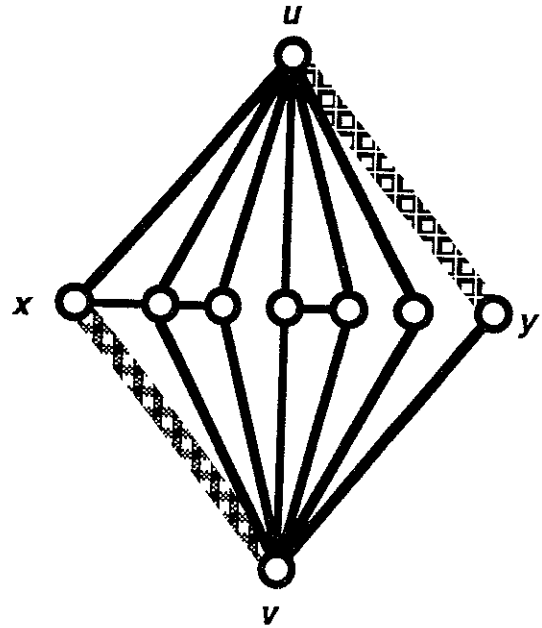


Figure 6.

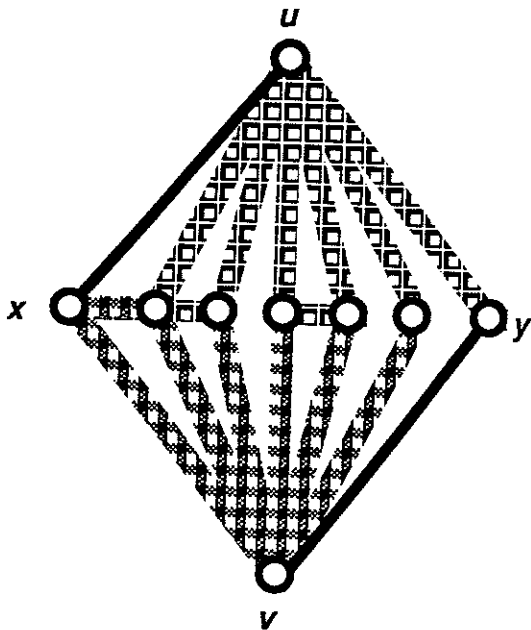


Figure 7.

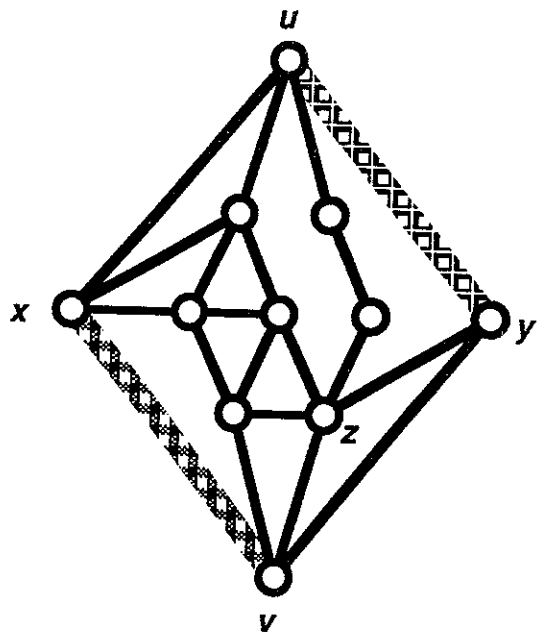


Figure 8.

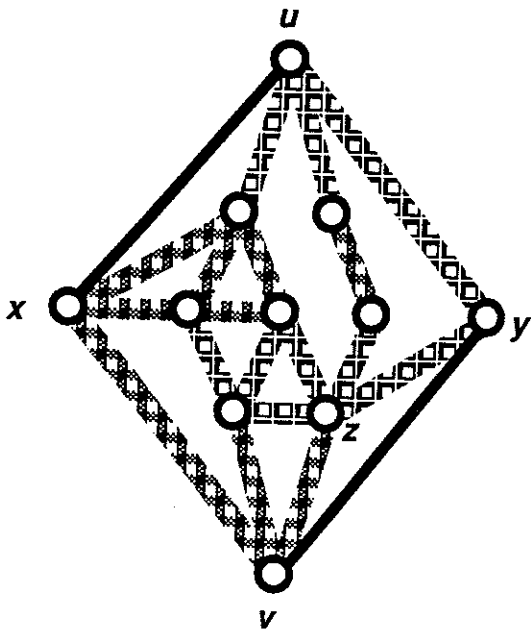


Figure 9.

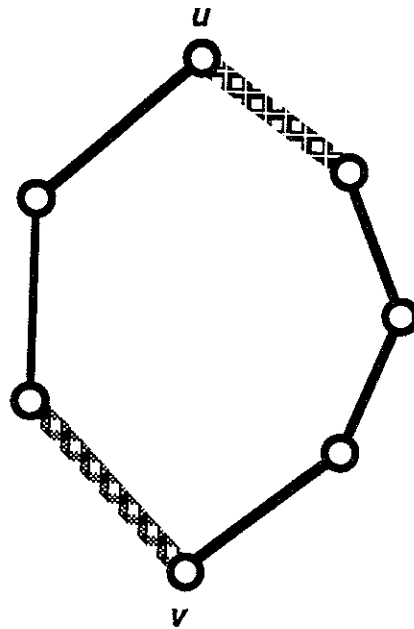


Figure 10.

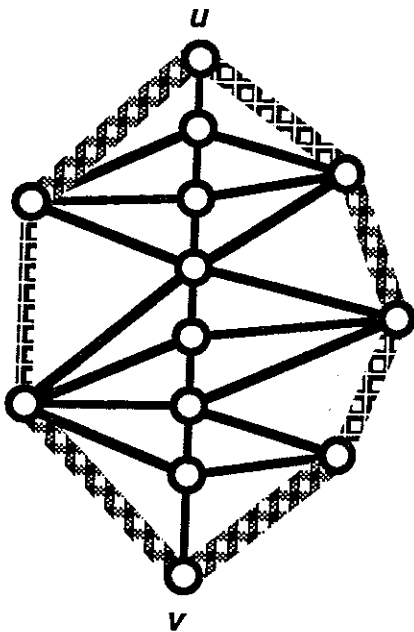


Figure 11.

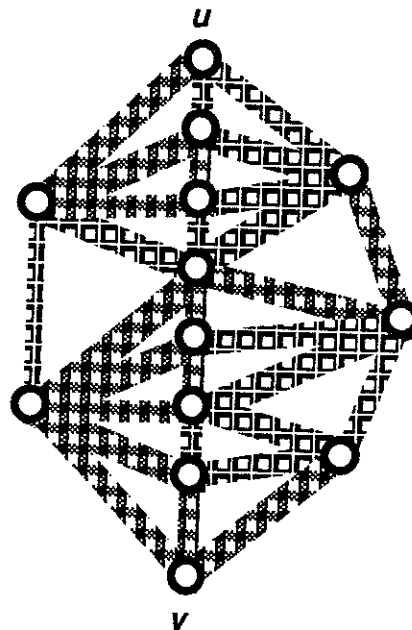


Figure 12.