Dual Eulerian Properties of Plane Multigraphs[1]

Bradley S. Carlson†, *C.Y. Roger Chen*‡ and *Dikran S. Meliksetian*§

†Department of Electrical Engineering
State University of New York at Stony Brook
Stony Brook, NY 11794-2350
Phone: (516)632-8474 or 8400
Fax: (516)632-8494
Internet: bcarlson@sbee.sunysb.edu

‡Department of Electrical and Computer Engineering
Syracuse University
Syracuse, NY 13244-1240
Phone: (315) 443-4301

§Department of Electrical Engineering
South Dakota School of Mines and Technology
Rapid City, SD 57701

Technical Report No. 645
College of Engineering and Applied Science
SUNY at Stony Brook
November 6, 1992

Abstract— A plane multigraph is said to be dual Eulerian if both itself and its dual contain an Euler path or circuit, and the Euler paths have corresponding edge sequences. In this paper several properties of plane multigraphs are derived, and a necessary and sufficient condition for a plane multigraph to be dual Eulerian is given. Although the necessary and sufficient condition for a multigraph to be Eulerian is somewhat trivial; the necessary and sufficient condition for a plane multigraph to be dual Eulerian is not. Nevertheless, the question of whether or not a plane multigraph is dual Eulerian can be answered in time proportional to a linear function of the number of edges of the graph, and an algorithm which answers this question is presented in this paper. This theory can be applied to the layout synthesis of functional cells for CMOS VLSI circuits.

[1]This research was supported in part by the National Science Foundation under grant number MIP-8909792.

I INTRODUCTION

The problem of identifying plane multigraphs which are dual Eulerian is investigated in this paper. A *dual Eulerian plane multigraph* is one in which both itself and its dual contain an Euler path or circuit, and the Euler paths have corresponding edge sequences. The problem is restricted to planar multigraphs, since these are the only graphs for which a dual is defined. Furthermore, it is assumed that a specific imbedding of a planar graph, called a *plane graph*, is given [1]. The problem of whether or not a planar graph admits an imbedding which is dual Eulerian is a different problem, and is not addressed in this paper.

This problem is of interest in the design of Complementary Metal-Oxide Semiconductor (CMOS) Very Large Scale Integrated (VLSI) circuits [2]. In VLSI design it is desirable to design the physical implementations of circuits such that they require a minimum amount of silicon area. When circuits are represented by undirected multigraphs, dual paths correspond to linear placements of transistors which require a minimum amount of silicon area [2].

This paper is organized as follows. The necessary mathematical definitions and some preliminary results are presented in the next section. The derivation of the algorithm and the main results are presented in Section III. The algorithm is specified in pseudo-code and analyzed in Section IV. An example illustrating the application of the algorithm is presented in Section V, and the paper is concluded in Section VI.

II DEFINITIONS AND PRELIMINARY RESULTS

$\Gamma(V, f, E)$ is an undirected multigraph with a set V of vertices, a set E of edges and a function $f : E \rightarrow P_2(V)$, where $P_2(V)$ are the subsets of V of size two [3]. Furthermore, the edges are labeled for convenience in representation and presentation. A *path* $p = v_0e_0v_1e_1v_2 \cdots v_{n-1}e_{n-1}v_n$ in Γ is a sequence of alternating vertices and edges which begins and ends at a vertex, $f(e_i) = \{v_i, v_{i+1}\}$ $(0 \leq i \leq n-1)$ and $e_i \neq e_j$ for $i \neq j$. The *length* of a path is the number of edges contained in it. A path is a *circuit* if $v_0 = v_n$. A multigraph Γ is said to be *Eulerian* if and only if it contains a path which contains every edge of Γ. A circuit is *elementary* if each vertex is distinct with the exception of the first coinciding with the last. The edge set of a circuit in Γ is a *cycle* [3]. An *elementary cycle* is the cycle of an elementary circuit. Γ is said to be *planar* if it can be drawn on the plane with edges meeting only at the vertices [3]. An imbedding of a planar multigraph on the plane is called a *plane* multigraph [1]. A *face* of a plane graph Γ is a domain of the plane surrounded by edges of Γ such that any two points in it can be joined by a line not crossing any edge [4]. A *facial circuit*

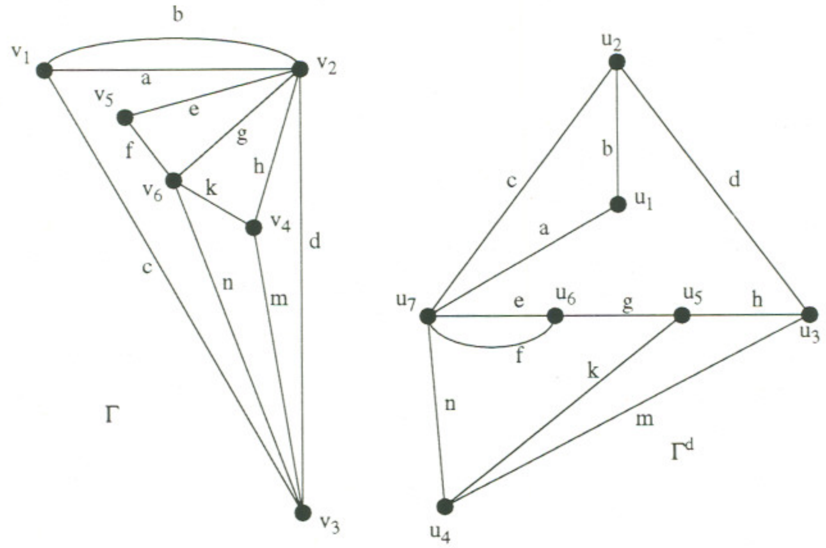

Fig. 1. A plane undirected multigraph Γ and its dual Γ^d.

is a circuit which forms the boundary of a face in Γ. A *facial cycle* is the cycle of a facial circuit. The dual multigraph Γ^d of a plane multigraph Γ can be constructed by placing a vertex in each closed face of Γ and one vertex in the infinite region (i.e., the region of the plane not surrounded by edges), and connecting two vertices, say v_1 and v_2, in Γ^d by an edge labeled e if and only if the edge labeled e in Γ is on the boundary of the faces of Γ corresponding to v_1 and v_2 of Γ^d. The edge sets of Γ and Γ^d are the same, and the vertices of Γ^d correspond to the faces of Γ and vice versa. A plane multigraph Γ and its dual Γ^d are shown in Fig. 1. Henceforth it is assumed that Γ is a plane undirected multigraph. We restrict ourselves to plane multigraphs, since their duals are unique. A path in Γ is said to be a *dual path* if there exists a path in Γ^d with the same edge sequence. For example, $v_2\, d\, v_3\, m\, v_4\, k\, v_6\, g\, v_2\, e\, v_5\, f\, v_6$ is a path in Γ of Fig. 1, and $u_2\, d\, u_3\, m\, u_4\, k\, u_5\, g\, u_6\, e\, u_7\, f\, u_6$ is a path in Γ^d. Since these paths have corresponding edge sequences, each one is a dual path. A dual path is a *dual circuit* if it is a circuit in both Γ and Γ^d. A path can be identified by its sequence of edges, and this alternative means of identification is used throughout the sequel whenever it is unambiguous or the ambiguity is unimportant. In Γ of Fig. 1 $a\,e\,f\,g$ is a path, but it is not a dual path since $a\,e\,f\,g$ is not a path in Γ^d. Similarly, $e\,a\,b\,c$ is a path in Γ^d, but is not a dual path since it is not a path in Γ. Γ is said to be *dual Eulerian* if and only if it contains a dual path which

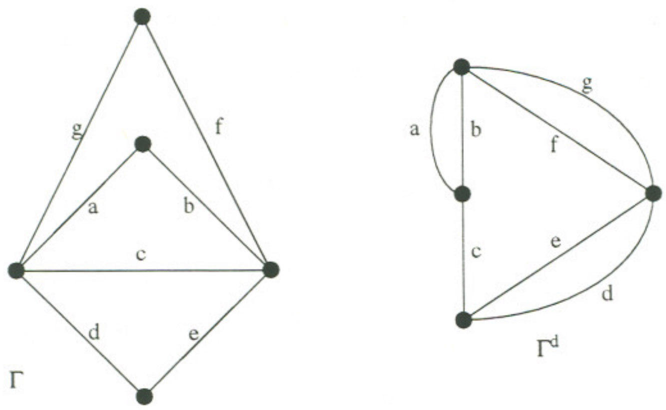

Fig. 2. Example illustrating that both Γ and Γ^d being Eulerian is not a sufficient condition for them to be dual Eulerian.

contains every edge. Of course if Γ is dual Eulerian, then so is Γ^d. An obvious necessary condition for Γ to be dual Eulerian is that both Γ and Γ^d be Eulerian[2]. A simple counter example shown in Fig. 2 illustrates that this is not a sufficient condition.

A *submultigraph* of a plane undirected multigraph $\Gamma(V, f, E)$ is a plane undirected multigraph $\Gamma'(V', f_{|E'}, E')$, where $V' \subseteq V$, $E' \subseteq E$ and $f_{|E'}$ is the function f restricted to the set E'. A submultigraph $\Gamma'(V', f_{|E'}, E')$ of $\Gamma(V, f, E)$ is a *component* of Γ if there does not exist $v \in V'$ and $e \in E \setminus E'$ such that $v \in f(e)$. A *cocycle* in Γ is a set of edges of Γ such that the removal of these edges from Γ increases the number of components of Γ. The set of edges incident on a vertex is a cocycle and is referred to as a *vertex cocycle*. If the vertex cocycle of $v \in V$ is denoted by $f^*(v)$, then

$$f^*(v) = \{e \in E | v \in f(e)\}[3].$$

An *isthmus* is an edge which when removed from Γ increases the number of components of Γ (i.e., it is a cocycle of size one). It is assumed that Γ does not contain any isthmuses, since if it did, its dual would not satisfy the definition of a graph [3]. That is, the dual of a graph with

[2] A well known necessary and sufficient condition for a multigraph to be Eulerian is that it have no more than two vertices with an odd degree [3].

isthmuses contains loops (i.e., an edge which is incident on the same vertex at both its ends), and loops contradict the definition of the function $f : E \rightarrow P_2(V)$.

Two edges e_i and e_j are said to be in *series* if $e_i = e_j$ or $\exists v_k \in V \cdot \ni \cdot f^*(v_k) = \{e_i, e_j\}$ or there exists a path $v_p\, e_p\, v_{p+1} \cdots v_{n-1}\, e_{n-1}\, v_n$ such that $v_i\, e_i v_p\, e_p\, v_{p+1} \cdots v_{n-1}\, e_{n-1}\, v_n\, e_j\, v_j$ or $v_j\, e_j v_p\, e_p\, v_{p+1} \cdots v_{n-1}\, e_{n-1}\, v_n\, e_i\, v_i$ is a path and $|f^*(v_q)| = 2, \forall q (p \leq q \leq n)$. Two edges e_i and e_j are said to be in *parallel* if $f(e_i) = f(e_j)$.

Lemma 1 *The series and parallel relations are equivalence relations.*

Proof: First consider the series relation.

1. It is reflexive by definition.
2. Assume that e_i and e_j are in series, and without loss of generality assume $e_i \neq e_j$. There are two cases which need to be considered.
   (a) $\exists v_k \in V \cdot \ni \cdot f^*(v_k) = \{e_i, e_j\}$. Since $\{e_i, e_j\} = \{e_j, e_i\}$, the relation is symmetric.
   (b) There exists a path from e_i to e_j. Since the path is bidirectional, the relation is symmetric.
3. Assume e_i and e_j are in series and e_j and e_k are in series. Assume without loss of generality that $f(e_i) = \{v_1, v_2\}$, $f(e_j) = \{v_1, v_3\}$ and $f(e_k) = \{v_3, v_4\}$. $v_2\, e_i\, v_1\, e_j\, v_3\, e_k\, v_4$ is a path and $|f^*(v_1)| = 2$ since e_i and e_j are in series and $|f^*(v_3)| = 2$ since e_j and e_k are in series. Hence, e_i and e_k are in series, and the relation is transitive. The same argument is true if there is a path between e_i and e_j and/or e_j and e_k.

Consider the parallel relation.

1. Clearly $f(e_i) = f(e_i)$. Hence, the relation is reflexive.
2. If $f(e_i) = f(e_j)$, then $f(e_j) = f(e_i)$. Hence, the relation is symmetric.
3. If $f(e_i) = f(e_j)$ and $f(e_j) = f(e_k)$, then $f(e_i) = f(e_k)$. Hence, the relation is transitive.

□

Let the equivalence classes induced by the series and parallel relations be known as the series and parallel sets of edges, respectively. A series (respectively, parallel) set of edges in Γ corresponds to a parallel (respectively, series) set of edges in Γ^d, and vice versa. Define the *series reduction operation* on a graph Γ as the operation of replacing an even (respectively, odd) series set of edges in Γ by two (respectively, one) series edges, and the *parallel reduction operation* as the operation of replacing an even (respectively, odd) parallel set of edges in Γ by two (respectively, one) parallel edges. Define the *reduced graph* Γ_r of a graph Γ as the graph obtained by recursively applying the reduction operation to Γ until it can no longer be applied. For example, the reduced graph

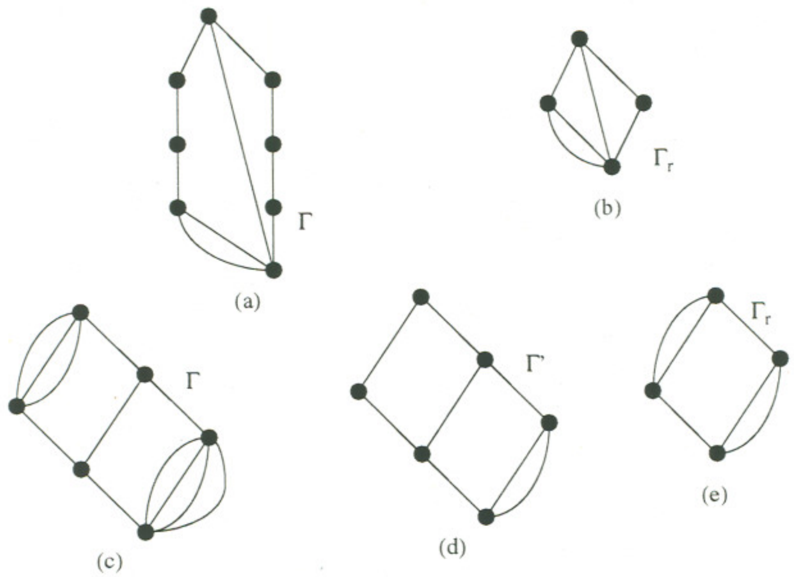

Fig. 3. Examples illustrating graph reduction.

Γ_r in Fig. 3(b) is obtained from Γ in Fig. 3(a) by applying the series reduction operation, and the graph Γ' in Fig. 3(d) is obtained from Γ in Fig. 3(c) by applying the parallel reduction operation. The series reduction operation can be applied to Γ' in Fig. 3(d) to obtain the reduced graph Γ_r in Fig. 3(e).

Theorem 2 *A graph Γ is dual Eulerian if and only if its reduced graph Γ_r is dual Eulerian.*

Proof: Assume Γ is dual Eulerian. The proof is by induction on the application of the reduction operators. Without loss of generality consider a series set of edges in Γ as shown in Fig 4(a). Γ^d has a corresponding parallel set of edges as shown in Fig 4(b). There are two possibilities that need to be considered for the basis case.

1. Without loss of generality assume the dual path $v_1\, e_1\, v_2\, e_2\, v_3\, \cdots\, v_k\, e_k\, v_{k+1}$ is contained in the dual Euler path of Γ. If k is even, then the corresponding dual path contained in the corresponding dual Euler path in Γ^d is either $u_1\, e_1\, u_2\, e_2\, u_1\, \cdots\, u_2\, e_k\, u_1$ or $u_2\, e_1\, u_1\, e_2\, u_2\, \cdots\, u_1\, e_k\, u_2$. Without loss of generality assume it is $u_1\, e_1\, u_2\, e_2\, u_1\, \cdots\, u_2\, e_k\, u_1$. If the even series (respectively, parallel) set of edges in Γ (respectively, Γ^d) is replaced by two edges, say e_x and e_y, in series (respectively, parallel) to obtain Γ_1 (respectively, Γ_1^d), then the dual path in Γ_1 (respectively, Γ_1^d) is $v_1\, e_x\, v_2\, e_y\, v_{k+1}$ (respectively, $u_1\, e_x\, u_2\, e_y\, u_1$). If Γ is dual Eulerian, then Γ_1 is dual

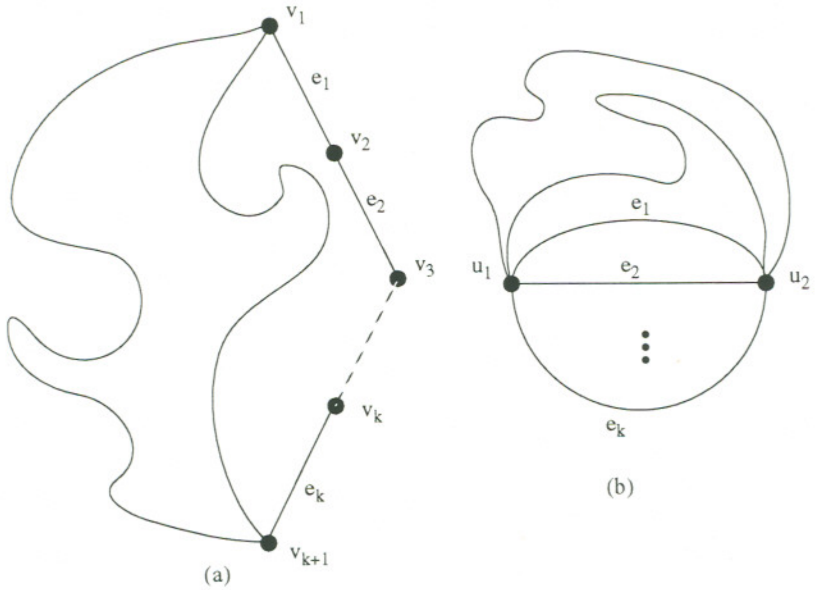

Fig. 4. Dual plane multigraphs with general series and parallel sets of edges.

Eulerian. If k is odd, then the corresponding dual path contained in the corresponding dual Euler path in Γ^d is either $u_1\, e_1\, u_2\, e_2\, u_1 \,\cdots\, u_1\, e_k\, u_2$ or $u_2\, e_1\, u_1\, e_2\, u_2 \,\cdots\, u_2\, e_k\, u_1$. Without loss of generality assume it is $u_1\, e_1\, u_2\, e_2\, u_1 \,\cdots\, u_1\, e_k\, u_2$. If the odd series (respectively, parallel) set of edges in Γ (respectively, Γ^d) is replaced by a single edge, say e_z, to obtain Γ_1 (respectively, Γ_1^d), then the dual path in Γ_1 (respectively, Γ_1^d) is $v_1\, e_z\, v_{k+1}$ (respectively, $u_1\, e_z\, u_2$). If Γ is dual Eulerian, then Γ_1 is dual Eulerian.

2. The dual Euler path in Γ begins at some vertex v_i $(2 \leq i \leq k)$. Without loss of generality assume the dual Euler path is $v_i\, e_{i+1}\, v_{i+1} \cdots v_k\, e_k\, v_{k+1} \cdots v_1\, e_1\, v_2 \cdots v_{i-1}\, e_i\, v_i$. If k is even, then without loss of generality assume the dual Euler path in Γ^d is $u_1\, e_{i+1}\, u_2 \,\cdots\, u_1\, e_k\, u_2 \,\cdots\, u_2\, e_1\, u_1 \,\cdots\, u_2\, e_i\, u_1$. If the even series (respectively, parallel) set of edges in Γ (respectively, Γ^d) is replaced by two edges, say e_x and e_y, in series (respectively, parallel) to obtain Γ_1 (respectively, Γ_1^d), then the dual Euler path in Γ_1 (respectively, Γ_1^d) is $v_2\, e_y\, v_3 \,\cdots\, v_1\, e_x\, v_2$ (respectively, $u_1\, e_y\, u_2 \,\cdots\, u_2\, e_x\, u_1$). If Γ is dual Eulerian, then Γ_1 is dual Eulerian. If k is odd, then without loss of generality assume the dual Euler path in Γ^d is $u_1\, e_{i+1}\, u_2 \,\cdots\, u_1\, e_k\, u_2 \,\cdots\, u_2\, e_1\, u_1 \,\cdots\, u_1\, e_i\, u_2$. If the even series (respectively, parallel) set of edges in Γ (respectively, Γ^d) is replaced by a single edge, say e_z, to obtain Γ_1 (respectively, Γ_1^d), then the dual Euler path in Γ_1 (respectively, Γ_1^d) is either $v_1\, e_z\, v_2 \,\cdots$ or $\cdots\, v_1\, e_z\, v_2$ (re-

spectively, $u_1\, e_z\, u_2 \cdots$ or $\cdots u_1\, e_z\, u_2$). If Γ is dual Eulerian, then Γ_1 is dual Eulerian.

For the induction hypothesis assume that the reduction operations have been applied recursively to Γ and the result is a graph Γ_{r-1} and if Γ is dual Eulerian, then Γ_{r-1} is dual Eulerian. By the same argument as was used to prove the basis case, Γ_r, which is obtained from Γ_{r-1} by applying the reduction operations, must be dual Eulerian if Γ_{r-1} is dual Eulerian. Hence, if Γ is dual Eulerian, then Γ_r is dual Eulerian.

Suppose Γ is not dual Eulerian. We claim that repeated applications of the series and parallel reduction operators cannot produce a graph Γ_r which is dual Eulerian. If Γ is not dual Eulerian, then there exists at least two edge disjoint dual paths in Γ which do not share end vertices. An application of the parallel reduction operation does not change the number of vertices in Γ, and therefore cannot cause two disjoint dual paths to be joined. An application of the series reduction operation eliminates vertices of degree two, but does not cause any two edges to be incident on the same vertex if they were not already. Hence, if Γ_r is dual Eulerian, then Γ is dual Eulerian. □

An interesting special case of the previous theorem is stated in the following corollary.

Corollary 3 *If Γ_r is K_2, then Γ is dual Eulerian.*

Corollary 3 is not a necessary condition for a graph to be dual Eulerian, since the graph of Fig. 5 is dual Eulerian, but its reduced graph is not K_2.

III Identifying Dual Eulerian Graphs

It is assumed throughout this section that Γ is a reduced graph. The determination of dual paths in a plane undirected multigraph Γ requires the traversal of edges in both itself and its dual in corresponding sequences. Suppose $v_1\, e_1\, v_2\, e_2\, v_3\, e_3\, v_4$ is a dual path in Γ and $u_1\, e_1\, u_2\, e_2\, u_3\, e_3\, u_4$ is its corresponding dual path in Γ^d. These dual paths can be extended in length if and only if there exists an edge $e_i \in E$ such that $e_i \in (f^*(v_4) \cap f^*(u_4)) \cup (f^*(v_1) \cap f^*(u_1))$ and $e_i \neq e_1$, $e_i \neq e_2$ and $e_i \neq e_3$. Therefore, the intersections of vertex cocycles of Γ and Γ^d indicate which edges can be adjacent in a dual path of Γ.

Let R denote the set of vertex cocycles of a graph $\Gamma(V, f, E)$ (i.e., $R = \{f^*(v) | v \in V\}$). Similarly, let R^d denote the set of vertex cocycles of Γ^d. Define the set T as follows:

$$T = \{x \cap y | x \in R; y \in R^d\}.$$

That is, take the intersection of all possible pairs of elements of R and R^d and let these be the elements of the set T.

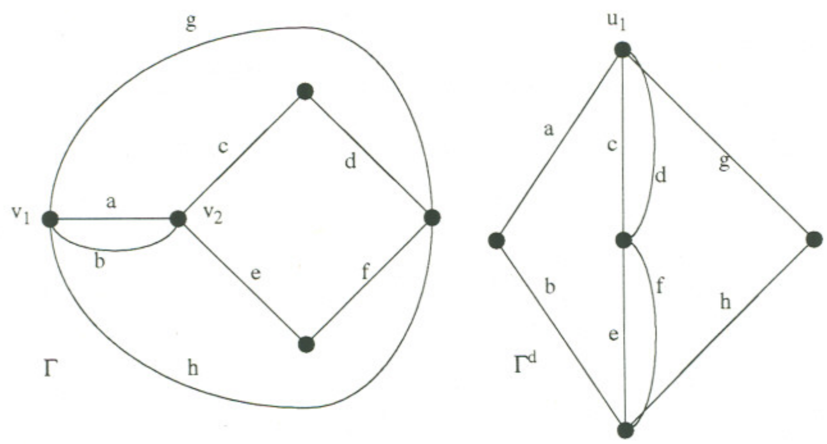

Fig. 5. A pair of dual plane multigraphs.

Lemma 4 *All nonempty elements of T have cardinality two.*

Proof: The set of vertex cocycles R^d of Γ^d corresponds to the set of facial cycles of Γ. Therefore T is formed by taking the pairwise intersections of vertex cocycles and facial cycles in Γ. Choose some vertex $v \in V$ and facial cycle ζ in Γ. If v is not in the facial circuit which corresponds to the facial cycle ζ, then the intersection is empty. If v is contained in the facial circuit corresponding to ζ, then the intersection must have exactly two elements because ζ is elementary. □

The elements of T represent sets of edges which are incident on a common vertex in both Γ and Γ^d. Suppose $\{a, b\}$ is an element of T for some Γ and Γ^d. This means that there exists a dual path in which a and b are adjacent. Now it must be determined which edges can be placed adjacent to a and b such that a larger dual path can be constructed. For this purpose a table is constructed which is known as a *successor table*. The successor table is constructed beginning with all possible dual paths of length two. This information can be obtained from T. Each element of T, say $\{a, b\}$, is ordered in two ways, $(a \to b)$ and $(b \to a)$. These two orderings represent the traversal of a then b and b then a, respectively. In order to determine which edge can be appended to the dual path $(a \to b)$, the possible successors in Γ and Γ^d are listed and their intersection is taken.

Table I. The successor table for the plane multigraphs shown in Fig. 5.

	ordered pairs	Γ-successors	Γ^d-successors	*dual*-successors
1	$(a \to g)$	$\{d, f, h\}$	$\{h\}$	$\{h\}$
2	$(g \to a)$	$\{b, c, e\}$	$\{b\}$	$\{b\}$
3	$(a \to b)$	$\{c, e\}$ or $\{g, h\}$	$\{e, f, h\}$	$\{e\}$ or $\{h\}$
4	$(b \to a)$	$\{c, e\}$ or $\{g, h\}$	$\{c, d, g\}$	$\{c\}$ or $\{g\}$
5	$(g \to h)$	$\{d, f\}$ or $\{a, b\}$	$\{b, e, f\}$	$\{f\}$ or $\{b\}$
6	$(h \to g)$	$\{d, f\}$ or $\{a, b\}$	$\{a, c, d\}$	$\{d\}$ or $\{a\}$
7	$(b \to h)$	$\{d, f, g\}$	$\{g\}$	$\{g\}$
8	$(h \to b)$	$\{a, c, e\}$	$\{a\}$	$\{a\}$
9	$(a \to c)$	$\{d\}$	$\{d, e, f\}$	$\{d\}$
10	$(c \to a)$	$\{b, g, h\}$	$\{b\}$	$\{b\}$
11	$(c \to e)$	$\{f\}$	$\{b, f, h\}$	$\{f\}$
12	$(e \to c)$	$\{d\}$	$\{a, d, g\}$	$\{d\}$
13	$(b \to e)$	$\{f\}$	$\{c, d, f\}$	$\{f\}$
14	$(e \to b)$	$\{a, g, h\}$	$\{a\}$	$\{a\}$
15	$(c \to d)$	$\{f, g, h\}$	$\{a, g\}$ or $\{e, f\}$	$\{g\}$ or $\{f\}$
16	$(d \to c)$	$\{a, b, e\}$	$\{a, g\}$ or $\{e, f\}$	$\{a\}$ or $\{e\}$
17	$(e \to f)$	$\{d, g, h\}$	$\{b, h\}$ or $\{c, d\}$	$\{h\}$ or $\{d\}$
18	$(f \to e)$	$\{a, b, c\}$	$\{b, h\}$ or $\{c, d\}$	$\{b\}$ or $\{c\}$
19	$(d \to g)$	$\{a, b, h\}$	$\{h\}$	$\{h\}$
20	$(g \to d)$	$\{c\}$	$\{c, e, f\}$	$\{c\}$
21	$(d \to f)$	$\{e\}$	$\{b, e, h\}$	$\{e\}$
22	$(f \to d)$	$\{c\}$	$\{a, c, g\}$	$\{c\}$
23	$(f \to h)$	$\{a, b, g\}$	$\{g\}$	$\{g\}$
24	$(h \to f)$	$\{e\}$	$\{c, d, e\}$	$\{e\}$

This intersection represents the set of edges which may be appended to $(a \to b)$ in order to create a dual path of length three. The successor table for the dual plane multigraphs shown in Fig. 5 is given in Table I. Placed in the first column of the successor table are two ordered pairs (each in a separate row) for each element in T. Placed in the second (respectively, third) column are the possible successors of the ordered pair in Γ (respectively, Γ^d). The intersection of the second and third columns is placed in the fourth column, and represents the edges which may succeed the ordered pair in a dual path (i.e., a successor common to both Γ and Γ^d). The sets in the second, third and fourth columns of the successor table are referred to as the Γ-*successors,* Γ^d-*successors* and *dual-successors*, respectively. For the example shown in Fig. 5 vertices v_2 of Γ and u_1 of Γ^d contribute the element $\{a, c\} \in T$ (i.e., $f^*(v_2) \cap f^*(u_1) = \{a, c\} \in T$). This element of T contributes two rows to the successor table as shown in rows 9 and 10 of Table I.

The ordered pairs in the first column of the successor table represent all possible dual paths of length two. The dual-successors of an ordered pair are a set of edges that can be appended to the dual path of length two to form a dual path of length three. Suppose $(x \rightarrow y)$ is an ordered pair of a successor table and $\{z\}$ is its dual-successor. $x\,y\,z$ is a dual path of length three, and therefore $y\,z$ must be a dual path of length two. Since all dual paths of length two are represented in the successor table, the ordered pair $(y \rightarrow z)$ must be in the table. The ordered pair $(y \rightarrow z)$ has a set of dual-successors, so the dual path can be extended. Before taking this development further, let us analyze one exceptional case.

The case in which two edges are in parallel requires special consideration, since the ordered pair does not identify a unique vertex at which the dual path ends. Two edges in parallel are referred to as a *two-element circuit.* For example in Γ of Fig. 5, $(a \rightarrow b)$ can imply either the dual path $v_1\,a\,v_2\,b\,v_1$ or $v_2\,a\,v_1\,b\,v_2$. Since it is desirable that an ordered pair identify a unique dual path, the ordered pairs corresponding to two-element circuits are split in two so that each identifies a unique dual path of length two. For example, rows 3, 4, 5, 6, 15, 16, 17 and 18 of the successor table shown in Table I must be split into two rows each. A new table is constructed by splitting these rows and is known as the augmented successor table. The augmented successor table derived from the successor table in Table I is shown in Table II.

In the case where an entry in the first column of the successor table corresponds to a two-element circuit in either Γ or Γ^d there are two sets in either the second or third column corresponding to the two possible ways to traverse the two-element circuit. Consider the two-element circuit $v_1\,a\,v_2\,b\,v_1$ in the graph of Fig. 5. The Γ-successor of $(a \rightarrow b)$ may be either $\{c, e\}$ or $\{g, h\}$ depending on whether $(a \rightarrow b)$ is traversed starting at v_2 or v_1, respectively. Rows 3 and 4 of Table I are contributed by the two-element circuit $\{a, b\} \in T$. The edge b is the dual-successor of the ordered pairs $(g \rightarrow a), (g \rightarrow h), (c \rightarrow a)$ and $(f \rightarrow e)$ (rows 2, 5, 10 and 18 of Table I, respectively). The dual path $a\,b$ is contained in the dual paths $g\,a\,b$ and $c\,a\,b$. Concatenating the dual paths $g\,a\,b$ and $c\,a\,b$ with the dual-successors of $(a \rightarrow b)$ yields the edge sequences $g\,a\,b\,e$, $g\,a\,b\,h$, $c\,a\,b\,e$, and $c\,a\,b\,h$. $g\,a\,b\,h$ and $c\,a\,b\,e$ are dual paths, but $g\,a\,b\,e$ and $c\,a\,b\,h$ are not. Therefore, the ordered pair $(a \rightarrow b)$ is split into two ordered pairs $(a \rightarrow b)'$ and $(a \rightarrow b)''$ with dual-successors $\{h\}$ and $\{e\}$, respectively (rows 17 and 18 of Table II, respectively), and the dual-successors of $(g \rightarrow a)$ and $(c \rightarrow a)$ are updated to $\{b\}'$ and $\{b\}''$, respectively (rows 2 and 6 of Table II, respectively). Similarly, all other two-element circuits are split.

Lemma 5 *The dual-successor of an ordered pair in the augmented successor table is unique.*

Proof: Assume the ordered pair is not a two-element circuit. Given an arbitrary ordered pair $(x \rightarrow y)$, without loss of generality assume it corresponds to the dual path $v_1\,x\,v_2\,y\,v_3$ in

Table II. The augmented successor table derived from the successor table shown in Table I.

	ordered pairs	Γ-successors	Γ^d-successors	*dual*-successors
1	$(a \to g)$	$\{d, f, h\}$	$\{h\}$	$\{h\}'$
2	$(g \to a)$	$\{b, c, e\}$	$\{b\}$	$\{b\}'$
3	$(b \to h)$	$\{d, f, g\}$	$\{g\}$	$\{g\}'$
4	$(h \to b)$	$\{a, c, e\}$	$\{a\}$	$\{a\}'$
5	$(a \to c)$	$\{d\}$	$\{d, e, f\}$	$\{d\}'$
6	$(c \to a)$	$\{b, g, h\}$	$\{b\}$	$\{b\}''$
7	$(c \to e)$	$\{f\}$	$\{b, f, h\}$	$\{f\}'$
8	$(e \to c)$	$\{d\}$	$\{a, d, g\}$	$\{d\}''$
9	$(b \to e)$	$\{f\}$	$\{c, d, f\}$	$\{f\}''$
10	$(e \to b)$	$\{a, g, h\}$	$\{a\}$	$\{a\}''$
11	$(d \to g)$	$\{a, b, h\}$	$\{h\}$	$\{h\}''$
12	$(g \to d)$	$\{c\}$	$\{c, e, f\}$	$\{c\}'$
13	$(d \to f)$	$\{e\}$	$\{b, e, h\}$	$\{e\}'$
14	$(f \to d)$	$\{c\}$	$\{a, c, g\}$	$\{c\}''$
15	$(f \to h)$	$\{a, b, g\}$	$\{g\}$	$\{g\}''$
16	$(h \to f)$	$\{e\}$	$\{c, d, e\}$	$\{e\}''$
17	$(a \to b)'$	$\{g, h\}$	$\{e, f, h\}$	$\{h\}$
18	$(a \to b)''$	$\{c, e\}$	$\{e, f, h\}$	$\{e\}$
19	$(b \to a)'$	$\{g, h\}$	$\{c, d, g\}$	$\{g\}$
20	$(b \to a)''$	$\{c, e\}$	$\{c, d, g\}$	$\{c\}$
21	$(g \to h)'$	$\{a, b\}$	$\{b, e, f\}$	$\{b\}$
22	$(g \to h)''$	$\{d, f\}$	$\{b, e, f\}$	$\{f\}$
23	$(h \to g)'$	$\{a, b\}$	$\{a, c, d\}$	$\{a\}$
24	$(h \to g)''$	$\{d, f\}$	$\{a, c, d\}$	$\{d\}$
25	$(c \to d)'$	$\{f, g, h\}$	$\{a, g\}$	$\{g\}$
26	$(c \to d)''$	$\{f, g, h\}$	$\{e, f\}$	$\{f\}$
27	$(d \to c)'$	$\{a, b, e\}$	$\{a, g\}$	$\{a\}$
28	$(d \to c)''$	$\{a, b, e\}$	$\{e, f\}$	$\{e\}$
29	$(e \to f)'$	$\{d, g, h\}$	$\{c, d\}$	$\{d\}$
30	$(e \to f)''$	$\{d, g, h\}$	$\{b, h\}$	$\{h\}$
31	$(f \to e)'$	$\{a, b, c\}$	$\{c, d\}$	$\{c\}$
32	$(f \to e)''$	$\{a, b, c\}$	$\{b, h\}$	$\{b\}$

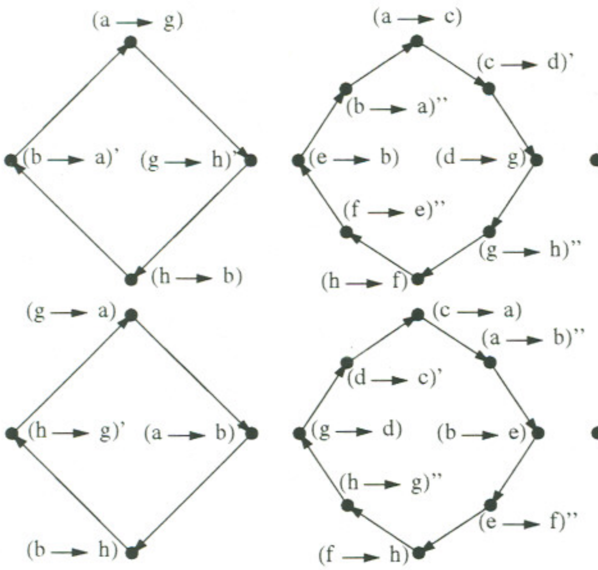

Fig. 6. Continuity graph corresponding to the augmented successor table given in Table II.

Γ and $u_1\, x\, u_2\, y\, u_3$ in Γ^d. Since $y \in (f^*(v_3) \cap f^*(u_3))$ and $|f^*(v_3) \cap f^*(u_3)| = 2$ (Lemma 4), $|(f^*(v_3) \setminus \{y\}) \cap (f^*(u_3) \setminus \{y\})| = 1$. Hence, the dual-successor is unique.

Assume the ordered pair is a two-element circuit. Using the same argument as in the preceding paragraph, the dual-successor is unique for each starting vertex. Since the ordered pair is split, the dual-successor is unique. □

A directed graph $\Lambda(V, A)$, henceforth known as the *continuity graph*, is constructed where V is the set of vertices and A the set of directed arcs. The continuity graph is constructed in such a way that there is an one-to-one correspondence between the vertices of Λ and the ordered pairs of the augmented successor table. Hence, for simplicity, in the following discussion the ordered pairs are used to denote the corresponding vertices of Λ. An arc is directed from vertex $(e_1 \rightarrow e_2)$ to vertex $(e_3 \rightarrow e_4)$ if and only if e_2 and e_3 are identical edges in Γ and $\{e_4\}$ is the dual-successor of $(e_1 \rightarrow e_2)$. The arc indicates that it is possible to traverse the edges $e_1, e_2 = e_3$ and e_4 in Γ and Γ^d in that order. The continuity graph constructed from the augmented successor table of Table II is shown in Fig. 6.

A *chain* in a directed graph Λ is a sequence of vertices $v_1\, v_2\, \cdots\, v_n$ such that there is an arc from v_i to v_{i+1} for $1 \leq i < n$. A chain is said to be *closed* if $v_1 = v_n$. A chain in the continuity graph Λ corresponds to a sequence of edges traversed in Γ and Γ^d. A *maximal chain* is a chain whose set of vertices is not properly contained in the set of vertices of any other chain. For example, $(a \to c)(c \to d)'(d \to g)$ is a chain in the continuity graph of Fig. 6 and $(a \to g)(g \to h)'(h \to b)(b \to a)'$ is a maximal chain.

The *mirror image* of a chain $v_1 v_2 \cdots v_k$ in a continuity graph is the chain $v'_k v'_{k-1} \cdots v'_1$ where v_i and v'_i contain the same edge labels with opposite ordering (e.g., if $v_i = (a \to b)$, then $v'_i = (b \to a)$).

Lemma 6 *For every chain in a continuity graph Λ, there is a mirror image of that chain also contained in Λ.*

Proof: Consider an arbitrary ordered pair $(a \to b)$ and its row in the augmented successor table. Suppose its dual-successor is c. Then the ordered pair $(b \to c)$ exists in the first column of the table, and $(a \to b)(b \to c)$ is a chain in Λ. It must be shown that $(c \to b)(b \to a)$ is also a chain in Λ. Since $(b \to c)$ is an ordered pair in the first column of the table, so is $(c \to b)$. The Γ-successors of $(a \to b)$ are the elements of the vertex cocycle of Γ which contains b and not a with the element b removed. That is, it is the set $f^*(x) \setminus \{b\}$ such that $b \in f^*(x)$ and $a \notin f^*(x)$. Similarly, the Γ-successors of $(c \to b)$ are the elements of the vertex cocycle $f^*(y)$ of Γ less b, where $b \in f^*(y)$ and $c \notin f^*(y)$. Since $b \in f^*(x) \cap f^*(y)$ and Γ has no loops, $x \neq y$. By hypothesis, a and b must be in the same vertex cocycle of Γ for some $v \in V$. b is contained in only two vertex cocycles (i.e., $|f(e)| = 2, \forall e \in E$) and $a \notin f^*(x)$, therefore $a \in f^*(y)$. Hence a is in the set of Γ-successors of $(c \to b)$. That is, Γ contains the subgraph shown in Fig. 7(a), and Γ^d contains the subgraph shown in Fig. 7(b). A similar argument shows that a is in the set of Γ^d-successors of $(c \to b)$ in the augmented successor table and therefore $(c \to b)(b \to a)$ is a chain in Λ. □

For example, the chains $(a \to g)(g \to h)'(h \to b)(b \to a)'$ and $(a \to b)'(b \to h)(h \to g)'(g \to a)$ in the continuity graph of Fig. 6 are mirror images of one another. The mirror image of a chain corresponds to the traversal of edges in Γ and Γ^d in the opposite direction.

Lemma 7 *Every vertex in a continuity graph Λ has exactly one incoming arc and one outgoing arc.*

Proof: Choose an arbitrary vertex $v \in V$. From Lemma 5 it is known that v has one outgoing arc. From Lemma 6 it is known that there is a corresponding $v' \in V$ that is the mirror image of v. Since v has one outgoing arc, v' has only one incoming arc. It is also known from Lemma 5 that v' has only one outgoing arc; hence, v has only one incoming arc. □

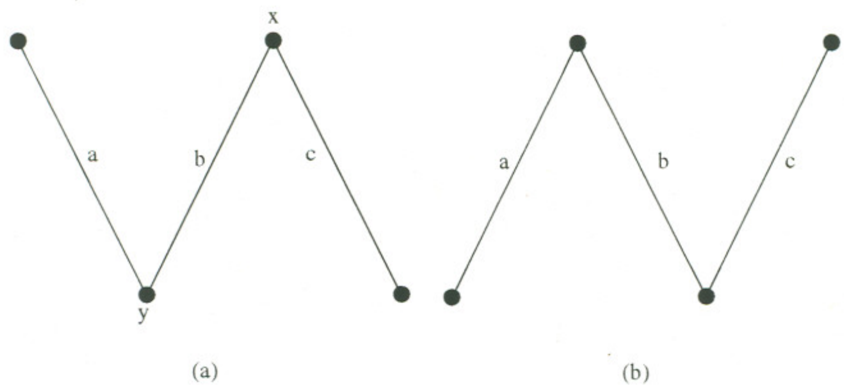

Fig. 7. Subgraphs illustrating the proof of Lemma 6.

Corollary 8 *A chain in a continuity graph Λ is maximal if and only if it is closed.*

Proof: If a chain in Λ is closed, then by Lemma 7 this chain must be maximal. Suppose there exists a maximal chain in Λ which is not closed. There must be a vertex in Λ which has an incoming arc and no outgoing arc which is a contradiction to Lemma 7. □

The property of the continuity graph Λ stated in Lemma 7 makes the identification of chains in Λ trivial. Since each chain is contained in a maximal chain, the set of all maximal chains is sufficient to characterize the edge relationships of interest. A dichotomy of the vertex set of Λ is induced by the mirror image property. The dichotomy is obtained by placing in opposite sets chains which are mirror images of one another. Since the direction in which edges appear in the chains is not important, it is sufficient to analyze only one set of the dichotomy. Let the chains which connect the vertices of one set of the dichotomy be known as the *representative set of maximal chains*.

A *desired chain* in a continuity graph Λ is a chain, say $(e_0 \rightarrow e_1)(e_1 \rightarrow e_2)\cdots(e_n \rightarrow e_{n+1})$, such that $\forall i, 1 \leq i \leq n$, $\forall j, 1 \leq j < i$ and $\forall k, i < k \leq n$, $e_j \neq e_k$. A *maximal desired chain* is a desired chain of maximum length.

Lemma 9 *There is an one-to-one correspondence between desired chains of Λ and dual paths of Γ.*

Proof: Suppose there is a desired chain in Λ that does not correspond to a dual path in Γ. There are two possibilities: either there is an arc in Λ for which there is no sequence of edges in Γ or Γ^d or the chain in Λ corresponds to a sequence of edges in Γ or Γ^d which traverses the same edge more than once. The former case cannot occur by the construction rules of the continuity graph. The latter case is a sequence of edges which is not a desired chain.

Suppose there is a dual path in Γ, say $e_1 \, e_2 \, \cdots \, e_n$, for which there is no corresponding desired chain in Λ. For some e_i $(1 \leq i \leq n-2)$, the dual path $e_i \, e_{i+1} \, e_{i+2}$ is not a desired chain in Λ. The ordered pair $(e_i \rightarrow e_{i+1})$ is in the augmented successor table, since e_i and e_{i+1} must be incident on a common vertex. From Lemma 7 it is known that every ordered pair has a dual-successor and it is unique. Furthermore, e_{i+2} cannot coincide with either e_i or e_{i+1} even if e_i and e_{i+1} are contained in a two-element cycle. Hence, $e_i \, e_{i+1} \, e_{i+2}$ must be a desired chain in Λ. □

The dual path corresponding to the desired chain $(e_0 \rightarrow e_1)(e_1 \rightarrow e_2) \cdots (e_n \rightarrow e_{n+1})$ is $e_1 \, e_2 \, \cdots \, e_n$. The above derivation has led to the major result of this paper which is presented in the next theorem.

Theorem 10 Γ *is dual Eulerian if and only if there exists a desired chain which contains every edge of* Γ.

Proof: Assume there exists a desired chain which contains every edge. From Lemma 9 it is known that every desired chain has a corresponding dual path in Γ and Γ^d. Hence, Γ and Γ^d contain a dual Euler path.

Suppose there is no desired chain which contains every edge. From Lemma 9 there is a desired chain corresponding to every dual path. Hence, there is no dual path which contains every edge. □

Theorem 2 in conjunction with Theorem 10 guarantee the successful identification of dual Eulerian plane multigraphs.

IV Algorithm Realization and Analysis

The input to the algorithm is a specification of a plane connected undirected multigraph. The reduced graph can be obtained easily by searching for vertices of degree 2 in Γ and Γ^d, and applying the series and parallel reduction operations. It is not necessary to identify parallel edges, since a parallel set of edges in Γ (respectively, Γ^d) corresponds to a series set of edges in Γ^d (respectively, Γ). The imbedding of a multigraph on the plane may be specified by indexing the edges incident on each vertex in the clockwise direction. The indices are assigned in the clockwise

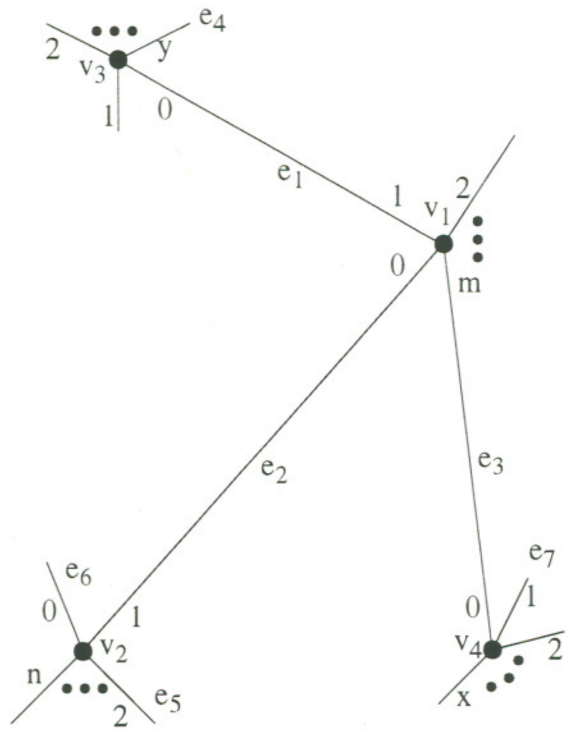

Fig. 8. Partial graph used in the explanation of procedure **Determine_Dual-Successors**.

direction with the first index being zero. Each edge has two indices assigned to it; one for each vertex it is incident on. The index of the edge e incident on the vertex v is denoted by $index(v, e)$. For example in the partial graph shown in Fig. 8, $index(v_1, e_1) = 1$. Each edge appears in exactly four elements of T, except for edges which are contained in two-element cycles; these edges appear in three elements of T. Consider the partial graph shown in Fig. 8. The elements of T in which the edge e_2 appears are $\{e_2, e_1\}, \{e_2, e_3\}, \{e_2, e_5\}$, and $\{e_2, e_6\}$. These four elements can be determined in constant time due to the data structure used for a plane graph. Two edges e_1 and e_2 are said to be *neighbors* of one another if and only if $index(v, e_1) = (index(v, e_2) + 1) \bmod |f^*(v)|$ or $index(v, e_1) = (index(v, e_2) - 1) \bmod |f^*(v)|$. The algorithm to form the set T is shown in Fig. 9.

Lemma 11 *$|T| = 2|E| - k + 1$, where k is the number of two-element circuits.*

Proof: Each $v \in V$ contributes exactly $|f^*(v)|$ elements to T. For each $v \in V$ these elements are unique except for vertices which are contained in two-element circuits. The same two element set of edges in T is contributed by different vertices if the vertices are contained in a two-element circuit. In addition, the empty set is an element of T. Hence,

$$|T| = \sum_{\forall v \in V} |f^*(v)| - k + 1.$$

```
Procedure Determine_T(){
    for (each e ∈ E){
        for (each neighbor, say e_i of e){
            if (e_i is not marked)
                add {e, e_i} to T;
        }
        mark e;
    }
}
```

Fig. 9. Algorithm for determining the set T.

Since

$$\sum_{\forall v \in V} |f^*(v)| = 2|E|[3],$$

$$|T| = 2|E| - k + 1.$$

□

Consider the example shown in Fig. 5. The neighbors of the edge b in Γ are a, e and h. Therefore, during the construction of T the sets $\{b, e\}$, $\{a, b\}$ and $\{b, h\}$ are placed in T and another edge is considered. The neighbors of the edge a in Γ are b, c and g. The sets $\{a, c\}$ and $\{a, g\}$ are placed in T. Since b is marked, $\{a, b\}$ is not placed in T a second time. This process is repeated for each edge and the result is $T = \{\{a,g\},\{a,b\},\{g,h\},\{b,h\},\{a,c\},\{c,e\},\{b,e\},\{c,d\},\{e,f\},\{d,g\},\{d,f\},\{f,h\}\}$. This algorithm is clearly $O(|E|)$, since the number of neighbors of an edge is constant.

Lemma 12 *The augmented successor table contains exactly* $4|E|$ *rows.*

Proof: From Lemma 11 it is known that $|T| = 2|E| - k + 1$, where k is the number of two-element cycles. Each element of T (except the empty set) contributes two rows to the successor table. Therefore the successor table has $4|E| - 2k$ rows. Each two-element cycle contributes two rows to the successor table and each row is split when the augmented successor table is formed. Hence, the number of rows in the augmented successor table is $4|E| - 2k + 2k = 4|E|$. □

The dual-successors of the ordered pairs are determined using the algorithm shown in Fig. 10. In the graph of Fig. 5, the dual-successors of the ordered pairs $(a \to g)$ and $(g \to a)$ are h and b, respectively. The dual-successors of the ordered pairs $(b \to h)$ and $(h \to b)$ are g and a, respectively. The body of the *for* loop in the algorithm **Determine_Dual-Successors** is executed in constant

```
Procedure Determine_Dual-Successors(){
    for (each {e_i, e_j} ∈ T){ /*Fig. 8*/
        if ({e_i, e_j}is a two-element cycle){
                split it into two elements and mark the direction of traversal in each;
                place one of them in T so that it gets processed in a successive iteration;
        }
        if (index(v_1, e_j) = (index(v_1, e_i) + 1) mod |f*(v_1)|){
            /*e_i and e_j correspond to e_1 and e_2 in Fig. 8, respectively*/
            the dual-successor of (e_j → e_i) is the edge e_l such that
            (index(v_3, e_l) + 1) mod |f*(v_3)| = index(v_3, e_i);
            /*e_l corresponds to e_4 in Fig. 8*/
            the dual-successor of (e_i → e_j) is the edge e_l such that
            (index(v_2, e_l) − 1) mod |f*(v_2)| = index(v_2, e_j);
            /*e_l corresponds to e_5 in Fig. 8*/
        }
        else{
            /*e_i and e_j correspond to e_3 and e_2 in Fig. 8, respectively*/
            the dual-successor of (e_i → e_j) is the edge e_l such that
            (index(v_2, e_l) + 1) mod |f*(v_2)| = index(v_2, e_j);
            /*e_l corresponds to e_6 in the graph of Fig. 8*/
            the dual-successor of (e_j → e_i) is the edge e_l such that
            (index(v_4, e_l) − 1) mod |f*(v_4)| = index(v_4, e_i);
            /*e_l corresponds to e_7 in Fig. 8*/
        }
    }
}
```

Fig. 10. Algorithm for determining the dual-successors.

```
Procedure Determine_Representative_Set_of_Maximal_Chains(){
    while (an unvisited ordered pair exists in the augmented successor table){
        Choose any unvisited ordered pair, say (a → b), and mark it and its mirror image visited;
        (x → y) = (a → b); /*assign (a → b) to the temporary variable (x → y)*/
        do{
            choose the dual-successor of (x → y), say z;
            (x → y) = (y → z);
            mark (y → z) and (z → y) visited;
        }while((a → b) ≠ (x → y));
        record the maximal chain;
    }
}
```

Fig. 11. Algorithm for determining the representative set of maximal chains.

time, and the body of the loop is executed once for each element of T. Therefore the time complexity of the procedure is $O(|E|)$.

Consider the augmented successor table given in Table II. The procedure **Determine_Representative_Set_of_Maximal_Chains** works as follows. The ordered pair $(a \to g)$ is chosen as the starting point of the first maximal chain. The ordered pairs $(a \to g)$ and $(g \to a)$ are marked visited. Since the dual-successor of $(a \to g)$ is $\{h\}'$, the next ordered pair in the chain is $(g \to h)'$. The ordered pairs $(g \to h)'$ and $(h \to g)'$ are marked visited and the next ordered pairs in the chain are $(h \to b)$, and $(b \to a)'$. Since $(a \to g)$ is the next ordered pair and it has already been visited, the maximal chain is complete. Not all the ordered pairs have been visited so the body of the outer *while* loop is repeated. The next unvisited ordered pair in the augmented successor table is $(a \to c)$. Beginning with this ordered pair yields the maximal chain $(a \to c)(c \to d)'(d \to g)(g \to h)''(h \to f)(f \to e)''(e \to b)(b \to a)''$. The procedure continues until all the ordered pairs in the table have been marked visited.

Lemma 7 and Theorem 8 guarantee the correctness of the procedure **Determine_Representative_Set_of_Maximal_Chains**. It is clear that the procedure **Determine_Representative_Set_of_Maximal_Chains** has time complexity $O(|E|)$, since it visits exactly half the rows of the augmented successor table and performs a constant time task at each row.

Since an edge of Γ appears in at most four elements of T, it appears in at most eight vertex labels in the continuity graph Λ, and therefore in at most four vertex labels in the representative

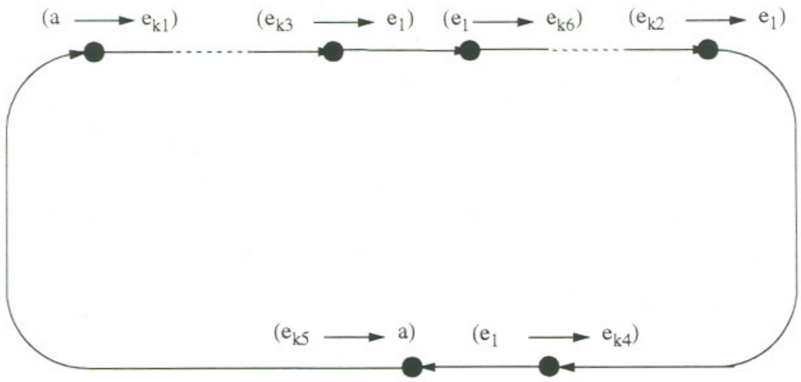

Fig. 12. A maximal chain containing a repeated edge, namely e_1.

set of maximal chains. Hence, an edge in Γ can appear at most four times in a maximal chain. Based on these observations the set of maximal desired chains is obtained from the representative set of maximal chains as follows. Choose a maximal chain from the representative set of maximal chains, say $v_1 \, v_2 \, \cdots \, v_n \, v_1$. A desired chain is a *subchain* of a maximal chain which contains no repeated edges. A subchain is a chain which is contained in another chain. Therefore, a desired chain is maximal if it begins with an edge which appears twice in the maximal chain, and ends just prior to an edge which is already contained in the desired chain. The maximal desired chains are determined using a constructive algorithm as follows. Begin a maximal desired chain at a vertex $v_i = (e_k \rightarrow e_p)$ $(1 \leq i \leq n)$ for which there is another vertex $v_j = (e_l \rightarrow e_k)$ $(1 \leq j \leq n$ and $j \neq i-1)$ for arbitrary p and l. That is, the label e_k appears four times in the maximal chain. The maximal desired chain is $v_i \, v_{i+1} \, v_{i+2} \, \cdots \, v_k$, where v_k is of the form $(e_{k_1} \rightarrow e_l)$ and there exists $v_m = (e_{k_2} \rightarrow e_l)$ for some m $(i \leq m \leq k-1)$ and some arbitrary k_i $(i = 1, 2)$ (see Fig. 12). Begin construction of the next maximal desired chain at $v_{m+1} = (e_l \rightarrow e_{k_3})$ for arbitrary k_3. Repeat this until the first maximal desired chain constructed from this maximal chain is repeated. If each repeated edge has been used as the start of a maximal desired chain, then choose a new maximal chain and repeat the outermost loop of the algorithm; otherwise, choose a repeated edge which has

```
Procedure Determine_Maximal_Desired_Chains(){
    for (each maximal chain in the representative set of maximal chains){
        if (the maximal chain is of the form ···, (e_i → e_{i+1}), (e_{i+1} → e_{i+2}), ···, (e_j → e_{i+1}),
        (e_{i+1} → e_{j+1}), ···){ /*it has a repeated edge, namely e_{i+1}*/
            while ( ∃e_{i+1} which is not the start of a maximal desired chain){
                    choose an e_{i+1} arbitrarily;
                    (y → z) = (e_{i+1} → e_{i+2});
                    do{
                        (m → n) = (y → z);
                        while (n ≠ e_{k+1}, ∀(e_k → e_{k+1}) ∈ MDC)
                        /*MDC is the maximal desired chain currently being constructed*/
                                (m → n) = (n → e_l);
                        (y → z) = (e_{k+1} → e_{k+2});
                    }while ((y → z) ≠ (e_{i+1} → e_{i+2}));
            }
        }
        else
            the maximal chain is a maximal desired chain;
    }
}
```

Fig. 13. Algorithm for determining the maximal desired chains.

not been used as the start of a maximal desired chain and continue. If a repeated edge does not exist in the maximal chain, then the maximal chain is a maximal desired chain. The algorithm is shown in Fig. 13. The maximal desired chains for the example of Fig. 5 are $a\,g\,h\,b$, $a\,c\,d\,g\,h\,f\,e\,b$ and $c\,e\,f\,d$. Since the maximal desired chain $a\,c\,d\,g\,h\,f\,e\,b$ contains every edge, the graphs shown in Fig. 5 are dual Eulerian.

The maximal chains are stored as linked lists of pointers to the data structures representing the edges. Each edge appears exactly twice in the set of maximal desired chains and each existence is a pointer to the same data object. Therefore, when the data structure of an edge is updated, the change is reflected in every instance of that edge. The maximal desired chains are recorded as a pair of pointers to the start and end positions in the maximal chain from which it is derived. The first task performed by the procedure **Determine_Maximal_Desired_Chains** is the search for the starting point of a maximal desired chain. This task is performed once for each maximal chain and traverses each link in a maximal chain at most once; hence, this task requires $O(|E|)$ time. Associated with each edge is an index which is initialized to the maximum positive integer. The starting position of a maximal desired chain being constructed is assigned an index of zero, and a pointer is recorded to this position. The links of the maximal chain are traversed assigning to each edge an index one greater than the previous edge. If an edge is traversed for which its index is less than the edge previously traversed, then this edge already exists in the current maximal desired chain and the maximal desired chain terminates. Contained in the data structure for each edge are two pointers, one to each instance in the set of maximal chains. Hence, the start of the next maximal desired chain is obtained in constant time and the traversal of edges continues at the end of the maximal desired chain previously completed. This clever use of pointers results in an algorithm which visits each link of the maximal chains only once, and performs a constant amount of work at each link. Hence, the procedure **Determine_Maximal_Desired_Chains** has time complexity $O(|E|)$.

Theorem 13 *The overall time complexity of the algorithm which identifies dual Eulerian plane multigraphs is* $O(|E|)$.

V Application of the Theory

In the design of functional cells for CMOS VLSI circuits it is desirable to physically implement each functional cell such that it requires a minimum amount of area [2]. For example, the CMOS functional cell shown in Fig. 14(a) can be physically implemented as shown in Fig. 14(b); however, if the dual Eulerian theory presented in this paper is applied, then it can be implemented as shown in Fig. 14(c). The reduction in area due to the application of the dual Eulerian theory is significant.

VI Conclusions

It has been shown in this paper that the question of whether or not a plane undirected multigraph is dual Eulerian can be answered in a time proportional to a linear function of the number of edges in the graph, and an algorithm has been presented which answers this question. Interesting properties of plane multigraphs have been presented and can be summarized as follows.

1. A plane undirected multigraph Γ is dual Eulerian if and only if its reduced graph Γ_r is dual Eulerian (Theorem 2).
2. A dual path of length two can be extended in a unique way (Lemma 5).
3. A plane undirected multigraph Γ is dual Eulerian if and only if there exists a desired chain which contains every edge of Γ (Theorem 10).
4. The set of maximum length dual paths can be computed in linear time (Theorem 13).

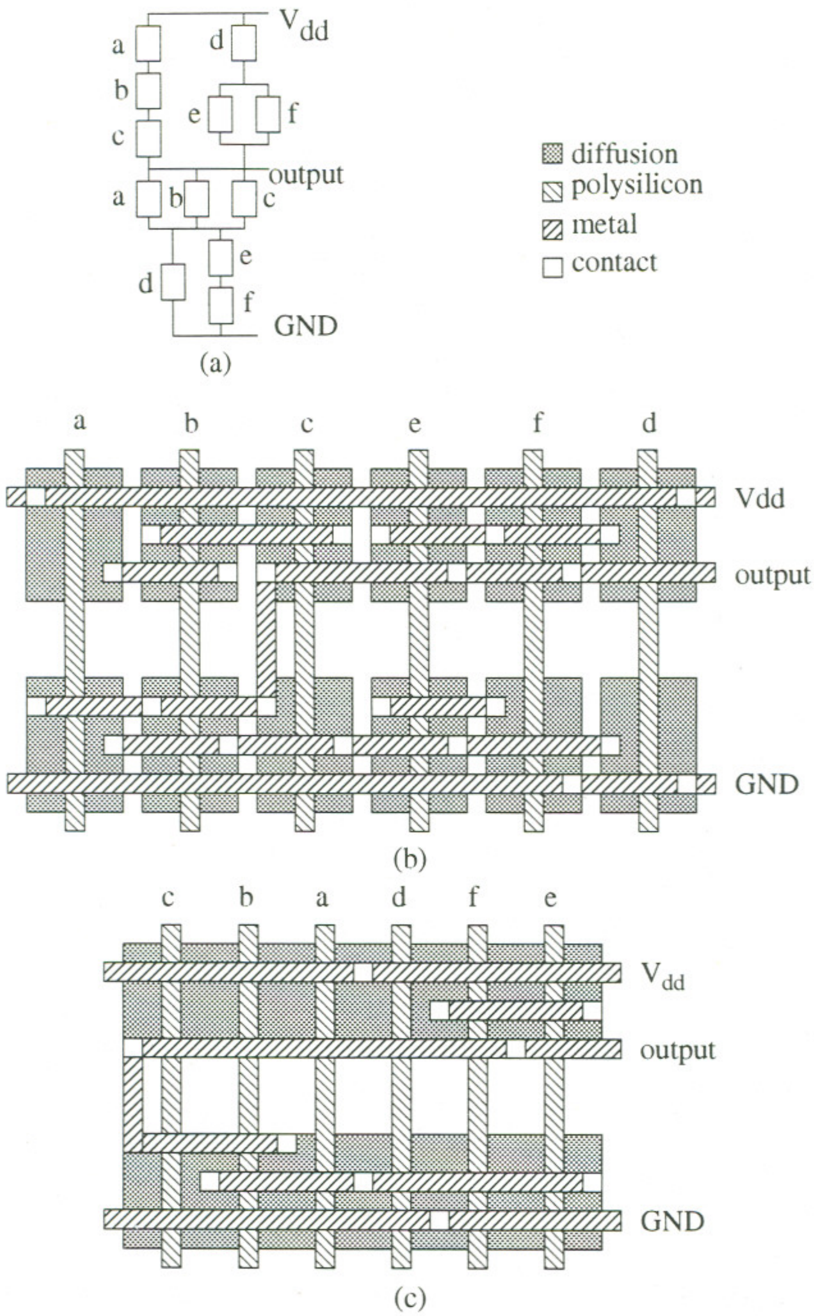

Fig. 14. (a) A CMOS functional cell, (b) a physical implementation of the cell and (c) a physical implementation of the cell after applying the dual Eulerian theory.

The question of whether or not a planar multigraph admits an imbedding which is dual Eulerian is interesting. Although an imbedding of a planar multigraph can be computed efficiently, it is not known whether or not all possible imbeddings of a planar multigraph can be computed efficiently. That is, the number of imbeddings is not easily countable for general planar multigraphs. Of course the imbedding, and hence the dual, is unique for triconnected planar multigraphs [3].

References

[1] F. Harary. *Graph Theory.* Addison-Wesley, 1971.

[2] T. Uehara and W.M. VanCleemput. *Optimal Layout of CMOS Functional Arrays*, IEEE Trans. on Comp., C-30:305–314, May 1981.

[3] J.E. Graver and M.E. Watkins. *Combinatorics with Emphasis on the Theory of Graphs.* Springer-Verlag, 1977.

[4] F.P. Preparata and R.T. Yeh. *Introduction to Discrete Structures.* Addison-Wesley, 1973.