JOURNAL OF COMPUTER AND SYSTEM SCIENCES 7, 323–331 (1973)

A $V \log V$ Algorithm for Isomorphism of Triconnected Planar Graphs*

J. E. HOPCROFT AND R. E. TARJAN

Department of Computer Science, Cornell University, Ithaca, New York 14850

Received May 17, 1971; revised September 17, 1972

An algorithm for determining whether two triconnected planar graphs are isomorphic is presented. The asymptotic growth rate of the algorithm is bounded by a constant times $|V| \log |V|$ where $|V|$ is the number of vertices in the graphs.

INTRODUCTION

The graph isomorphism problem is to determine if there exists a one-to-one mapping of the vertices of a graph onto the vertices of another graph which preserves adjacency of the vertices. At present there is no known algorithm for determining if two arbitrary graphs are isomorphic with a running time which is asymptotically less than exponential. Gotlieb and Corneil [2] have exhibited an efficient algorithm for a large class of graphs, namely those graphs with no k-strongly regular subgraph for large k.

Weinberg [9] has exhibited an algorithm with asymptotic running time of $|V|^2$ for isomorphism of triconnected planar graphs where $|V|$ is the number of vertices of the graph. The reason for restricting attention to triconnected planar graphs is that a triconnected planar graph has a unique representation on a sphere. In this paper we show that isomorphism of triconnected planar graphs can be tested in time proportional to $|V| \log |V|$.

DEFINITIONS AND NOTATION

A *graph* $G = (V, E)$ consists of a finite set of *vertices* V and a finite set of *edges* E. If the edges are unordered pairs (v, w) of vertices, the graph is *undirected*. If the edges are ordered pairs (v, w) of vertices, the graph is *directed*; v is called the *tail*

* This research was supported by the National Science Foundation under grant number NSF-GJ-96 and the Office of Naval Research under grant numbers N-00014-67-A-9112-0057 NR 004-402 and N-00014-67-A-0077-0021 NR 044-417. Reproduction in whole or part is permitted for any purpose of the United States Government.

571/7/3-6

of the edge and w is called the *head.* If (v, w) is a directed edge, its *reversal*, denoted by $(v, w)^r$, is the directed edge (w, v). If G is a graph, a *path* $p: v_1 \overset{*}{\Rightarrow} v_n$ in G is a sequence of edges $(v_1, v_2), (v_2, v_3), \ldots, (v_{n-1}, v_n)$ leading from v_1 to v_n. The path p is *simple* if $v_i \neq v_j$ for $i \neq j$. The path p is a *cycle* if $v_1 = v_n$ and $v_i \neq v_j$ for $i \neq j$ excluding the case $v_1 = v_n$.

A graph is *biconnected* if for each triple of distinct vertices v, w, and a, there is a path $p: v \overset{*}{\Rightarrow} w$ such that a is not on the path. A graph is *triconnected* if for each quadruple of distinct vertices v, w, a, and b, there is a path $p: v \overset{*}{\Rightarrow} w$ such that neither a nor b is on the path p. A graph G is *planar* if there exists a mapping of the edges of G into the plane in such a way that (1) each edge (v, w) is mapped into a simple curve, with v and w being mapped to the endpoints of the curve and (2) mappings of two distinct edges have only their common endpoints in common. Such a mapping of a planar graph is called a *planar embedding.* The connected sets of points which form the complement of the image of G in the plane are called *faces.* A triconnected planar graph has two representations in the plane [10] in the sense that for any two embeddings G_1 and G_2 either (1) for each vertex v in G_1 the order of edges around v in G_1 is the same as the order of the edges around v in G_2 or (2) for each vertex v in G_1 the order of edges around v in G_1 is the reverse of the order of the edges around v in G_2. Two graphs G_1 and G_2 are *isomorphic* if there exists a mapping of the vertices and edges of G_1 onto the vertices and edges of G_2 such that if edge e_1 is mapped onto e_2, then the head and tail of e_1 are mapped onto the head and tail, respectively, of e_2. Two planar embeddings are isomorphic if in addition to the foregoing property, the order of edges out of each vertex is preserved. If G_1 and G_2 are triconnected planar graphs, then G_1 and G_2 are isomorphic if and only if any planar embedding of G_1 is isomorphic to one of the two planar embeddings of G_2.

Let G be a fixed planar embedding of a graph G. The path $(v_1, v_2), (v_2, v_3), \ldots, (v_{n-1}, v_n)$ is said to be a *primary path* if for each i, $1 < i < n$, (v_i, v_{i+1}) is either the edge to the immediate right of (v_{i-1}, v_i) or the edge to the immediate left of (v_{i-1}, v_i) in the embedding of edges about v_i in the plane. Let p_1 be the path $(v_1, v_2), (v_2, v_3), \ldots, (v_{n-1}, v_n)$ and let p_2 be the path $(w_1, w_2), (w_2, w_3), \ldots, (w_{n-1}, w_n)$. If for each i, $1 < i < n$, (w_i, w_{i+1}) is the kth edge to the right of (w_{i-1}, w_i) implies that (v_i, v_{i+1}) is the kth edge to the right of (v_{i-1}, v_i) in the planar embedding, then p_2 is said to be the path *corresponding* to p_1 starting with edge (w_1, w_2).

Although the edges of G are undirected, we give an orientation to each edge (v_1, v_2) by treating it as two directed edges (v_1, v_2) and (v_2, v_1). Whenever we refer to an edge, we will be referring to one of the directed versions of that edge. Let λ be a mapping of edges into the integers such that $\lambda(e_1)$ equals $\lambda(e_2)$ if and only if the number of edges on the face to the right (left) of e_1 is the same as the number of edges on the face to the right (left) of e_2 and the degrees of the heads (tails) of e_1 and e_2 are the same.

Edges e_1 and e_2 are said to be *distinguishable* if there exist edges e_3 and e_4, a primary path p_1 starting with edge e_1 and ending with edge e_3, and a corresponding primary path p_2 starting with e_2 and ending with e_4 such that $\lambda(e_3) \neq \lambda(e_4)$. If e_1 and e_2 are not distinguishable they are said to be *indistinguishable.*

Basic Results

In this section we establish the basic result on which the isomorphism algorithm is based. Namely, two planar embeddings G_1 and G_2 are isomorphic if and only if there exists an edge e_1 in G_1 and an indistinguishable edge e_2 in G_2.

LEMMA 1. *Let G_1 and G_2 be planar embeddings of two triconnected planar graphs. Edge a_1 in G_1 is indistinguishable from edge b_1 in G_2 if and only if $a_1{}^r$ is indistinguishable from $b_1{}^r$.*

Proof. Since we can substitute $a_1{}^r$ and $b_1{}^r$ for a_1 and b_1 in the statement of the lemma, we need only show that if a_1 is distinguishable from b_1, then $a_1{}^r$ is distinguishable from $b_1{}^r$. The proof is by induction on the shortest primary path $a_1, a_2, \ldots, a_k$ such that the corresponding path $b_1, b_2, \ldots, b_k$ has the property that $\lambda(a_k)$ does not equal $\lambda(b_k)$. For k equal to one the result is immediate. Assume the result true for all values of k less than n. Let $a_1, a_2, \ldots, a_n$ be the shortest primary path such that the corresponding path $b_1, b_2, \ldots, b_n$ starting with b_1 has $\lambda(b_n)$ not equal to $\lambda(a_n)$. Without loss of generality assume that a_2 is to the immediate right of a_1 in the planar embedding of G_1. Since a_2 is distinguishable from b_2 by a pair of primary paths of length $n - 1$, $a_2{}^r$ is distinguishable from $b_2{}^r$ by some pair of primary paths p_1 and p_2. Hence, the path starting with $a_1{}^r$ and consisting of the edges around the face to the left of $a_1{}^r$ followed by p_1 together with the corresponding path starting with $b_1{}^r$ distinguishes $a_1{}^r$ and $b_1{}^r$.

LEMMA 2. *Let G_1 and G_2 be planar embeddings of two triconnected planar graphs. Let $p_1 = a_1, a_2, \ldots, a_n$ be a path in G_1. Let b_1 be an edge of G_2 indistinguishable from a_1 and let $p_2 = b_1, b_2, \ldots, b_n$ be the path corresponding to p_1 starting with b_1. Then a_n and b_n are indistinguishable.*

Proof. It suffices to show that a_2 is indistinguishable from b_2. The result follows by induction. If a_2 is the edge to the immediate right or left of a_1 in the planar embedding of G_1, then the result is immediate. Otherwise let $x_1, x_2, \ldots, x_m$ be the edges directed into the head of a_1 in clockwise order as they appear in the planar embedding. Let $a_1 = x_1$ and $a_2 = x_i{}^r$. Let $y_1, y_2, \ldots, y_m$ be the corresponding edges about the head of b_1 where $b_1 = y_1$. Clearly $b_2 = y_i{}^r$ since p_1 and p_2 are corresponding paths. Consider the path consisting of edges a_1, the edges clockwise around the face to the right of x_2^r starting with x_2^r and ending with x_3, the edges clockwise around the face to the right of x_4^r starting with x_4^r and ending with x_5, etc., finally ending with edge x_i or x_i^r depending on whether i is odd or even. For each edge in this path the corresponding edge in the corresponding path starting with b_1 must be indistinguishable since for each edge in the path, the next edge is either to the immediate right or immediate left in the planar embedding. Thus, if the path ends with x_i^r the result is immediate. If the path ends with x_i an application of Lemma 1 gives the result.

LEMMA 3. *Let G be a biconnected planar graph. Let $(v_1, v_2), (v_2, v_3),..., (v_{n-1}, v_n)$ be a simple path in G. Then there exists a face having an edge in common with the path which has the property that the set of all edges common to both the face and the path form a continuous segment of the path. Furthermore, when traversing an edge of the face while going from v_1 to v_n along the path, the face will be on the right.*

Proof. If the set of edges common to some face and to the path consists of at least two discontinous sets of edges from the path, in both cases the face being on the right of the path, then all faces adjacent to the path from the right between the two sets of edges are adjacent only on the right. Select one such face. Either it satisfies the conditions of the lemma or its edges intersect the path in at least two discontinous sets of edges. By repeating the process of selecting a face eventually a face satisfying the lemma is selected.

Thus, assume that every face which is adjacent to the path on the right is also adjacent to the path on the left. No face can be adjacent to the path on both the right and the left at the same edge since the graph is biconnected. Select a face. Assume that the edge closest to v_n at which the face is adjacent on the right is closer to v_1 than the edge closest to v_n at which the face is adjacent on the left. Then each succeeding face adjacent to the path on the right towards v_n must have the same property. But the face adjacent to (v_{n-1}, v_n) on the right cannot have this property. This is a contradiction. Thus, there exists a face satisfying the conditions of the lemma.

LEMMA 4. *Let G_1 and G_2 be planar embeddings of two triconnected planar graphs. Let e_1 and e_2 be edges of G_1 and G_2, respectively, which are indistinguishable. Let p be a path starting with edge e_1. Then p is a cycle if and only if the corresponding path starting with edge e_2 is a cycle.*

Proof. Assume the lemma to be false. Then there exist edges e_1 and e_2 such that (1) e_1 and e_2 are indistinguishable, (2) p_1 starts with e_1 and forms a cycle (which without loss of generality we may assume proceeds clockwise in the plane), and (3) the path p_2 starting with e_2 and corresponding to p_1 is not a cycle. We may further assume that (4) p_2 is simple, since if p_2 is not simple we may choose a proper subpath of p_2 which is a cycle, and the corresponding part of p_1 will be simple. Last, we may assume that (5) if e_3 is an edge on p_1 or its interior with one endpoint on p_1, e_4 is the corresponding edge in G_2, p_3 is a cycle starting with e_3 which lies on p_1 or its interior, and p_4 is the corresponding path starting with e_4, then p_4 is not a simple path. If (5) does not hold, we may replace p_1 and p_2 by p_3 and p_4, and repeat the process until paths p_1 and p_2 which satisfy (1)–(5) are found.

Let $a_1 ,..., a_n$ be the edges of p_1 and let $a_1',..., a_n'$ be the corresponding edges of p_2. By Lemma 3 there exists a face f_2 having an edge in common with p_2 such that all edges of f_2 which are common to p_2 form a continuous segment of p_2, and f_2 is on the right of p_2. Let $b_1',..., b_m'$ be the edges of f_2 not on p_2, and let $b_1 ,..., b_m$ be the corresponding edges in G_1. If, for some $i < m$, b_i terminates on p_1, then condition (5) is violated. Thus, assume that for no $i < m$ does b_i terminate on p_i. Since a_i and a_i' are indistinguishable and since $a_i',..., a_j'$, $b_1',..., b_m'$ is a face (for suitable i and j), it follows that b_m must terminate on p_1 and $a_i ,..., a_j$, $b_1 ,..., b_m$ is a cycle. This implies that $a_1 ,... a_i$, $b_m ,... b_1$, $a_{j+1} ,..., a_n$ is a cycle and the corresponding path starting with a_1' is simple, contradicting assumption (5).

Theorem 5. *Let G_1 and G_2 be planar embeddings of two triconnected planar graphs. Let e_1 and e_2 be edges of G_1 and G_2, respectively. There exists an isomorphism of G_1 onto G_2 mapping e_1 onto e_2 if and only if edges e_1 and e_2 are indistinguishable.*

Proof. The only if portion is obvious. Namely, if there exists an isomorphism of G_1 onto G_2 mapping e_1 onto e_2, then it is easily seen that e_1 and e_2 are indistinguishable. The if portion is somewhat more difficult. Assume that edges e_1 and e_2 are indistinguishable. Construct an isomorphism as follows. Identify edge e_1 with e_2. Whenever two edges are identified, automatically identify their reversals and their corresponding heads and tails. Identify the edges of the face to the right of e_1 with the corresponding edges of the face to the right of e_2. Both faces have the same number of edges since $\lambda(e_1)$ equals $\lambda(e_2)$. It follows from the fact that e_1 and e_2 are indistinguishable that no conflict can occur in these identifications.

So far, the isomorphism maps a simple closed path c_1 and its interior to a simple closed path c_2 and its interior. By Lemma 3 there exists a face f_1 on the exterior of the cycle c_1, having an edge in common with c_1, such that all edges and vertices of the face which are in common with the cycle form a continuous segment of the cycle. Let e_1 be an edge common to f_1 and c_1. Let e_2 be the corresponding edge in c_2. Let f_2 be the face on the exterior of the cycle c_2 with edge e_2. Identify edges around f_1 with the corresponding edges around f_2. The isomorphism is, thus, extended to the interior of simple closed paths c_3 and c_4 containing c_1 and c_2, respectively, unless a conflict arises. Repeating the process yields the desired isomorphism.

A conflict can arise in one of two ways: (1) an attempt is made to identify a vertex v_1 with a vertex v_2 where v_1 and v_2 are of different degree; or (2) an attempt is made to identify a vertex v_1 with a vertex v_2 which has already been identified with some $v_3 \neq v_1$. We now show that both situations are impossible.

Let f_1 and f_2 be the faces whose edges are being identified when the conflict arises. Let c_1 and c_2 be the simple closed paths previously constructed and let e_1 and e_2 be the corresponding edges with e_1 common to c_1 and f_1 and e_2 common to c_2 and f_2. Since e_1 and e_2 are indistinguishable, corresponding edges around f_1 and f_2 must be indistinguishable, and, hence, corresponding vertices must be of the same degree. Thus, we need only consider the case where an attempt is made to identify a vertex v_1 with a vertex v_2 which has already been identified with some $v_3 \neq v_1$. Let $a_1, \ldots, a_n$ be the edges of c_1 and $a_1', \ldots, a_n'$ be the corresponding edges of c_2. Let $a_1, \ldots, a_p$, $b_1, \ldots, b_m$ be the edges of f_1. Let $a_1, \ldots, a_p$, $b_1, \ldots, b_i$ be the edges of f_1 which have been identified at the time a conflict arises and let $a_1', \ldots, a_p'$, $b_1', \ldots, b_i'$ be the corresponding edges of f_2. If the edge b_i terminates at a vertex on c_1, then $i = m$ and $a_1, \ldots, a_p$, $b_1, \ldots, b_i$ is the entire face f_1. It follows that $a_1', \ldots, a_p'$, $b_1', \ldots, b_i'$ must be the face f_2 since f_1 and f_2 have the same number of edges, and the assumed conflict is impossible. Thus, b_i does not terminate at a vertex on c_1, implying that b_i' terminates at a vertex on c_2. For some r, $a_r', \ldots, a_n'$, $a_1', \ldots, a_p'$, $b_1', \ldots, b_i'$ is a cycle in G_2. The corresponding path, $a_r, \ldots, a_n$, $a_1, \ldots, a_p$, $b_1, \ldots, b_i$, is a simple path in G_1. But this is impossible by Lemma 4. Thus, a conflict is impossible, and the theorem is proved.

Isomorphism Algorithm

Theorem 5 gives a simple condition which is both necessary and sufficient for two planar embeddings G_1 and G_2 to be isomorphic. Namely there must exist an edge e_1 in G_1 which is indistinguishable from some edge e_2 in G_2. Since a triconnected planar graph has an embedding in the plane which is unique up to a left–right orientation, we construct an embedding of G_1 and two embeddings of G_2, one for each of the two possible orientations. These embeddings can be constructed [5, 8] in time proportional to the number of edges in G_1 and G_2. Since G_1 and G_2 are assumed to be planar the number of edges is bounded by three times the number of vertices. The edges of the embedded graphs are partitioned into classes such that two edges are in the same class if and only if they are indistinguishable. In an earlier version of this work [4] the indistinguishability classes were obtained by constructing finite automata whose states correspond to the edges of the graphs. An algorithm [3] for partitioning the states of a finite automata was applied. However, the full power of the partitioning algorithm for finite automata is not needed since not all finite automata can arise. A simpler partitioning algorithm is given here.

Let G be a planar embedding of a graph whose connected components are triconnected. For each edge e in G let $f(e, R)$ and $f(e, L)$ be the edges out of the head of e which are to the immediate right and left of e, respectively, in the planar embedding. Partition the edges into blocks so that two edges are in the same block if and only if they have the same value of λ. Let $B(1)$, $B(2), \ldots, B(n)$ be the blocks of the partition. The following algorithm will partition the edges of G into classes so that two edges will be in the same class if and only if they are distinguishable.

Partitioning Algorithm

```
begin
A:  for i := 1 until n do
        place (i, R) and (i, L) in set PROCESS;
    while PROCESS ¬= ∅ do
      begin
C:      select and delete (i, D) from PROCESS;
G:      set MOVE = ∅;
H:      for e ∈ B(i) do place f(e, D) in set MOVE;
I:      for e ∈ MOVE do
          begin
            let B(j) be block containing e;
J:          if |B(j) ∩ MOVE| ¬= |B(j)| then
              begin
                if B(j') not yet created then create B(j');
                delete e from B(j);
                add e to B(j');
              end;
          end;
K:      for each B(j') created in previous statement do
          for D = L, R do
            if (j, D) in PROCESS then add (j', D) to PROCESS
            else if |B(j')| ≤ |B(j)| then add (j', D) else add (j, D);
      end;
end.
```

Initially the edges are partitioned according to the value of λ. Statement A places each pair consisting of a block index and a symbol R or L in the set PROCESS. Statement C selects and deletes a pair (i, D) from the set PROCESS. Assume the value of D is R. Statements G, H and I partition each block j into blocks with indices j and j' so that no edge in block j is to the immediate right of an edge of block i in the planar embedding and every edge of block j' is to the immediate right of an edge of block i. If either block j or j' would be null, then the split does not take place (statement J). If block j is split into blocks j and j', then for D equal L and R if the pair (j, D) is already in the set PROCESS, (j', D) is also added to the set PROCESS. If the pair (j, D) is not in the set PROCESS, then one of (j, D) or (j', D) is added to the set PROCESS, depending on whether block j is of smaller cardinality than block j' (statement K).

LEMMA 6. *Let* $|E|$ *be the number of edges in* G. *There exists a constant* k *so that the partitioning algorithm terminates in at most* $k|E|\log|E|$ *steps.*

Proof. A random access model of computation [1] is implicitly assumed. We further assume a computer representation for a set such that an element can be selected and deleted from a set in time independent of the size of the set; and a representation for a partition such that the block of the partition containing a given element can be found and an element moved from one block to another in time independent of the number of elements and the number of blocks.

If edge e is contained in a block whose index is removed from PROCESS for some value of D, the index of a block containing e can never again be placed in PROCESS with the same value of D until it has been split and the portion containing e is at most one half the previous size. Thus, a given edge can be in a block when its index is removed from PROCESS at most $2\log|E|$ times. Each time the body of the while statement is executed, the index of a block is removed from the set process. The time necessary to execute the body of the while statement is proportional to the number of edges in the block. Prorate the time equally to the edges in the block. Since an edge e can be in a block whose index is removed from the set PROCESS at most $2\log|E|$ times, the total time charged to any edge is bounded by a constant times $\log|E|$ and the total time of execution of the algorithm is bounded by a constant times $|E|\log|E|$.

LEMMA 7. *Upon termination of the partitioning algorithm two edges are in the same block of the partition if and only if they are indistinguishable.*

Proof. The partitioning is not too fine since initially edges are partitioned according to the value of λ. The algorithm then places edges e_1 and e_2 into different blocks only if $f(e_1^r, D)$ and $f(e_2^r, D)$ for D equal R or L are already in different blocks implying that e_1 and e_2 are distinguishable.

Initially all pairs consisting of an index of a block and a value of D are placed in the set PROCESS. Whenever a pair (i, D) for D equal R (L) is removed from PROCESS, the algorithm partitions all blocks so that,

(*) for each j, either every edge in block j is to the immediate right (left) of an edge of block i in the planar embedding or no edge in block j is to the immediate right (left) of an edge of block i.

If block i is ever split into blocks i and i', then (i, R), (i', R), (i, L) and (i', L) are added to the set PROCESS. However, if i already satisfied property * for D equal $R(L)$, then every block containing an edge to the immediate right (left) of an edge in block i or i'

contains only edges to the immediate right (left) of edges in block i and i'. Thus, insuring property * for either i or i' insures property * for both i and i'. Hence, only the index of the smaller block is placed in PROCESS. When the algorithm terminates, set PROCESS is empty and every block satisfies property *. But property * for all blocks implies that any two edges in the same block are indistinguishable.

THEOREM 8. *Two triconnected planar graphs can be tested for isomorphism in time proportional to* $|V| \log |V|$ *where* $|V|$ *is the number of vertices in the graphs.*

Proof. Immediate from the preceding results.

Although the isomorphism algorithm as described here applies only to triconnected planar graphs it may be extended to arbitrary planar graphs [6, 7]. It is still unknown whether a linear-time algorithm exists for determining the isomorphism of planar graphs.

REFERENCES

1. S. A. COOK, Linear time simulation of deterministic two-way pushdown automata, IFIP Congress 71: Foundations of Information Processing, Proceedings, Ljubljana, Yugoslavia, August 1971; pp. 174–179, North Holland Publishing Co., Amsterdam, 1972.
2. D. G. CORNEIL AND C. C. GOTLIEB, An efficient algorithm for graph isomorphism, *J. ACM* **17** (1970), 51–64.
3. J. E. HOPCROFT, An $n \log n$ algorithm for minimizing states in a finite automation, *in* "Theory of Machines and Computations" (Z. Kohavi, Ed.), Academic Press, New York, 1972.
4. J. E. HOPCROFT, "An $n \log n$ Algorithm for Isomorphism of Planar Triply Connected Graphs," Technical Report STAN-CS-71-192, Computer Science Department, Stanford University, Stanford, CA, 1971.
5. J. E. HOPCROFT AND R. E. TARJAN, "Efficient Planarity Testing," Technical Report 72-125, Computer Science Department, Cornell University, Ithaca, NY, 1972.
6. J. E. HOPCROFT AND R. E. TARJAN, Isomorphism of planar graphs (working paper), *in* "Complexity of Computer Computations" (R. E. Miller and J. W. Thatcher, Eds.), Plenum Press, New York, 1972.
7. J. E. HOPCROFT AND R. E. TARJAN, A V^2 algorithm for determining isomorphism of planar graphs, *Inform. Process. Lett.* 1 (1971), 32–34.
8. R. TARJAN, "An Efficient Planarity Algorithm," TR STAN-CS-244-71, Computer Science Department, Stanford University, Stanford, CA, 1971.
9. L. WEINBERG, A simple and efficient algorithm for determining isomorphism of planar triply connected graphs, *IEEE Trans. on Circuit Theory* **13** (1966), 142–148.
10. H. WHITNEY, A set of topological invariants for graphs, *Amer. J. Math.* **55** (1933), 231–235.

Printed in Belgium