SIAM J. COMPUT.
Vol. 4, No. 3, September 1975

# FINDING A MAXIMUM CUT OF A PLANAR GRAPH IN POLYNOMIAL TIME*

F. HADLOCK†

**Abstract.** The problem of finding a maximum cut of an arbitrary graph is one of a list of 21 combinatorial problems (Karp–Cook list). It is unknown whether or not there exist algorithms operating in polynomial bounded time for any of these problems. It has been shown that existence for one implies existence for all. In this paper we deal with a special case of the maximum cut problem. By requiring the graph to be planar, it is shown the problem can be translated into a maximum weighted matching problem for which there exists a polynomial bounded algorithm.

**Key words.** maximum cut, planar graph, geometric dual, polynomial time

**1. Introduction.** In this paper, it is shown that the maximum cut problem can be translated into the maximum weighted matching problem when the graph under consideration is planar. For an arbitrary graph, several algorithms exist for finding a maximum cut [4] and [5]. Both require exponential time in worst-case situations. Since the maximum weighted matching problem has a polynomial bounded algorithm [1], [2], a maximum cut of a planar graph can be found in polynomial time by using the translation process to be presented.

**2. Maximum cuts and odd circuits.** An edge set $D$ whose removal leaves a subgraph free of odd circuits will be called an *odd-circuit cover*. The purpose of this section is to obtain an alternative formulation for the problem of finding a maximum cut. First the relationship between cuts and odd-circuit covers is established.

THEOREM 1. *An edge set is contained in a cut if and only if its complement is an odd-circuit cover.*

*Proof.* Let $Q$ be an edge set contained in a cut $C$. The intersection of any circuit with $C$ is even and so $\sim Q$ must contain an edge of any odd circuit. Hence $\sim Q$ is an odd-circuit cover.

Conversely, if $\sim Q$ is an odd-circuit cover, its removal leaves a graph free of odd circuits and hence bipartite. Thus $Q$ is contained in a cut.

As a consequence of Theorem 1, an alternative to looking for maximum cuts is to look for minimum odd-circuit covers. This is justified by the following corollary, which follows immediately from Theorem 1.

COROLLARY 1. *An edge set is a maximum cut if and only if its complement is a minimum odd-circuit cover.*

The following fact means we can confine our attention to a circuit basis rather than looking at the entire space in constructing an odd-circuit cover. Since a graph is bipartite if and only if its circuit space has an even basis [6], an edge set $D$ is an odd-circuit cover if and only if its removal leaves a subgraph with an even basis. The term even *basis* refers to a circuit basis in which every element is an even circuit.

**3. Odd-circuit covers and odd-vertex pairings.** To obtain a maximum cut of a planar graph $G$, we suppose some embedding and take as a basis the contours of the finite faces. It is more convenient to work with the geometric dual, $G_D$, of $G$, where

* Received by the editors November 26, 1973, and in revised form July 24, 1974.

† Department of Mathematics, Florida Atlantic University, Boca Raton, Florida 33432.

the odd basis elements (along with the contour of the infinite face, if odd) become precisely the set of odd vertices. An edge $e$ in $G$ corresponds to an edge $e'$ in $G_D$ if and only if the two faces separated by $e$ in the embedding of $G$ correspond to the endpoints of $e'$ in $G_D$.

An edge set whose removal leaves a subgraph free of odd vertices will be called an *odd-vertex pairing*. Thus a subgraph with an *odd-vertex pairing* as edge set has an Euler subgraph as complement. The following theorem establishes a correspondence between odd-circuit covers and odd-vertex pairings.

THEOREM 2. *An edge set $D$ is an odd-circuit cover of a planar graph $G$ if and only if the corresponding edge set $P$ is an odd-vertex pairing for the geometric dual $G_D$ of $G$.*

*Proof.* Let $G_D$ be the geometric dual of $G$ for some embedding of $G$, with $D$ and $P$ corresponding edge sets of $G$ and $G_D$, respectively. Let $G'$ and $G'_D$ be the subgraphs of $G$ and $G_D$ left by the removal of $D$ and $P$. Circuits of $G$ correspond by the 1–1 edge correspondence to cut-sets of $G_D$. This is also true for $G'$ and $G'_D$. In particular, circuit basis elements of $G'$ correspond to cut-set basis elements of $G'_D$ as follows. A circuit basis element of $G$ is the contour of a finite face. Its edges correspond in 1–1 fashion with the edges of $G_D$ which are incident with the vertex representing that face. The set of edges incident with the vertex is a cut-set basis element.

If $D$ is an odd-circuit cover, the circuit basis for $G'$ is even. Since the edge correspondence is 1–1, the cut-set basis of $G'_D$ is even. Consequently the degree must be even for any vertex of $G'_D$ corresponding to a finite face of $G'$. The vertex corresponding to the infinite face cannot be the sole odd vertex. Hence $P$ is an odd-vertex pairing.

The converse follows by a similar argument.

To find a maximum cut of a planar graph, it suffices to find a minimum odd-vertex pairing of its dual. The following theorem gives a useful characterization of odd-vertex pairings.

THEOREM 3. *For an edge set $P$ of an arbitrary multigraph $G$, $P$ is a minimum odd-vertex pairing if and only if $P$ forms an edge disjoint collection of paths with odd vertices of $G$ as endpoints, using each once as endpoint, and with minimum sum of path lengths.*

*Proof.* Let $P$ be a minimum odd-vertex pairing for a multigraph $G$. The parity of a vertex refers to its degree: odd or even. If $H$ is the subgraph left by the removal of $P$, since $H$ is an Euler subgraph, a vertex has the same parity in $P$ as in $G$. In any component of a graph, there must be an even number of odd vertices; hence any odd vertex in $P$ is connected to another. Remove a path connecting a pair of odd vertices from both $P$ and $G$ to obtain subgraphs $P_1$ and $G_1$. $P_1$ is an odd-vertex pairing for $G_1$ since its removal leaves $H$. Any vertex has the same parity in $P_1$ as $G_1$. In going from $P$ to $P_1$, the number of odd vertices has been reduced by two. Repeating the process eventually yields an odd-vertex pairing $P_i$ with no odd vertices for a multigraph $G_i$. Since any vertex has the same parity in $G_i$ as $P_i$, $G_i$ is Euler. Since $P$ was assumed to be minimal, $G_i = H$ and $P_i = (V, \varnothing)$ (i.e., no edges). Then $P$ is the disjoint collection of paths with the odd vertices of $G$ as endpoints, using each once as endpoint. The sum of the path lengths is minimum since $P$ is minimum.

Now suppose $P$ is a collection of edge-disjoint paths with odd vertices as endpoints, using each endpoint once as endpoint, and with minimum sum of path lengths. Remove $P$ from $G$, one path at a time. Denote by $H$ the subgraph remaining after $P$ has been removed. The removal of each path leaves the endpoints even and does not alter the parity of intermediate vertices. Since any vertex odd in $G$ appears once as an endpoint, it is even in $H$, and so $P$ is an odd-vertex pairing. $P$ is minimum since the sum of path lengths is minimum.

**4. Odd-vertex pairings.** The task of pairing odd vertices so as to minimize the sum of the lengths of the paths pairing them is easily posed as a maximum matching problem as observed in [3].

Given a multigraph $G$, a minimum odd-vertex pairing $P$ for $G$ is obtained as follows. Let $G_c$ be the complete graph with vertices corresponding to the odd vertices of $G$. With each edge $e = (u, v)$, associate the weight $W - d(u, v)$ where $d(u, v)$ is the length of the minimum length path connecting $u$ and $v$ and $W = 1 + \max \{d(u, v) | u, v \text{ odd in } G\}$. Let $M$ be a maximum matching of $G_c$. Then M defines a minimum odd vertex pairing as follows. For each edge $e = (u, v)$ in $M$, include in $P$ the edges of *any* minimum length path connecting $u$ and $v$ in $G$.

The problem is now in a form for which there exists an algorithm [2] for its solution. It is an algorithm which is good in the sense that the amount of time it requires is a polynomial function of an input parameter (the number of vertices in this case).

**5. An example.** To illustrate the process of translating a solution to the maximum matching problem into a solution to the maximum cut problem, we use an example (Fig. 1) in which the matching problem is solved by inspection. A minimum odd-circuit cover may be found by determining a minimum odd-vertex pairing of the geometric dual $G_D$ (Fig. 2). In turn, this may be found by determining a maximum matching for the complete graph on the odd vertices of $G_D$ (Fig. 3). Since the weights here are 1 or 2, any complete matching with all edge weights 2 is a maximum matching (Fig. 4). A minimum odd-vertex pairing for $G_D$ is obtained by taking a minimum path connecting $u$ and $v$ for each edge $(u, v)$ in the maximum matching. In this case each minimum path is single edge $(u, v)$.

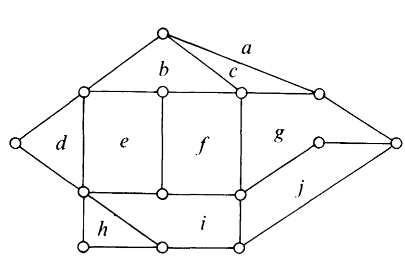


FIG. 1. *Planar graph G*

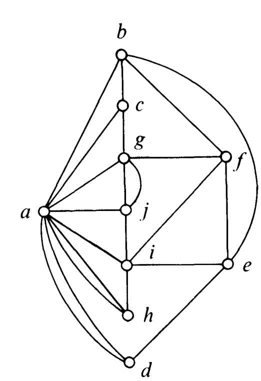


FIG. 2. *Dual of G, $G_D$*

| | $a$ | $c$ | $d$ | $g$ | $h$ | $i$ |
|---|---|---|---|---|---|---|
| $a$ | | 2 | 2 | 2 | 2 | 2 |
| $c$ | | | 1 | 2 | 1 | 1 |
| $d$ | | | | 1 | 1 | 1 |
| $g$ | | | | | 1 | 1 |
| $h$ | | | | | | 2 |
| $i$ | | | | | | |

FIG. 3. *Edge weights for complete graph $G_c$ on odd vertices of $G_D$*

$$M = \{(a, d), (c, g), (h, i)\}$$

FIG. 4. *Maximum matching for $G_c$*

$$P = \{(a, d), (c, g), (h, i)\}$$

FIG. 5. *Minimum odd-vertex pairing for $G_D$*

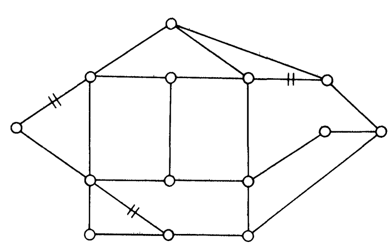

FIG. 6. *Minimum odd circuit cover for G*

Their collection forms an odd-vertex pairing (Fig. 5). The corresponding minimum odd-circuit cover for $G$ consists of the marked edges (Fig. 6). Its complement is a maximum cut.

**6. Conclusions.** Finding a maximum cut of a planar graph is a special case, as remarked earlier, of a problem on a list [8] of combinatorial optimization problems, including the traveling salesman problem and the problem of vertex coloring a graph with the fewest number of colors. If any of these problems have a polynomial bounded algorithm, all do. In this paper, the existence of a polynomial bounded algorithm for a special case (planar graphs) of one of these problems may aid in defining special cases of the others for which polynomial bounded algorithms exist. At the same time, an attempt to extend this approach to the general case might lend insight as to why a polynomial bounded algorithm does *not* (as is widely believed) exist for the general case.

## REFERENCES

[1] J. EDMONDS, *Paths, trees, and flowers*, Canad. J. Math., 17 (1965), pp. 449–467.

[2] ———, *Maximum matching and a polyhedron with 0,1-vertices*, J. Res. Nat. Bur. Standards Sect. B, 69B (1965), pp. 125–130.

[3] S. GOODMAN AND S. HEDETNIEMI, *Eulerian walks in graphs*, this Journal, 2 (1973), pp. 16–27.

[4] F. HADLOCK, *The minimal 2-coloration problem*, Proc. 3rd Southeastern Conf. on Combinatorics, Graph Theory, and Computing, F. Hoffman, R. B. Levow, R. S. D. Thomas, eds., Utilitas Mathematica, Winnipeg, Canada, 1972, pp. 221–241.

[5] ———, *Optimal graph partitions*, Proc. 4th Southeastern Conf. on Combinatorics, Graph Theory, and Computing, F. Hoffman, R. B. Levow, R. S. D. Thomas, eds., Utilitas Mathematica, Winnipeg, Canada, 1973, pp. 309–328.

[6] F. HARARY, *Graph Theory*, Addison-Wesley, Reading, Mass., 1969.

[7] S. JOHNSON, *Approximation algorithms for combinatorial problems*, Proc. 5th Annual ACM Symp. on Theory of Computing, Austin, Tex., 1973, pp. 38–49.

[8] R. KARP, *Reducibility among combinatorial problems*, Complexity of Computer Computations, R. E. Miller and J. W. Thatcher, eds., Plenum Press, New York, 1972, pp. 85–103.