COVERING ORTHOGONAL POLYGONS WITH SQUARES

LARRY J. AUPPERLE
Dept. of Computer Science, Princeton U., Princeton, NJ 08544
HAROLD E. CONN
Dept. of Computer Science, Johns Hopkins U., Baltimore, MD 21218
J. MARK KEIL
Dept. of Computational Sci., U. of Saskatchewan, Saskatoon, Canada S7N 0W0
JOSEPH O'ROURKE
Dept. of Computer Science, Smith College, Northampton, MA 01063

ABSTRACT

The problem of covering an orthogonal polygon with a minimum number of squares is shown to be solvable in polynomial time for simply-connected polygons, making this one of the few minimum covering problems known to be tractable. The polynomial algorithm is based on establishing that a particular graph associated with the problem is chordal, a subclass of perfect graphs for which efficient clique covering algorithms are known. For polygons with holes, the minimum covering problem is shown to be NP-complete. The square cover problem finds application in image processing, where maximal squares are used to compress pictures into efficient data structures.

1. INTRODUCTION

We consider the problem of finding a minimum square cover of an orthogonal polygon. An *orthogonal polygon* is a polygon having all its edges parallel to either of two given orthogonal directions. It may contain holes that are themselves orthogonal polygons. We restrict our consideration to orthogonal polygons having all vertices at integral coordinates. These orthogonal polygons are also referred to as *boards*. A unit square with vertices at integral coordinates is called a *block* *. A *square* is a square subset of the blocks of a given board. A *square cover* of a board is a collection of squares each of which is a subset of the board and such that the set union of the collection of squares is equal to the board.

In this paper we analyze the complexity of the minimum square cover problem first for simply connected boards and then for boards containing holes. Given a board B, we associate with it a graph $G(B)$ as follows. Each block in the board B corresponds to a node in the graph $G(B)$. There is an edge between two nodes in $G(B)$ iff the corresponding blocks in B belong to a square of B. We show that the graphs associated with simply-connected boards are chordal graphs. This is a strengthening of a result due to Albertson and O'Keefe. In [2] they showed that these graphs are perfect graphs. Chordal graphs form a subclass of perfect graphs, and using a known algorithm for finding a minimum clique cover of a chordal graph [4], we can find a minimum square cover of a board with no holes in polynomial time, $O(N^{2.5})$ for boards with N blocks.

In [1] Aupperle improves the complexity of the algorithm to $O(rc \; min(r,c))$, where r and c are the number of rows and columns of blocks in the board. This complexity is $O(N^{1.5})$ in the worst case.

When a board is allowed to contain holes, we show that the problem of finding a minimum square cover is NP-complete.

This problem is interesting for two reasons, one theoretical and one practical. On the theoretical side, there has been considerable interest in geometric minimum covering problems, but unfortunately most of the most natural problems are NP-complete [7]. Just recently it was shown by Culberson and Reckhow [3] that one of the more interesting unsettled cases, minimum covers of simple orthogonal polygons by rectangles, is also NP-complete. Thus minimum coverage by squares turns out to be one of the few of these covering problems that can be solved in polynomial time.

Coverage by squares also has practical applications. The *medial axis* (also called the *symmetric axis* or *skeleton*) of a polygon is the locus of centers of maximal disks contained in the polygon. When

* For image processing applications, a board is a picture or an image, and a block is a pixel.

specialized to the L_∞ metric for applications to digital images, the medial axis is the locus of centers of maximal squares of odd side length [8]. The digital medial axis transform (*MAT*) is used for picture compression: simple images may be covered by few squares, and easily reconstructed from the *MAT* [10]. The image processing literature has assumed that the NP-completeness results for coverage by rectangles obviated any fast algorithm for coverage by squares [10]; thus no attempt was made to find *minimum* covers. Our results show that in the important special case of a connected image region, pessimism was unwarranted: a minimum cover can be found quickly.

Additionally, Scott and Iyengar [9] have defined the *TID*, Translation Invariant Data Structure, a method for representing images. The *TID* for a given image consists of a list of maximal squares covering all pixels within the image, sorted by certain position and size criteria. In order to reduce the cost of storing and manipulating a *TID*, the underlying list of maximal squares should be as small as possible.

2. COVERING A SIMPLY-CONNECTED BOARD

In this section we prove that if B is a simply connected board, then $G(B)$ is a chordal graph. A graph is said to be *chordal* if it has the property that every (simple) cycle of length strictly greater than 3 possesses a chord. Before proceeding with the proof, some definitions and a lemma from [2] are needed. Given a board B and a square S in that board, S is said to be *maximal* if it is not contained in any larger square of the board B. A maximal square S is said to be *disconnecting* if the interior of $B-S$ is not connected. A block of B is a *boundary* block if it intersects the boundary of B in at least one edge. A block in a square S is a *border* block of S if either: (a) it shares an edge with a block of $B-S$, or (b) it is a boundary block. A $1\times p$ rectangular subset of the blocks of a given board having three sides coincident with the boundary of the board is called a *knob*.

Lemma 1: [2] Let S be a maximal square in a simply-connected board B. Then either S is a disconnecting square or S contains a knob of B.

Theorem 1: Let B be a simply-connected board. Then the associated graph $G(B)$ is chordal.

Proof: By induction on the number of blocks in a board. As the induction hypothesis, assume the claim is true for simply-connected boards having fewer than m blocks. Let B be a simply-connected board having m blocks. Let S be a maximal square in B, and let $C = \{c_1, c_2, \cdots, c_n\}, n \geq 4$, be the set of blocks in B corresponding to the nodes of a simple cycle of length at least 4 in $G(B)$. By Lemma 1, there are two cases to consider.

Case I: S is disconnecting. Let $B_1, \ldots, B_j$, $j \geq 2$, be the components of $B-S$.

Observation 1: Two blocks are in a square of B iff they are in a square of $S \cup B_i$ for some $i \leq j$. (Because S is disconnecting.)

Suppose the blocks of C are all contained in $S \cup B_i$ for some i. Then by Observation 1, the subgraphs induced by C in $G(B)$ and in $G(S \cup B_i)$ are identical. Since the board $S \cup B_i$ has fewer than m blocks and is simply-connected, apply the induction hypothesis and conclude that C has a chord in $G(S \cup B_i)$, and thus also in $G(B)$.

Otherwise, suppose that some blocks of C are in different components of $B-S$, say B_p and B_q. By Observation 1, any path from B_p to B_q must pass through a block in the square S. Since C is a cycle, there are two distinct paths from any of its blocks in B_p to any of its blocks in B_q. Since C is vertex-simple, there must be two distinct blocks of C in S. The nodes in $G(B)$ corresponding to these two blocks in S are connected by an edge, and thus the cycle must have a chord.

Case II: S is not disconnecting. Then, by Lemma 1, S contains a knob of B. Call the knob K.

Observation 2: Any block in a knob contained in S can only be connected (in $G(B)$) to some other block in S.

If some block in C, call it c_i, is contained in any knob in S, then c_i is connected to two other blocks in S. These blocks are connected, so C has a chord.

Suppose no block of C is contained in any knob in the square S. Assume that K is a $1\times p$ knob in a $p\times p$ square S, and that the board is oriented as shown in Figure 1. Then there are two cases to consider: either (A) each block in K has at least p blocks of B beneath it, or (B) some block of K, say b_1, has exactly $p-1$ blocks beneath it.

Case A: Each block in K has at least p blocks beneath it.

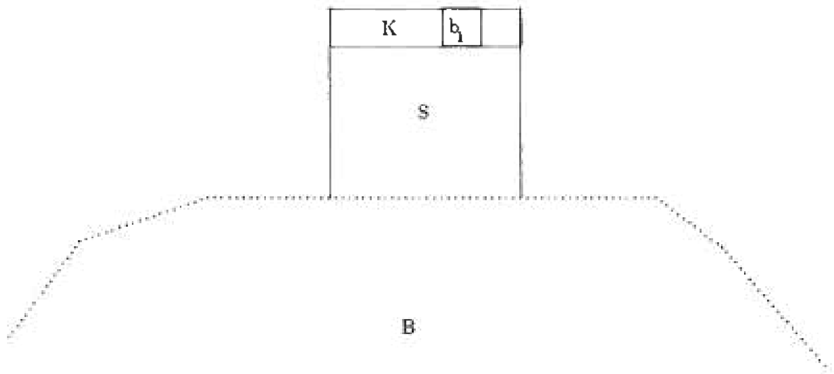

Figure 1

Consider the board $B-K$. By Observation 2, any block in K can only be connected to some other block in S. Thus, except for S, every square of B is also a square of $B-K$. So if two blocks of C are connected by a square of B other than S, they are connected by that same square in $B-K$.

Now suppose two blocks of C are connected by S in B. Since these blocks are not in K, they are connected by the $p \times p$ square consisting of rows 2 through p of S and the p blocks immediately below row p of S.

So two blocks of C that are not in K are in a square of B iff they are in a square of $B-K$. Thus the subgraphs induced by C in $G(B)$ and in $G(B-K)$ are identical. Since $B-K$ is simply connected and has fewer than m blocks, the induction hypothesis may be applied to conclude that C has a chord in $G(B-K)$ and thus also in $G(B)$.

Case B: Some block of K, call it b_1, has exactly $p-1$ blocks of B beneath it. The last of these must be a boundary block, call it b_2. See Figure 2.

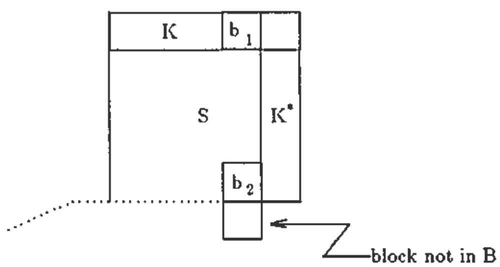

Figure 2

Since S is not disconnecting, one of the paths of border blocks of S from b_1 to b_2 must consist entirely of boundary blocks. Thus one of the vertical sides of S must be the boundary of another knob, call it K^*, as shown in Figure 2.

Let $L = K \cup K^*$. Consider the board $B-L$. Two blocks of C not in L are in a square of B iff they are in a square of $B-L$. The board $B-L$ is simply-connected and has fewer than m blocks, so the induction hypothesis may be applied to conclude that C has a chord in $G(B-L)$ and thus C also has a chord in $G(B)$. □

Using an algorithm due to Gavril [4], we can find a minimum clique cover of $G(B)$ in $O(N^{2.5})$ time, where N is the number of blocks in the board B. Then the following lemma due to Albertson and O'Keefe allows us to use the minimum clique cover of $G(B)$ to find a minimum square cover of B.

Lemma 2: [2] Given a clique in $G(B)$, the set of corresponding blocks in B is entirely contained in a single square of B.

The complexity of the algorithm can be improved by exploiting the geometric setting in which $G(B)$ is defined by employing a procedure due to Scott and Iyengar [9] for finding all maximal squares in B. In [1] Aupperle uses this approach to develop an algorithm for minimum square cover of a simply-connected board. The algorithm runs in time and space $O(rc\ min(r,c))$, where r and c are the number of rows and columns of blocks in the board. This complexity is $O(N^{1.5})$ in the worst case.

3. BOARDS WITH HOLES

In this section we show that the problem of finding a minimum square cover of an arbitrary board containing holes is an NP-complete problem. For a discussion of the theory of NP-completeness see [5].

Our proof uses a transformation from the NP-complete problem of Planar 3-Satisfiability (P3SAT). An instance of P3SAT is an instance of 3-Satisfiability with the additional property that a graph associated with the instance is a planar graph. The associated graph consists of a set of nodes corresponding to the variables and clauses, and with an edge between any variable and any clause in which that variable (or its negation) appears. See [6] for a more detailed discussion of P3SAT.

The transformation involves constructing an orthogonal polygon that simulates an arbitrary instance of P3SAT. The polygon must have the property that calculation of the size of a minimum square cover of the polygon will answer the question of satisfiability of the corresponding instance of P3SAT.

The polygon is built up from three types of components: variable loops, wires, and junctions. Variable loops allow the generation of an arbitrary number of copies of a variable and an arbitrary number of copies of the negation of that same variable. Wires allow the propagation of truth values from variable loops to junction figures. Junctions represent the logical AND of the three input values. The individual components are described in more detail below.

The orthogonal polygon is assembled from component pieces laid out on a grid of unit squares. For reference purposes when describing the component figures, it is useful to label alternating grid lines as "odd" and "even" in both the horizontal and vertical directions. All the squares needed for a minimum cover of the orthogonal polygon are 2×2 squares of 4 blocks.

3.1 Variable Loop Figures

Variable loop figures provide the capability of generating an arbitrary number of copies of a variable and an arbitrary number of copies of the negation of that same variable. Variable loop figures are positioned so that wires may be attached to them at "even"-labeled vertical grid lines. A variable loop figure may be arbitrarily high to accommodate as many copies of a variable or its negation as are required by the instance of P3SAT. Wires representing non-negated variables are attached to the variable loop in such a way that they are centered on "even"-labeled horizontal grid lines. Wires representing negated variables are centered on "odd"-labeled horizontal grid lines.

A variable loop figure has two minimum covers. In one of these, squares protrude out from the variable loop figure and into the attached wire figures that represent non-negated variables. This cover corresponds to the assignment of TRUE to that variable. In the other cover, squares are flush with wires representing non-negated copies of the variable, and protrude out into the wires representing negated copies of the variable. This cover corresponds to the assignment of FALSE to that variable.

Figure 3 illustrates these concepts regarding variable loops and their covers. In the figure, shaded squares indicate those that must be present in any minimum cover. Dotted lines represent "even"-labeled grid lines; dashed lines represent "odd"-labeled grid lines.

3.2 Wire Figures

A straight wire figure consists of a $2 \times n$ rectangle, where n is the number of unit squares traversed by the figure in a horizontal direction, and is always an even number. Wires are always attached to variable loops and junctions at "even"-labeled vertical grid lines. A wire is considered to

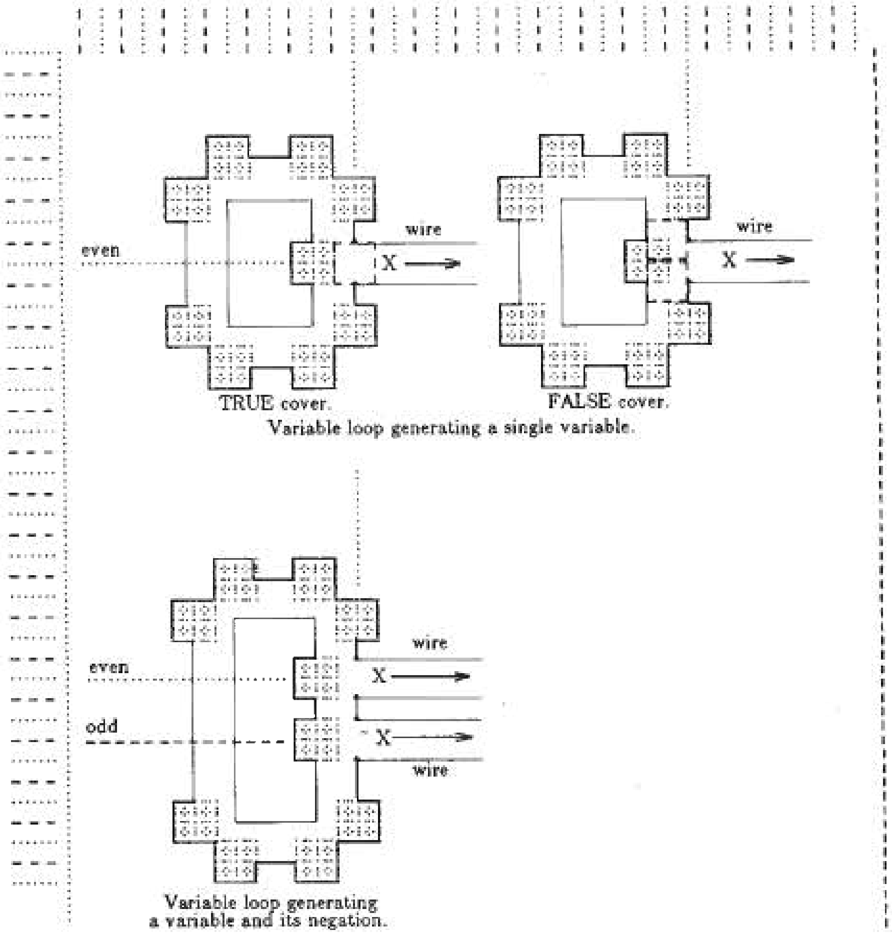

Figure 3

be carrying a TRUE value when the first 2×1 rectangle on its left hand side (*source*) is covered by a square that protrudes out from a variable loop. Consequently, in a minimum cover, the right hand side of the wire (*destination*) has half a square protruding out from it and into a junction figure.

Wires can be bent at 90° angles as shown in Figure 4. When a wire is bent, it may no longer possess the desired behavior with respect to minimum coverage by squares. We say that a wire figure is *valid* with respect to propagation of truth values if it has exactly two minimum covers: one with squares flush at both the source and destination of the wire, and the other with half a square protruding from the destination of the wire into the attached junction figure. (In the latter case, coverage of the first 2×1 rectangular area at the source of the wire is provided by a square that protrudes from the attached variable loop.) The following lemma guarantees that the wires used in constructing the orthogonal polygon are valid.

Lemma 3: If a wire figure has its source and destination centered on horizontal grid lines having like labels (both "even" or both "odd"), then that figure is valid with respect to propagation of truth values.

Proof: By induction on the number of bends in a wire. □

In the construction of the orthogonal polygon simulating an instance of P3SAT, wires will always be attached to variable loops and junctions in such a way that their sources and destinations are centered on horizontal grid lines having like labels.

Figure 4 illustrates these concepts regarding wire figures and their covers. In the figure, shaded squares indicate those that must be present in every minimum cover of the figure.

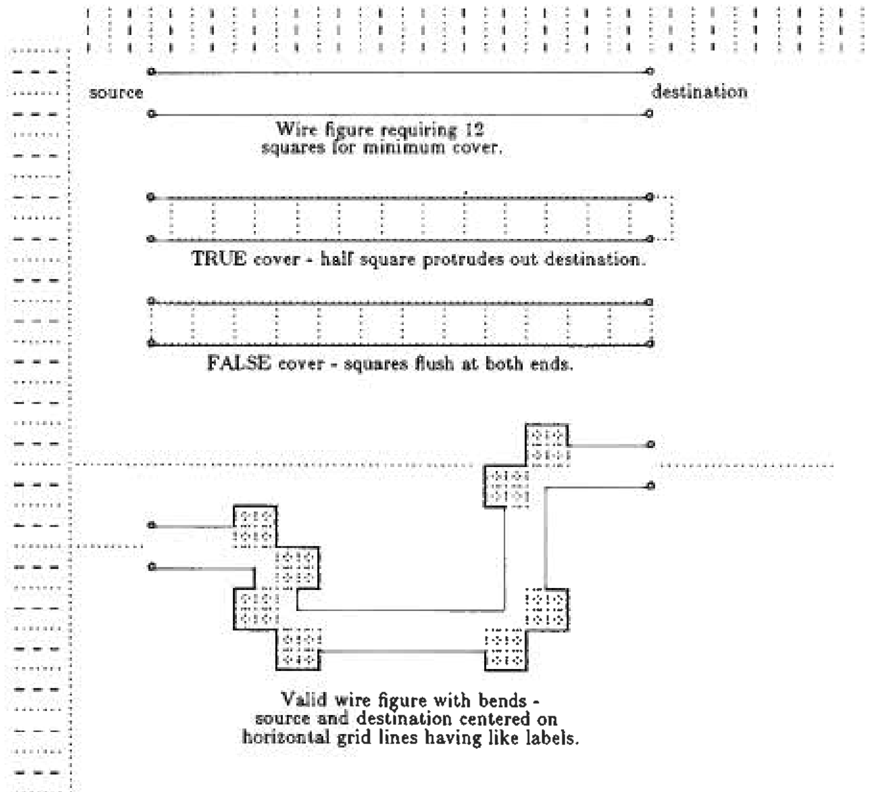

Figure 4

3.3 Junctions

Junction figures provide the capability of testing the value of the logical AND of the three input variables. They are positioned with their left boundary aligned on an "even"-labeled vertical grid line. Input wires representing non-negated variables are centered on "even"-labeled horizontal grid lines, and input wires representing negated variables are centered on "odd"-labeled horizontal grid lines. Two slightly different junction figures are used: one for clauses in which all the variables are either negated or non-negated, and one for clauses in which there are two negated variables and one non-negated or vice versa. Since the order of appearance of the literals (variable or negation of a variable) in a clause does not affect the truth value of the clause, assume that the odd literal appears first. Figure 5 displays the two types of junction figures and the various combinations of inputs.

Lemma 4: A junction figure can be covered with 12 squares iff all three input wires carry a value of TRUE . If any input wire carries a value of FALSE , then 13 squares are required to cover the junction.

Proof: Omitted.

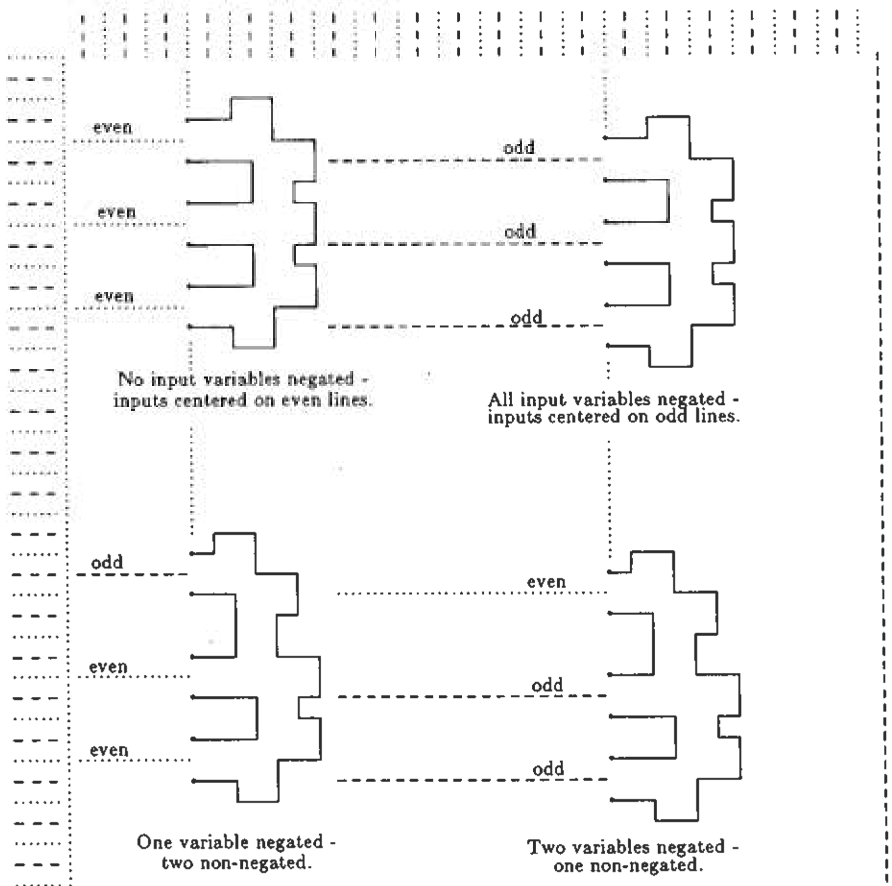

Figure 5

Figure 6 displays minimum covers of a junction figure for representative combinations of input values. Covers of the other type of junction figure and other combinations of inputs are similar.

An instance of P3SAT consists of the conjunction of a set of clauses, where each clause is the disjunction of three literals. In testing the satisfiability of such an instance, it would be useful to have a junction figure that simulates the logical OR of three input literals. Such a junction figure is presented in [1]. However, the AND junction figure described here is somewhat simpler, and can also be used to test the satisfiability of an instance by applying DeMorgan's law as follows. Negate each literal in the instance and let each OR connective in each clause be replaced by an AND connective. Now if every assignment of truth values to the variables in the instance results in every clause having value FALSE , then this is equivalent to each of the original clauses being made TRUE , and the instance is satisfiable.

3.4 Overview of Construction

Given an instance I of P3SAT, examine the literals in each clause. If any clause consists of one negated variable and two non-negated variables, rearrange the order of appearance of the literals (if necessary) so that the negated variable appears first in the clause. If any clause consists of two negated variables and one non-negated variable, rearrange the order so that the non-negated variable appears first. After this rearrangement, every clause has its literals in an order that is compatible

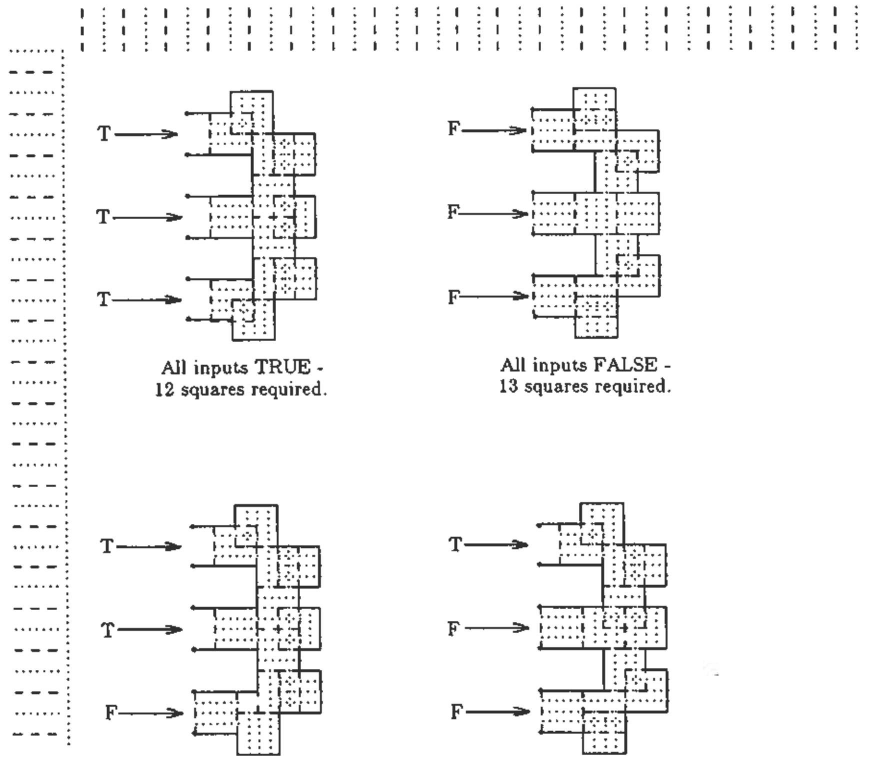

Figure 6

with one of the two junction types. The rearrangement process does not affect the satisfiability or the planarity of the instance of P3SAT.

Next apply DeMorgan's law as previously described to convert OR connectives into AND connectives. During this process all literals will be negated.

For every variable, scan the clauses counting all appearances of that variable and all appearances of its negation. Set up a variable loop with the required number of copies of the variable and its negation. Count the number of squares required to cover all such variable loops. Let this number be V.

Run wires from the variable loops to junction figures representing the clauses. Count the number of squares required to cover the wires. Let this number be W, and let the number of junction figures be J.

The preceding construction can be done in polynomial time. Now, find a minimum square cover of the orthogonal polygon just constructed. Let the number of squares in the minimum cover be C. If $C = V + W + 13*J$, then I is satisfiable. If $C < V + W + 13*J$, then I is not satisfiable.

Thus if the minimum square cover could be found in polynomial time, we would also have determined the satisfiability of I in polynomial time, so finding the minimum square cover must be NP-hard. Clearly the minimum square cover problem is in NP, so we have the following theorem:

Theorem 2: The problem "are there k squares that cover an arbitrary orthogonal polygon?" is NP-complete.

4. DISCUSSION

The algorithm for simply-connected boards is polynomial in the number of blocks in the board. An interesting open problem is to find an algorithm that is polynomial in the number of vertices in the board. The number of squares needed to cover a board can be arbitrarily high in relation to the number of vertices, as in an elongated rectangle. So any such algorithm would have to have some of the output squares encoded instead of explicitly output.

The results presented here can be extended to minimum coverage of boards by fixed aspect ratio rectangles of a given aspect ratio. A rectangle's aspect ratio is the ratio of its height to its width.

In [1] Aupperle considers coverage by fixed aspect ratio rectangles having side lengths that are an integral multiple of one block. He shows that for multiply-connected boards, minimum coverage is an NP-complete problem. For simply-connected boards, he shows that the graphs associated with the boards are not perfect, and thus not chordal.

Using rectangles of a fixed aspect ratio with side lengths an integral multiple of one block, there exist boards that cannot be covered. Another approach is to allow side lengths to be a rational multiple of one block. Then all boards can be covered, and both the results presented here extend: simply-connected boards have associated graphs that are chordal, and minimum coverage of multiply-connected boards is NP-complete. This can be seen by performing a transformation on the board to be covered. If the board is to be covered by rectangles with aspect ratio h/w, then increase the height of the board by a factor of w and the width of the board by a factor of h. Then the problem is reduced to one of coverage by squares.

5. ACKNOWLEDGEMENTS

The first author was partially supported by the Natural Sciences and Engineering Research Council of Canada. The second and fourth authors were partially supported by NSF grant DCR-83-51468.

REFERENCES

[1] L. J. Aupperle, "Covering Regions by Squares," Master's Thesis, University of Saskatcl Saskatoon, Canada, July 1987.

[2] M. O. Albertson and C. J. O'Keefe, "Covering regions with squares," *SIAM J. Alg. Disc.* vol. 2, 3, pp. 240-243, Sept. 1981.

[3] J. C. Culberson and R. A. Reckhow, "Covering Polygons is Hard Preliminary Abstract," Rept. TR 88-6, Department of Computing Science, University of Alberta, 1988.

[4] F. Gavril, "Algorithms for minimum coloring, maximum clique, minimum covering by and maximum independent set of a chordal graph," *SIAM J. Computing*, vol. 1, 2, pp. 1ℓ June 1972.

[5] M. R. Garey and D. S. Johnson, *COMPUTERS AND INTRACTABILITY A Guide Theory of NP-Completeness*, W. H. Freeman and Company, New York, 1979.

[6] D. Lichtenstein, "Planar formulae and their uses," *SIAM J. Comput.*, vol. 11, 2, pp. 3: May 1982.

[7] J. O'Rourke, *Art Gallery Theorems and Algorithms*, Oxford University Press, New York,

[8] A. Rosenfeld and A. C. Kak, *Digital Picture Processing*, Academic Press, New York, 198:

[9] D. S. Scott and S. S. Iyengar, "TID - a translation invariant data structure for storing in *Communications of the ACM*, vol. 29, no. 5, pp. 418-429, 1986.

[10] A. Y. Wu, S. K. Bhaskar, and A. Rosenfeld, "Computation of Geometric Properties fr Medial Axis Transform in $O(n \log n)$ Time," *Computer Vision, Graphics, and Image P ing*, v. 34, 76-92, 1986.