Approximating the Rectilinear Polygon Cover Problems (Extended Abstract)*

Piotr Berman† Bhaskar DasGupta‡

Abstract

The rectilinear polygon cover problem is one in which a certain class of features of a rectilinear polygon of n vertices has to be covered with the minimum number of rectangles included in the polygon. In particular, we consider covering the entire interior, the boundary and the set of corners of the polygon. Most of these problems are known to be NP-complete. In this extended abstract we show:

(a) The corner cover problem is NP-complete.

(b) The boundary and the corner cover problem can be approximated within a ratio of 4 of the optimum in $O(n \log n)$ and $O(n^{1.5})$ time, respectively.

(c) The corner cover problem for polygons without holes can be approximated within a ratio of 2 of the optimum in $O(n \log n)$ time.

(d) No polynomial-time approximation scheme exists for the interior and the boundary cover problems, unless $P = NP$.

1 Introduction.

The problem of covering a certain class of features of a rectilinear polygon with the minimum number of rectangles included inside the polygon belongs to a more general class of geometric covering and decomposition problems. Some of these problems are surveyed in [17]. Particular attention has been paid in covering the entire interior, the boundary and the corners of a given rectilinear polygon. This problem has important applications like storing images[13], and manufacture of integrated circuits[14].

Masek[13] was the first to show that the interior cover problem is NP-complete for rectilinear polygons with holes. Conn and O'Rourke[5] later showed that the boundary cover problem is also NP-complete for polygons with holes, even if the polygon is in general position. They also show that the corner cover problem is NP-complete if we require each concave corner to be covered by two rectangles along both the perimeter segments defining the corner. For a long time the complexity of the problem was unknown for polygons without holes, until Culberson and Reckhow[6] showed the interior and boundary cover problems are NP-complete even if the polygon has no holes and is in general position. The complexity of the corner cover problem for polygons without holes is still unknown, although it is conjectured to be NP-complete in [5].

Various special cases of the rectilinear cover problem is known to be solvable in polynomial time. Franblau and Kleitman[8] gave a polynomial time algorithm for covering vertically convex rectilinear polygons with rectangles, which improved a previous result of [4]. Lubiw[11][12] gave polynomial time algorithm for somewhat larger class of polygons, called the plaid polygons. Conn and O'Rourke[5] gave polynomial time algorithm for covering the convex corners of a rectilinear polygon or horizontal perimeter segments of a

*This research was partially supported by NSF Grant CCR-9114545

†Department of Computer Science, Pennsylvania State University, University Park , PA 16802, berman@cs.psu.edu

‡Department of Computer Science, Pennsylvania State University, University Park , PA 16802, bhaskar@cs.psu.edu

rectilinear polygon in general position. Investigation of special cases of this problem has also given rise to special kinds of perfect graphs of interest[15].

In contrast to the covering problem, the rectilinear decomposition problem (when no overlapping of rectangles are allowed) has a polynomial time solution for polygons without degenerate (point) holes[16][18]. However, the problem becomes NP-complete when degenerate holes are allowed[10].

In this extended abstract we assume that the reader is familiar with the standard terminologies and definitions, for example, as used in [4], [5], [6], [7], and [8]. See *fig.* 1 for illustration.

2 NP-completeness of the corner cover problem.

Conn and O'Rourke in [5] proved a similar problem to be NP-complete: for each corner adjacent portions of the boundary segments have to be covered by rectangles. We prove that the problem is NP-complete even if only the corners themselves have to be covered.

A careful examination of their component figures (*fig.* 4 of [5]) shows that the *wire*, *not*, *crossover*,*true*, *false* and *switchback* components satisfy our requirement. However, the *generator* and *and/or* components fail to satisfy our requirements. We modify these two components as shown in *fig.* 2.

Theorem 2.1 *The problem "are there k rectangles so that each corner of the polygon is covered ?" is NP-complete.*

3 Heuristics for various cover problems.

We assume, *wlog*, that the corners of the given polygon have integer coordinates.

3.1 General boundary cover problem.

The following heuristic achieves a performance ratio of 4.

Input : A rectilinear polygon P having n vertices, possibly with rectilinear holes.
Output: A set of rectangles covering the boundary of P.

Algorithm: For each edge (a, b) of P create an adjacent rectangle R of width 1. First, extend this rectangle lengthwise until it hits the boundary. Then, extend maximally the width of this rectangle (see *fig.* 3). The set of all such rectangles constitutes an approximate cover.

Lemma 3.1 *The performance ratio of the above heuristic is* 4. *It runs in* $O(n \log n)$ *time.*

Remark: The example where the heuristic attains a performance ratio of 4 is shown in *fig.* 4.

3.2 General corner cover problem.

The following heuristic guarantees a performance ratio of 4 for this type of cover.

Input: A rectilinear polygon P having n vertices, possibly with rectilinear holes.
Output: A set of rectangles covering the corners of P.

Algorithm: Form the two sets S and T; S contains, for each horizontal segment e of P, an adjacent rectangle of width 1 of maximal horizontal extent (this is the ***principal side*** for this rectangle), and T contains for each vertical segment e of P an adjacent rectangle of width 1 of maximal vertical extent (the principal side is defined similarly). Now, form a bipartite graph $G = (S \cup T, E)$, where $E = \{(y, z) \in S \times T \mid \text{principal sides of } y \text{ and } z \text{ share a corner}\}$ (see *fig.* 5 for an example). Construct a minimum vertex cover R of G using maximum matching. The set of rectangles R constitutes our approximate cover.

Lemma 3.2 *The above heuristic has a performance ratio of* 4 *and runs in* $O(n^{1.5})$ *time.*

Remark: The example where the heuristic attains a performance ratio of 4 is also shown in *fig.* 4.

3.3 Corner cover for polygons without holes.

A succint overview of the heuristic is as follows.

Input: A rectilinear polygon P of n vertices without holes.
Output: A set of rectangles covering the corners of P.

Algorithm outline:

- The first phase partitions the polygon, if permissible, into smaller subpolygons - this constitutes an approximate divide-and-conquer approach. The resulting "primitive" polygons will be easier to cover.
- The second phase finds the "*safe*" rectangles, which always belong to an optimal cover.
- The next phase finds a partial cover with certain desired properties. The remaining uncovered vertices are organized as a set of chains and loops (see *fig.* 7 for an example).
- The last phase covers the chains and loops in a greedy manner.

Although the heuristic is relatively straightforward, its analysis requires us to introduce an amortization scheme and use several technical lemmas. Space limitation prevents any further discussions of the details of the heuristic or its analysis. Details are available in [2]. We state the following theorem without proof.

Theorem 3.1 *The heuristic uses at most* $2\theta - 1$ *rectangles, when* θ *is the optimal cover size. It runs in* $O(n \log n)$ *time.*

Remark: The example where the heuristic attains a performance ratio of 2 is shown in *fig.* 6.

4 Impossibility of approximation schemes.

In this section we prove the following theorem.

Theorem 4.1 *No polynomial time approximation scheme exists for the interior and the boundary cover problems, unless* $P = NP$.

Let (F, P) and (H, Q) be two combinatorial optimization problems where F and H are the cost functions and P and Q are the feasibility predicates. Let $opt_{F,P}(x)$ and $opt_{H,Q}(x)$ be the optimal cost values for an instance x and $quality(a, b) = \max\left(\frac{a-b}{b}, \frac{b-a}{a}\right)$ for positive integers a and b. We say (F, P) can be reduced to (H, Q) preserving approximation [3] with amplification c if and only if there exists two deterministic polynomial time algorithms T_1 and T_2 and a positive constant c such that for all x and y, if $\bar{x} = T_1(x)$ then

(1) $Q(\bar{x}, y) \Rightarrow P(x, T_2(\bar{x}, y))$, and

(2) if $Q(\bar{x}, y)$ then $quality(opt_{F,P}(x), F(T_2(\bar{x}, y))) \leq c(quality(opt_{H,Q}(\bar{x}), H(y)))$.

This reduction ensures that any approximation (H, Q) results in a similarly good approximation of (F, P).
In [6] Culberson and Reckhow show how to reduce the *satisfiability* problem to the interior and boundary cover problems for the proof of NP-completeness of these problems. However, their reduction is not approximation preserving.

Assume that our input is a graph G with v vertices, e edges, and the maximum degree of of any vertex is a constant b. We show how to reduce the vertex cover problem for G to the interior cover problem for a polygon with holes preserving approximation. Because of a recent result of Arora et al. [1], this shows that a polynomial time approximation scheme for the covering problem is impossible, unless $P = NP$.

The overall scheme is shown in *fig.* 8. We use a *gadget* for each vertex. *Beams* (rectangles) coming out of a gadget indicate that this vertex participates in vertex cover. The beams are first translated, then permuted appropriately, again translated and finally enter the edge-gadgets. Each edge gadget is coupled with two beams and represents an edge between the two vertices which correspond to the two beams.

We describe each component in more details. Some of the components used are modified from [6].

Vertex gadget: The vertex gadget is shown in full details in *fig.* 10. It consists of b beam machines when b is the degree of the vertex ($b = 3$ in the figure). Each beam machine can be covered optimally with 6 rectangles with only one rectangle (the *beam*) extending through its mouth in horizontal or vertical direction (see *fig.* 9). This structure has the following properties.

- **(a)** There is an optimal cover of this gadget with $8b + 1$ rectangles when no beam from any of the beam machines extend vertically downwards. This corresponds to the case when this vertex does not participate in a vertex cover.
- **(b)** If a beam from any beam machine extends vertically downwards then $8b+2$ rectangles are necessary and sufficient to cover this gadget. The same property holds when more than one beam from one or more beam machines extends vertically downwards.

Edge gadget: This gadget is shown in *fig.* 11. If either of its two input beams are used then it can be covered minimally with 4 rectangles, otherwise it requires at least 5 rectangles.

Translation stage: It consists of $2e$ pairs of joints. A pair of joint rectangles is an aligned pair of beam machines (*fig.* 12). If the incoming beam for the left polygon of the joint is present (*i. e.* the corresponding edge is present) then the outgoing beam from the right polygon of the joint should be used for optimal cover, otherwise the common horizontal rectangle between should be used. The unique background cover for this stage involves $4e$ rectangles and is shown in *fig.* 12. There are two translation stages. They are used to allow the optimal covering of permutation stages not to affect the vertex gadgets and the edge gadgets.

Permutation stage: There are at most $2e$ permutation stages. This is needed because the order in which the beams come out of the vertex gadgets is not necessarily the same as they arrive at the edge gadgets. Each stage consists of staircases and holes as shown in *fig.* 13. In stage i we put the i^{th} beam from left in the required permutation in its correct place. If a beam is present and covers one notch of the left-side hole, the vertical beam of the beam machine at the right is used for the optimal cover; otherwise, for optimal cover, the horizontal beam of this beam machine covers the notch of the right-side hole. We need exactly 8 rectangles to cover each such stage.

Lemma 4.1 *Let G be a graph with v vertices, e edges, and with the maximum degree of any vertex is a constant b, and let P be the polygon constructed from it by the above procedure. Then, there exists $p \leq 14bv$ such that a vertex cover of G of size m corresponds to a interior cover of P of size $s = 29e+p+(8b+2)v+m$. Conversely, an interior cover of P, which after a polynomial time transformation has size θ, corresponds to a minimal vertex cover of size $\theta - s$.*

Lemma 4.2 *The reduction of the bounded degree vertex cover problem to the rectilinear cover problem as outlined above is approximation preserving.*

Because of the result in [1], Theorem 4.1 follows from Lemma 4.2 in the case of interior cover. A careful examination of our construction shows that the result holds for boundary cover as well.

Acknowledgements.

We wish to thank J. Culberson, H. E. Conn, J. O'Rourke, and G. Schnitger for help.

References

[1] S. Arora, C. Lund, R. Motwani, M. Sudan and M. Szegedy. On the Intractability of Approximation Problems. Manuscript, 1992.

[2] P. Berman and B. DasGupta. Results on Approximation of the Rectilinear Cover Problem. Tech Rep. No. CS-92-07, The Pennsylvania State University, April, 1992.

[3] P. Berman and G. Schnitger. On the complexity of approximating the Independent Set Problem. Information and Computation, 96, 1, pp. 77-94, January, 1992.

[4] S. Chaiken, D. J. Kleitman, M. Saks and J. Shearer. Covering Regions by Rectangles. SIAM J. Algebraic Discrete methods, 2, pp. 394-410.

[5] H. E. Conn and J. O'Rourke. Some restricted Rectangle covering problem. Tech. Rep. JHU-87/13, The Johns Hopkins University, 1987.

[6] J. Culberson and R. A. Reckhow. Covering Polygon is Hard. 29th FOCS, 1988, pp. 601-611.

[7] D. S. Franzblau. Performance Guarantees on a Sweep-line Heuristic for Covering Rectilinear Polygons with Rectangles. SIAM J. Disc. Math., 2, 3, pp. 307-321, 1989.

[8] D. S. Franzblau and D. J. Kleitman. An Algorithm for Constructing Regions with Rectangles. 16th STOC, 1984, pp. 167-174.

[9] J. E. Hopcroft and R. M. Karp. An $n^{\frac{5}{2}}$ Algorithm for Maximum Matching in Bipartite Graphs, SIAM J. Computing, 2, 4, pp. 225-231, 1973.

[10] A. Lingas. The Power of Non-rectilinear Holes. ICALP 1982.

[11] A. Lubiw. Ordering and some Combinatorial Optimization Problems with some Geometric Applications. Ph. D. Thesis, University of Waterloo, 1985.

[12] A. Lubiw. The Boolean Basis Problem and how to Cover Polygons by Rectangles. University of Waterloo, Manuscript, 1988.

[13] W. J. Masek. Some NP-complete Set Covering Problems. MIT, Manuscript, 1978.

[14] C. Mead and L. Conway. Introduction to VLSI Systems. Addision-Wesley. Chapters 2 and 4.

[15] R. Motwani, A. Raghunathan and H. Saran. Perfect Graphs and Orthogonally Convex Covers. 4th SIAM Conf. on Discr. Math., June, 1988.

[16] T. Ohtsuki. Minimum Dissection of Rectilinear Polygons. Proc. IEEE Symp. on Circuits and Systems, 1982, pp. 1210-1213.

[17] J. O'Rourke and K. J. Supowit, Some NP-hard Polygon Decomposition problems. IEEE Trans. Inform. Theory, 29, pp. 181-190, 1983.

[18] L. Pagli, E. Lodi, F. Luccio, C. Mugnai and W. Lipski. On Two Dimensional Data Organization 2, Fund. Inform. , 2, 1979, pp. 211-226.

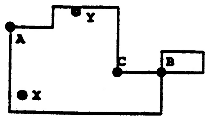

Fig. 1.

A - convex vertex
B - concave vertex
C - degenerate convex vertex
(X,Y)- an anti-rectangle set

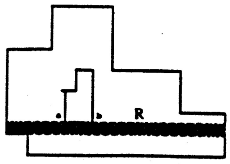

Fig. 3.

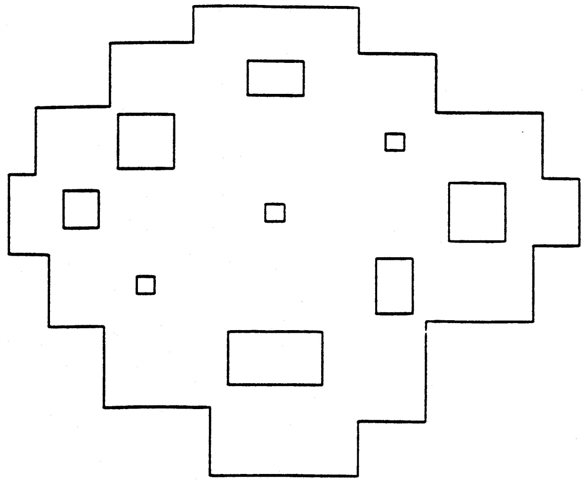

Fig. 4. A polygon with k*k holes such that no two are horizontally or vertically alligned (shown for k=3).

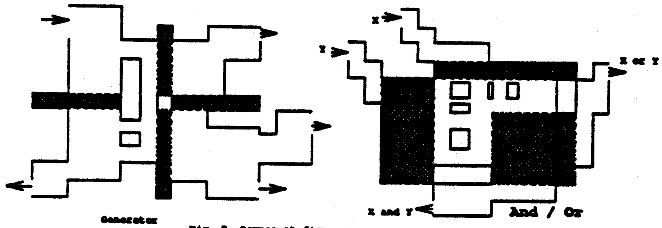

Fig. 2. component figures.
(essential rectangles shaded)

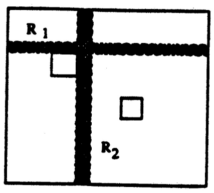

Fig. 5. R $_1$ and R $_2$ are two of the vertices of the of the graph and (R $_1$,R $_2$) is an edge.

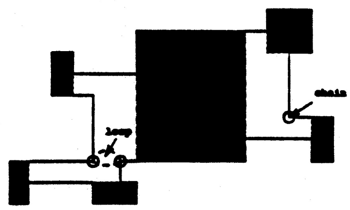

Fig. 7.

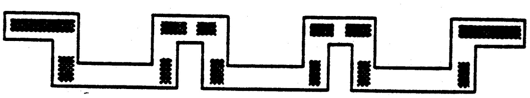

Fig. 6. Worst case example for corner cover without holes. The polygon has k bends (shown for k=13). The optimal cover has (k+1)/2 rectangles, whereas the heuristic generates a cover with k-1 rectangles. (the rectangles in this cover are shown shaded)

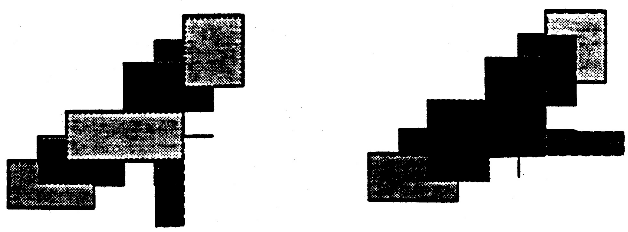

Fig. 9. Two optimal covers for the beam machine.

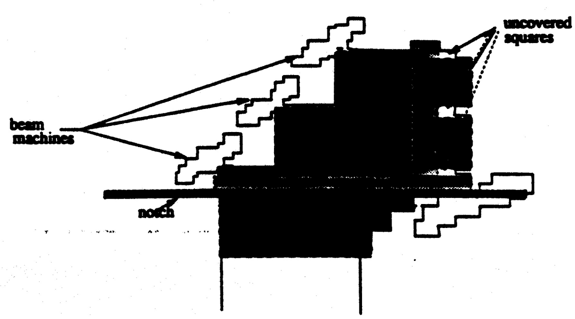

Fig. 10. Vertex gadget with background squares and uncovered squares

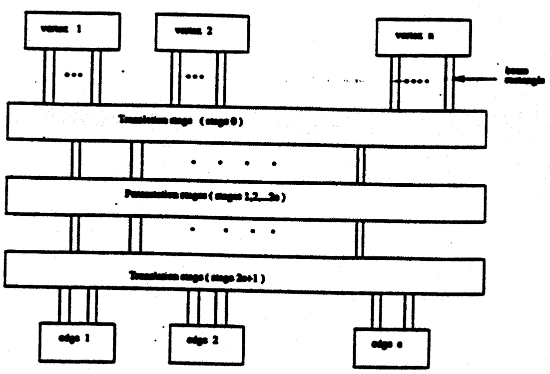

Fig. 8. The overall scheme.

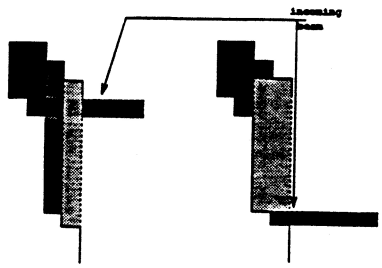

Fig. 11. Edge gadget.

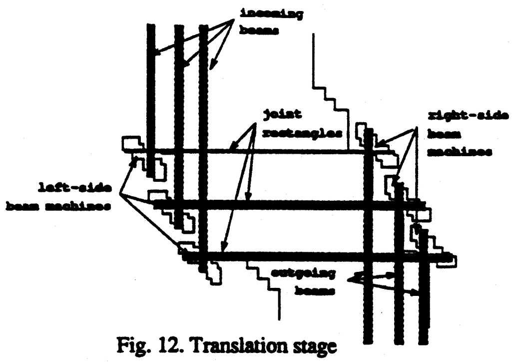

Fig. 12. Translation stage for 3 beams.

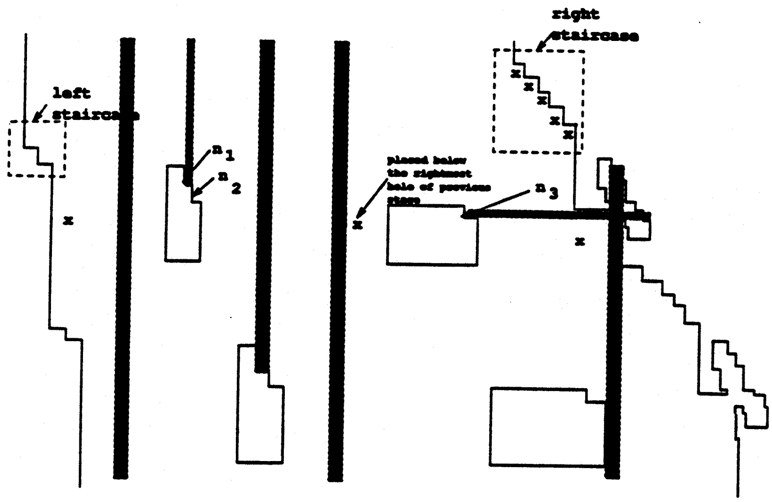

Fig. 15. Anti-rectangle points for a permutation stage (marked by X)

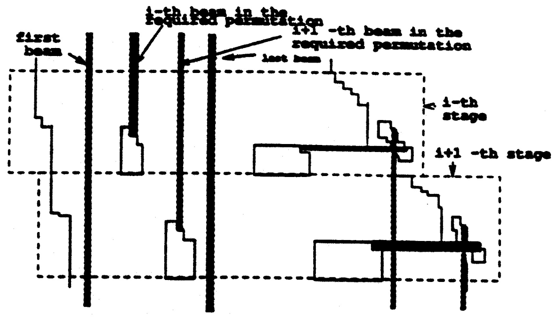

Fig. 13. Permutation stages.

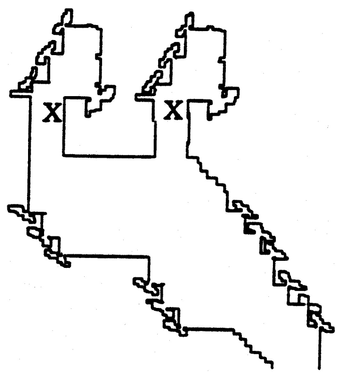

Fig. 14. Joining vertex gadgets. (anti-rectangle points marked by X)

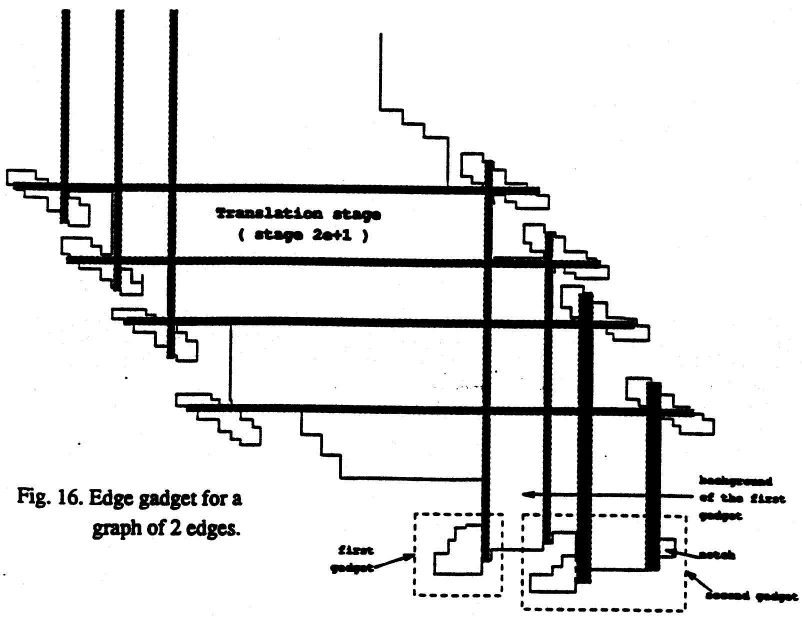

Fig. 16. Edge gadget for a graph of 2 edges.