Non-orthogonal Ray Guarding

Ian Sanders
Department of Computer Science
University of the Witwatersrand
Johannesburg.
E-mail: `ian@cs.wits.ac.za`

Abstract

In an earlier paper the notion of a ray guard, a guard that can only see along a single ray, was introduced. Ray guarding means siting the fewest possible guards that guard all adjacencies (shared edges or parts of edges) in an orthogonal arrangement of adjacent non-overlapping rectangles. In the earlier paper the problem was restricted by requiring that the direction of sight be parallel to one of the Cartesian axes. This problem was shown to be NP-Complete by a transformation from the vertex cover problem for planar graphs. This paper discusses the more general problem where the rays are not restricted to being orthogonal, the same ray can thus cut both horizontal and vertical adjacencies between adjacent rectangles. The problem is shown to be NP-Complete by a transformation from planar vertex cover. The problem of siting ray guards to cover the adjacencies between adjacent convex polygons is a more general case of the non-orthogonal ray guarding problem and the NP-Completeness proof can be extended to this problem as well. Current work is on developing heuristic algorithms for non-orthogonal ray guarding of adjacent rectangles and for the non-orthogonal ray guarding of convex polygons.

1 Introduction

Guarding and covering problems are common in the field of Computational Geometry (see O'Rourke's monograph [7], the survey papers by Shermer [10] and Urrutia [12] and the summaries of results by O'Rourke [8] and Suri [11]). This research focusses on a new variation on guarding problems where the guards can only see along a single ray. The problem has its origin in the area of town planning and urban design — the idea of *Space Syntax Analysis* [6]. Space Syntax Analysis gives a globalising perspective of town design by determining how easy it is to traverse the town. This analysis is accomplished by the positioning of axial lines on a town plan — the fewest such lines are required.

The problem is similar to the many guarding problems [2, 3, 4, 5] since the lines can be thought of as guards whose vision is restricted to a single ray. The situation can be envisaged as an art gallery made up of a number of adjacent rooms where the designers wish to position the most doors between rooms (to allow easy access) in such a fashion that all doorways can be guarded by the minimum number of ray guards. Sanders *et al.* [9] showed that the orthogonal ray guarding problem (where the guards vision is restricted to being parallel to one of the cartesian axes) is NP-Complete.

This paper addresses the non-orthogonal ray guarding problem — finding the minimum number of maximal non-orthogonal rays which cut all of the adjacencies of a collection of adjacent rectangles. A maximal ray is a ray which cuts as many adjacencies as possible. The problem is discussed in more detail in Section 2. In Section 3 the non-orthogonal ray guarding problem is shown to be NP-Complete using a similar transformation to that discussed in Sanders *et al.* [9] for orthogonal rays. In Section 4 some ideas for future research are briefly discussed.

2 Statement of the Problem

Given a number of adjacent, orthogonally-aligned rectangles find the fewest rays (line segments), contained wholly inside the rectangles, required to cut all of the boundaries shared between adjacent rectangles. An additional requirement is that each ray should cut as many of the shared boundaries as possible — a *maximal* ray.

As in the work by Sanders *et al.* [9], depending on how the problem is considered there are two similar but distinct problems which can be addressed — adjacencies can be crossed more than once but every adjacency must be cut at least once; and any adjacency can only be cut once. In this paper only the first variation is addressed. Figure 1 shows an example of this.

The problem considered in this paper can be stated as below.

Non-orthogonal ray guarding
Instance: A collection of orthogonal rectangles $R_1 \ldots R_n$, where each R_i is adjacent to at least one other rectangle, and a positive integer $O \leq 4n$.
Question: Is there a set P of (possibly non-orthogonal) rays where each ray is maximal in length, each ray is contained wholly within the rectangles, each adjacency is crossed at least once by the lines in P and $|P| \leq O$?

In section 3 ***non-orthogonal ray guarding*** is shown to be NP-Complete.

3 Proving the problem is NP-Complete

In Sanders *et al.* [9] it is shown that the problem of finding the minimum number of maximal *orthogonal* rays to guard all of the adjacencies in a configuration of adjacent rectangles (***ray guard***) is NP-Complete. Sanders *et al.* only discusses the problem of horizontal rays and vertical adjacencies — the problem for vertical rays and horizontal adjacencies is similar. The proof is accomplished through a transformation from ***biconnected planar vertex cover*** [9] to ***ray guard*** where ***biconnected planar vertex cover*** and ***ray guard*** are defined as

Biconnected planar vertex cover
Instance: Biconnected planar graph $G = (V, E)$, positive integer $B \leq |V|$.
Question: Is there a vertex cover of size B or less for G, i.e. a subset $V' \subseteq V$ with $|V'| \leq B$ such that for each edge $\{u, v\} \in E$ at least one of u and v belongs to V'?

Ray guard
Instance: A collection of orthogonal rectangles $R_1 \ldots R_n$, where each R_i is adjacent to at least one other rectangle, and a positive integer $O \leq 4n$.
Question: Is there a set P of (*horizontal*) rays where each ray is maximal, each ray is contained wholly within the rectangles, each vertical adjacency is crossed at least once by the rays in P and $|P| \leq O$?

The transformation from ***biconnected planar vertex cover*** is done by mapping vertices in a biconnected planar graph to choice rays. In this mapping an edge between two vertices represents an adjacency which is guarded by two choice rays.

The transformation is done in two steps. First, a biconnected planar graph is transformed to a 'stick diagram' as defined below.

Stick diagram
Instance: A collection H of horizontal lines and U of vertical lines such that each vertical line is cut by exactly two horizontal lines, and a positive integer $S \leq |H|$.
Question: Is there a set of horizontal lines, $H' \subseteq H$, such that every vertical line in U is cut at least once and $|H'| \leq S$?

In this 'stick diagram' each vertex in the original graph is mapped to a horizontal line repre-

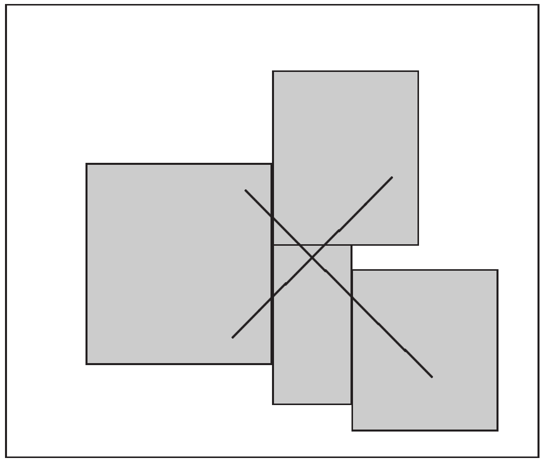

Figure 1: An example of the problem

senting a choice ray and each edge in the original graph is mapped to a vertical line which is cut by the two horizontal lines which represent the two vertices to which the edge is incident. The problem then becomes that of choosing the minimum number of horizontal lines to cut all of the vertical lines. ***Stick diagram*** is thus NP-Complete — if it is possible to determine in polynomial time which of the set of horizontal lines cut all of the vertical lines in the stick diagram then it is possible to solve ***biconnected planar vertex cover*** in polynomial time. Finding the minimum set of horizontal lines is equivalent to finding the minimum vertex cover of the original graph. See Figure 2 for an example of how a biconnected planar graph is transformed to a stick diagram.

Second, the stick diagram is represented as a collection of adjacent rectangles and horizontal rays cutting all of the adjacencies in the collection of rectangles. These horizontal rays are of two types "essential rays" which are the only rays to cut a particular adjacency and "choice rays" where a number of rays (none of which are essential) cut some adjacency. Not all of the choice rays are necessary to guard all of the adjacencies in the collection of rectangles. If it is possible to determine in polynomial time which of the set of choice rays guard all of the adjacencies in the diagram then it is possible to solve ***stick diagram*** in polynomial time — finding the minimum set of choice rays is equivalent to finding the minimum set of horizontal lines. Thus ***ray guard*** has been shown to be NP-Complete [9].

In this paper the fact that ***stick diagram*** is NP-Complete [9] is used to show that ***non-orthogonal ray guarding*** is also NP-Complete. This is done by transforming an instance of ***stick diagram*** to an instance of ***non-orthogonal ray guarding***. Once again this is accomplished by using "choice units" although the units used here are somewhat different to those used in Sanders *et al.* [9] and different non-orthogonal choice rays are generated. Note that, in the remainder of this paper any reference to a ray should be taken to mean a ray which is not necessarily orthogonal.

Theorem 3.1 ***Non-orthogonal ray guarding*** *is NP-Complete*

Proof
Clearly ***non-orthogonal ray guarding*** is in NP. Given a set of non-orthogonal rays it is possible to check in polynomial time that each adjacency has been cut by at least one ray.

Now transform ***stick diagram*** to ***non-orthogonal ray guarding***.

A collection of rectangles which create non-orthogonal choice rays can be represented by a canonical choice unit, *ccu*, shown in Figure 3. In this ccu, the adjacencies between the middle

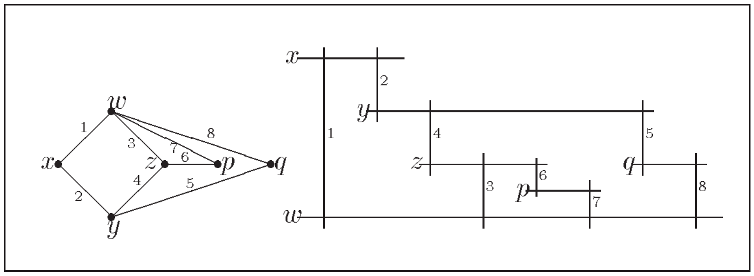

Figure 2: An example of the transformation of a biconnected planar graph to a stick diagram

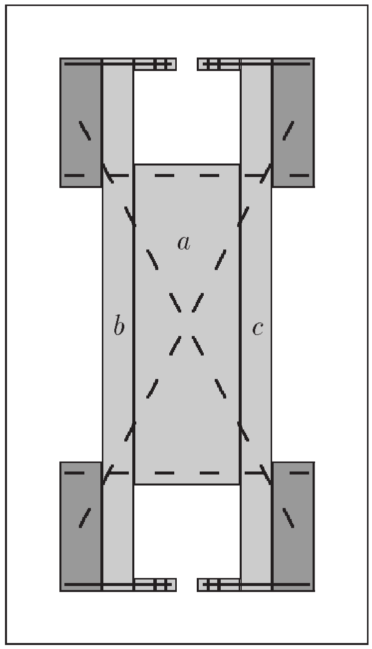

Figure 3: The Canonical Choice Unit which produces non-orthogonal choice rays

rectangle a and the rectangles b and c can be cut by four "sets" of non-orthogonal rays (the upper, lower and two diagonal sets). Figure 3 shows as a dashed line a representative ray from each of the four sets. Only one of these rays is actually necessary to cut the adjacencies between rectangles a, b and c. All the other adjacencies are cut by the rays which originate in the "horns" of the ccu. These rays *do not* have to be horizontal but the size and position of the rectangles in the horns means that the rays are restricted to a small range of different slopes. Scaling of the canonical choice unit does not change the fact that it can/does produce choice rays.

The transformation proceeds by replacing each vertical line in the stick diagram by a ccu of an appropriate size. The horizontal lines which cut through the vertical line are represented by a subset of the choice rays of the ccu. It is necessary to show that these canonical choice units can be joined together in a fashion which maintains the relation between the horizontal lines in the stick diagram and the choice rays in the configuration of adjacent rectangles. The situation here is somewhat different from Sanders *et al.* [9], there the fact that the choice and essential rays had to be horizontal could be used to control the stopping or continuing of rays. In this case the size

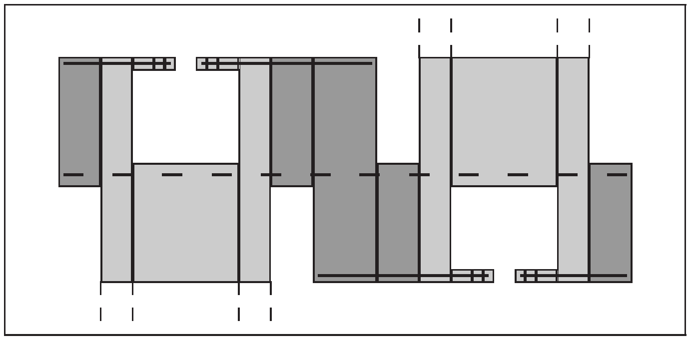

Figure 4: Connecting the upper portion of one ccu to the lower portion of the next

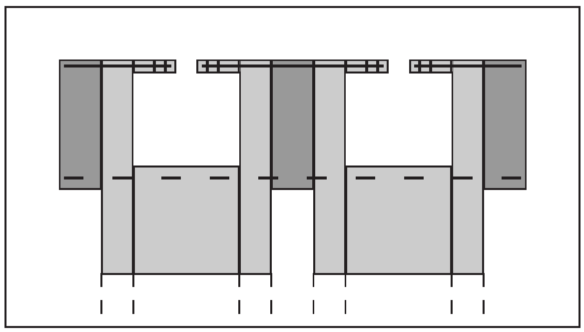

Figure 5: Connecting the upper portions of two ccus

and length to breadth ratio of the ccu's and the connecting rectangles are more crucial.

There are four ways in which horizontal lines could cut through successive vertical lines (actually there are only two ways, each with a vertical reflection). These are

- the horizontal line could be the upper (lower) line through one vertical line and the upper (lower) line through the next vertical line,
- the horizontal line could be the upper (lower) line through one vertical line and the lower (upper) line through the next vertical line.

It must be shown that in each of these cases it is possible to connect two ccu's in such a fashion that the choice is preserved.

These cases are now considered in turn.

- *upper to lower*
  Here the two ccu's are connected by placing a rectangle of appropriate size into the position indicated in Figure 4. This "connecting rectangle" ensures that the upper choice ray through the left ccu and the lower choice ray through the right ccu are the same choice ray. After adding the connecting rectangles, the adjacency between rectangles a, b and c in each ccu is still only cut by choice rays. The sets of choice rays which cut each ccu from the bottom left to the top right (and top left to bottom right) are still possibilities for cutting the adjacencies between a, b and c in each ccu but there are longer rays which cut the same adjacencies so rays from these sets will not be in the final set of rays for the collection of rectangles. The rays from the horns in the left ccu cannot be extended to cut the adjacencies between a, b and c in the right ccu. The only ray which could be extended into the right ccu in this case is the upper choice ray from the lefthand ccu. Thus the choice in both ccu's is maintained. All

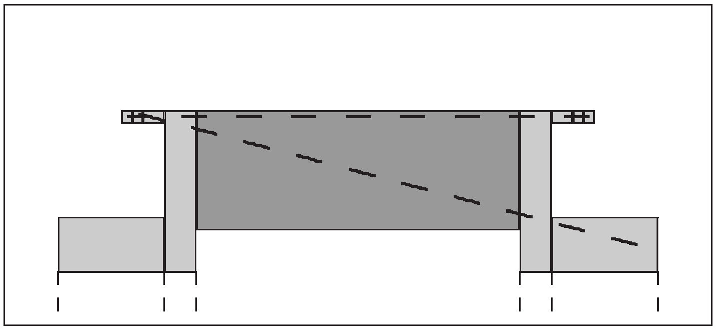

Figure 6: Possible rays from a "horn"

other adjacencies are cut by essential rays.

- *lower to upper*
  This is a mirror image of the case above about the x-axis, see Figure 4.
- *upper to upper*
  In this case no connecting rectangles are required, it is enough to simply merge the appropriate connector rectangles. This is shown in Figure 5. Again the choice rays are maintained and the essential rays cut all other adjacencies.
- *lower to lower*
  Again this case is a mirror image of the case above, see Figure 5.

A potential problem could arise when a number of ccu's are joined together to make up a collection of adjacent rectangles which represents a stick diagram. This could happen because it is possible for the ray through the rectangles in one of the horns to cut one of the adjacencies previously cut only by choice rays. This can be seen in Figure 6 where the range of rays from the horn in the left ccu includes rays which cut an adjacency which was previously only cut by choice rays. This problem can be overcome by some very simple changes to the structure of the ccu. The horn could be made longer — this would be accomplished by keeping the two smaller rectangles in the horn the same size and making the longer rectangle still longer. This would have the effect of reducing the angle at which rays could leave the horn and thus no essential rays could cut the adjacency previously cut only by choice rays. The horn could also be made narrower by changing the vertical extent of the rectangles in the horn. This would have the same effect of reducing the angles at which rays could leave the horn. Other ways of addressing the problems could be to increase the height of the b and c type rectangles and move the horns higher up these rectangles. In this case the range of angles at which rays could leave the horn is kept constant but the required range is increased. A similar approach would be to make the connector rectangle shorter.

The transformation from ***stick diagram*** to ***non-orthogonal ray guarding*** is thus accomplished by inserting an appropriately sized ccu for each vertical line and then joining these up by using the appropriate connecting rectangles working from the leftmost to the rightmost ccu. ccu's should also be scaled as necessary to ensure that the rays from the horns (which are essential) cannot interfere which the issue of choice. Figure 7 shows the final result afer taking an instance of ***stick diagram*** and converting it to an instance of ***non-orthogonal ray guarding***. Note that in this diagram some ccu's have been scaled to ensure that the essential rays from the horns do not interfere with the choice.

It is now necessary to show that there is a solution to ***stick diagram*** if and only if there is a solution to ***non-orthogonal ray guarding***. The construction of the collection of adjacent rectangles from the stick diagram changes the horizontal lines in the stick diagram to choice rays in the collection of rectangles. It also in-

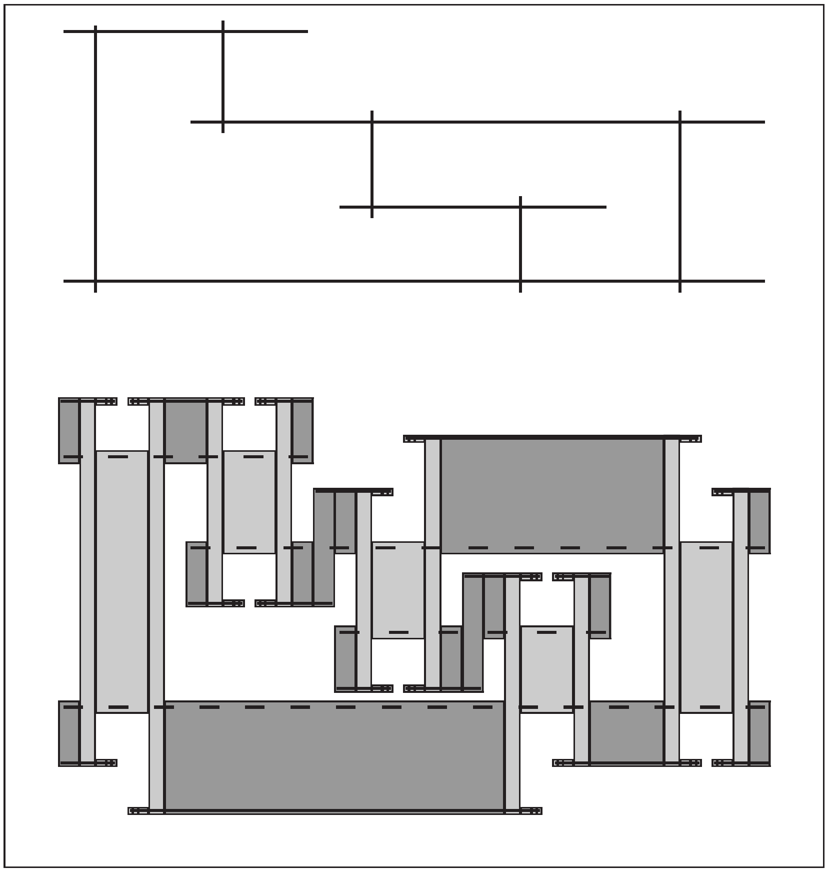

Figure 7: An example of converting a stick diagram to a collection of adjacent rectangles

troduces 4 essential rays for every ccu added (note, some of these rays could be shared across ccu's — see Figure 7 for an example of this), these essential rays must be in the final solution to ***non-orthogonal ray guarding***. Suppose there is a solution for ***stick diagram***, i.e. there exists a set of lines H' such that $|H'| \leq S$, then there must be a solution P to ***non-orthogonal ray guarding*** with $|P| = |H'| + 4|U| - p$, where U is the number of vertical lines (or ccu's) and p is the number of shared essential rays. This is because the essential rays must be in P and the choice rays which correspond to the selected horizontal lines in S must also be in P. Conversely if there is a solution P to ***non-orthogonal ray guarding*** then there must be a solution $S = P - \{e \mid e \text{ is an essential line in } P\}$.

This transformation can clearly be done in polynomial time — each vertical line is visited twice, once when it is replaced by a ccu and a second time when it is connected to the ccu(s) to its right in the stick diagram. If the stick diagram can be drawn then a configuration of ccu's can be drawn by scaling the ccu's to be the same size as the vertical lines that they represent. The ccu's (and their connecting rectangles) can thus be drawn as a non overlapping collection of adjacent rectangles — an instance of ***non-orthogonal ray guarding***.

Non-orthogonal ray guarding is thus NP-Complete. □

The result which has been proved above can be extended to rays with arbitrary orientation cutting the adjacent edges between convex polygons — rectangles are a special case of convex polygons.

4 Future Research

As the problem has been shown to be NP-Complete exact solutions in the general case cannot be found in reasonable time, work is thus currently underway at deriving heuristics which will give acceptable approximations in the general case. Work is also underway in attempting to determine configurations of adjacent rectangles for which the problem can be solved exactly in polynomial time. In both of these situations it is important to be able to efficiently determine if it is possible to draw a straight line through the adjacencies between a number of adjacent rectangles. Bilbrough and Sanders [1] have produced a linear expected-time algorithm to answer this question.

Once the work on non-orthogonal rays and rectangles has been completed, heuristics for approximate solutions and special cases which can be solved in polynomial time for convex polygons will be studied.

Later work will be in relating the results of the cases considered above to the types of configurations of convex polygons which would be obtained in the original town planning problem domain.

5 Conclusion

Orthogonal ray guarding of configurations of adjacent rectangles has been shown to be an NP-Complete problem [9]. This paper shows that non-orthogonal ray guarding of configurations of adjacent rectangles is also NP-Complete. The result in this paper can be extended to rays with arbitrary orientation cutting the adjacent edges between convex polygons — rectangles are a special case of convex polygons.

References

[1] J. Bilbrough and I. D. Sanders. A linear algorithm for partial edge visibility. In *Proceedings of the 1998 SAICSIT Research and Development Symposium*, pages 200–210. South African Institute of Computer Scientists and Information Technologists, November 1998.

[2] I. Bjorling-Sachs and D. L. Souvaine. A tight bound for guarding general polygons with holes. Technical Report LCSR-TR-165, Laboratory for Computer Science Research, Hill Centre for the Mathematical Sciences, Busch Campus, Rutgers University, New Brunswick, New Jersey, 1991.

[3] I. Bjorling-Sachs and D. L. Souvaine. An efficient algorithm for guard placement in polygons with holes. *Discrete & Computational Geometry*, 13:77–109, January 1995.

[4] J. Czyzowicz, E. Rivera-Campo, N. Santoro, J. Urrutia, and J. Zaks. Guarding rectangular art galleries. *Discrete Mathematics*, 50:115–120, 1995.

[5] L. Gewali and S. Ntafos. Covering grids and orthogonal polygons with periscope guards. *Computational Geometry: Theory and Applications*, 2:309–334, 1993.

[6] B. Hillier, J. Hanson, J. Peponis, J. Hudson, and R. Burdett. Space syntax, a different urban perspective. *Architecture Journal*, 30:47–63, 1983.

[7] J. O'Rourke. *Art Gallery Theorems and Algorithms*. Number 3 in The International Series of Monographs on Computer Science. Oxford University Press, New York, 1987.

[8] J. O'Rourke. Visibility. In J. E. Goodman and J. O'Rourke, editors, *Handbook of Discrete and Computational Geometry*, pages 467–479. CRC Press, Boca Raton, 1997.

[9] I. D. Sanders, D. J. Lubinsky, M. Sears, and D. Kourie. Orthogonal ray guarding of adjacencies between orthogonal rectangles. *South African Computer Journal*, (23):18–29, 1999.

[10] T. Shermer. Recent results in art galleries. *Proceedings of the IEEE*, 80(9):1384–1399, September 1992.

[11] S. Suri. Polygons. In J. E. Goodman and J. O'Rourke, editors, *Handbook of Discrete and Computational Geometry*, pages 429–444. CRC Press, Boca Raton, 1997.

[12] J. Urrutia. Art gallery and illumination problems. In J. R. Sack and J. Urrutia, editors, *Handbook on Computational Geometry*. Elsevier Science. To appear.