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. The ray guarding problem is in siting the fewest possible such guards so that they 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 [6] for example). This researc considers a 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 — Hillier *et al*'s idea of *Space Syntax Analysis* [5]. The idea of Space Syntax Analysis is to give a globalising perspective of the 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 art-gallery guarding problems [2, 4, 3] 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, Lubinsky and Sears [7] 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 — the problem of finding the minimum number of maximal non-orthogonal lines which cut all of the adjacencies of a collection of adjacent rectangles. 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.* [7]. 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 maximal straight line segments contained wholly inside the rectangles which will pass through all of the shared boundaries (adjacencies) between adjacent rectangles.

As in the work by Sanders *et al.* [7], depending on how the problem is considered there are 2 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.

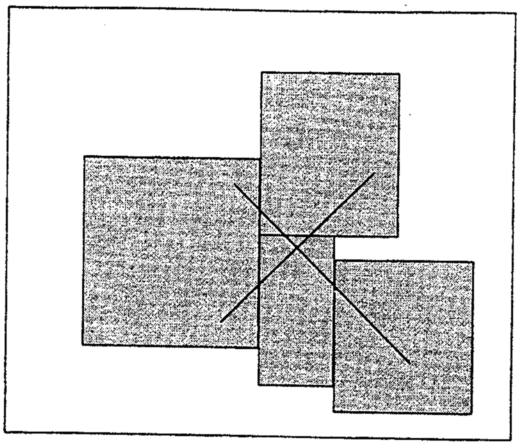

Figure 1: An example of the problem

3 Proving the problem is NP-Complete

The problem to be considered can be stated as below.

non-orthogonal lines
Instance: A collection of orthogonal rectangles $R_1 \dots R_n$, 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 lines where each line is maximal in length, each line is contained wholly within the rectangles, each adjacency is crossed at least once by the lines in P and $|P| \leq O$?

In Sanders *et al.*[7] ***stick diagram*** was shown to be NP-Complete. This result will be used to show that ***non-orthogonal lines*** is also NP-Complete. This will be done by transforming an instance of ***stick diagram*** to ***non-orthogonal lines***. Once again this is accomplished by using "choice units" although the units used here are somewhat different to those used in Sanders *et al.*[7] and different choice lines are generated.

Theorem 3.1 ***non-orthogonal lines*** *is NP-Complete*

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

Now transform ***stick diagram*** to ***non-orthogonal lines***.

A collection of rectangles which create choice lines can be represented by a canonical choice unit, shown in Figure 2. This canonical choice unit, *ccu*, is based on the fact that choice can be produced. In this case, the adjacencies between the middle rectangle a and the rectangles b and c can be cut by four "sets" of lines. Figure 2 shows as a dashed line a representative line from each of the four sets. Only one of these lines is actually necessary to cut the adjacencies between rectangles a, b and c. Note that these two lines do not have to be horizontal they can be at any angle provided they cut the adjacencies between rectangles as listed. All the other adjacencies are cut by the lines which originate in the "horns" of the ccu. Again the lines *do not* have to be horizontal but the size and position of the rectangles in the horns means that the lines 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 lines.

This transformation will proceed 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 to be represented by a subset of the choice lines of the ccu. It is necessary to show that these canonical choice units can be joined together in a fashion which does not limit the choice available. The situation here is somewhat different from Sanders *et al.*[7], there the fact that the choice and essential lines had to be horizontal could be used to control the stopping or continuing of lines. In this case the size and length to breadth ratio of the ccu's and the

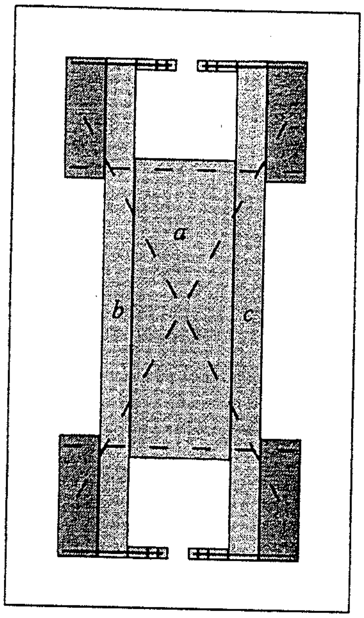

Figure 2: The Canonical Choice Unit which produces non-orthogonal choice lines

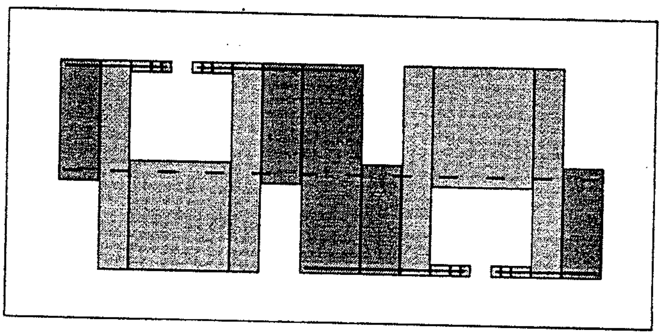

Figure 3: Connecting the upper portion of one ccu to the lower portion of the next

connecting rectangles are more crucial.

There are four ways in which horizontal lines could cut through successive vertical lines. These are

1. the horizontal line could be the upper line through one vertical line and the lower line through the next vertical line,
2. the horizontal line could be the lower line through one vertical line and the upper line through the next vertical line,
3. the horizontal line could be the upper line through one vertical line and the upper line through the next vertical line,
4. the horizontal line could be the lower line through one vertical line and the lower 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

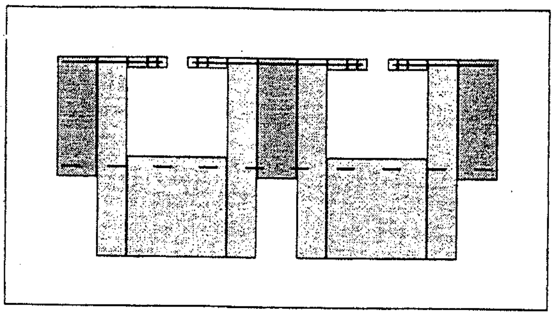

Figure 4: Connecting the upper portions of two ccus

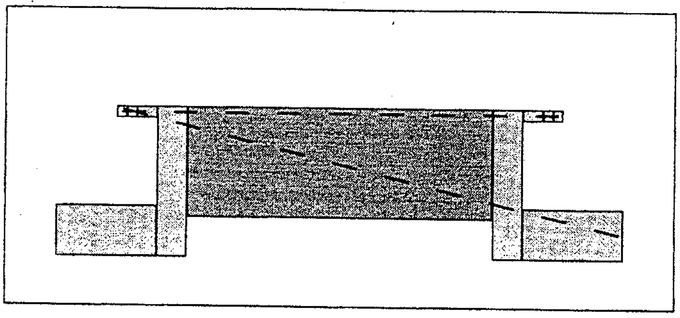

Figure 5: The possible lines from a "horn"

in Figure 3. This "connecting rectangle" ensures that the upper choice line through the left ccu and the lower choice line through the right ccu are the same choice line. After adding the connecting rectangles, the adjacency between rectangles a, b and c in each ccu is still only cut by choice lines. The sets of choice lines 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 but there are longer lines which cut the same adjacencies so lines from these sets will not be in the final set of lines for the collection of rectangles. The lines 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 line which could be extended is the choice line from the lefthand ccu. Thus the choice in both ccu's is maintained. All other adjacencies are cut by essential lines.

- *lower to upper*
  This is a mirror image of the case above about the x-axis, see Figure 3.
- *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 4. Again the choice lines are maintained and the essential lines cut all other adjacencies.
- *lower to lower*
  Again this case is a mirror image of the case above, see Figure 4.

There are potential problems which could arise when a number of ccu's are joined together to make up a collection of adjacent rectangles which represents a stick diagram. These problems could arise because it is possible for the line through the rectangles in one of the horns to cut one of the adjacencies previously cut only by choice lines. This can be seen in Figure 5 where the range of lines from the horn in the left ccu includes lines which cut an adjacency which was previously only cut by choice lines. 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 lines could leave the horn and thus no lines could cut the adjacency previously cut only by choice lines. 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 lines 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 lines could leave the horn are 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 lines*** 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 lines from the horns (which are essential) cannot interfere which the issue of choice. Figure 6 shows the final result afer taking an abitrary instance of ***stick diagram*** and converting it to an instance of ***non-orthogonal lines***. Note that in this diagram some ccu's have been scaled to ensure that the essential lines from the horns do not interfere with the choice.

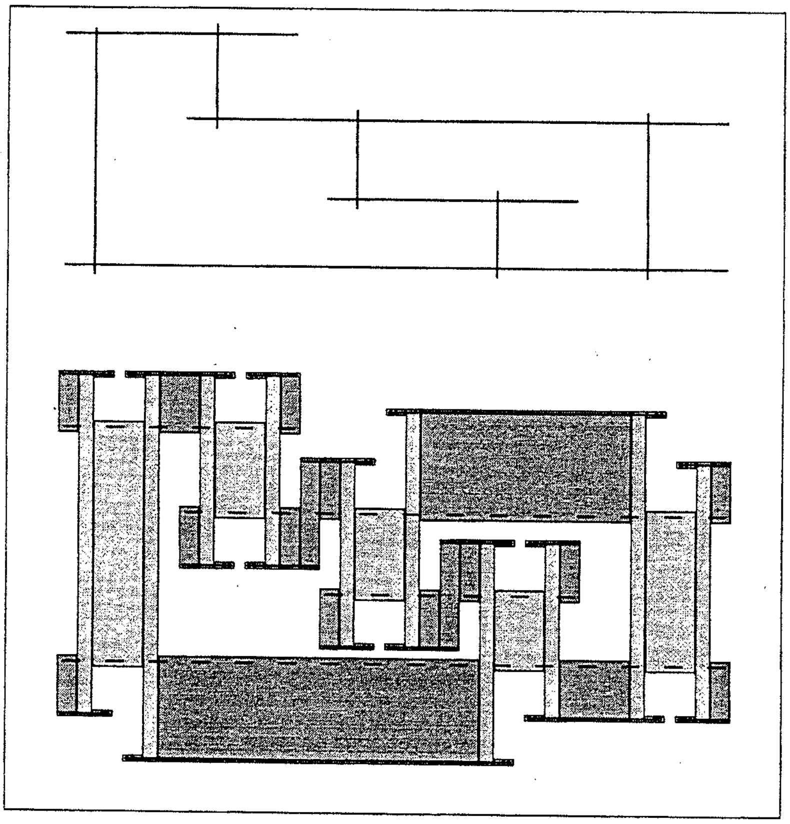

Figure 6: An example of converting a stick diagram to a collection of adjacent rectangles

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 lines***. The construction of the collection of adjacent rectangles from the stick diagram changes the horizontal lines in the stick diagram to choice lines in the collection of rectangles. It also introduces 4 essential lines for every ccu added (note, some of these lines could be shared across ccu's — see Figure 4 for an example of this), these essential lines must be in the final solution to ***non-orthogonal lines***. Suppose there is a solution for ***stick diagram***, i.e. there exists a set of lines S such that $| S | \leq K$, then there must be a solution P to ***non-orthogonal lines*** with $O = | S | + 4V - p$, where V is the number of vertical lines (or ccu's) and p is the number of shared essential lines. This is because the essential lines must be in P and the choice lines which correspond to the selected horizontal lines in S must also be in P. Conversely if there is a solution P to ***non-orthogonal lines*** 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 lines***.

non-orthogonal lines is thus NP-Complete. □

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 time algorithm to answer this question.

Once the work on non-orthogonal lines 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 [7]. 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*. South African Institute of Computer Scientists and Information Technologists, (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. L Gewali and S Ntafos. 'Covering grids and orthogonal polygons with periscope guards'. *Computational Geometry: Theory and Applications*, **2**:309–334, (1993).

5. B Hillier, J Hanson, J Peponis, J Hudson, and R Burdett. 'Space syntax, a different urban perspective'. *Architecture Journal*, **30**:47–63, (1983).

6. J O'Rourke. *Art Gallery Theorems and Algorithms*. Oxford University Press, 1987.

7. I D Sanders, D J Lubinsky, and M Sears. 'Ray guarding configurations of adjacent rectangles'. In *Proceedings of The 1997 National Research and Development Conference (SAICSIT 97)*, pp. 221–238. Potchefstroom University for Higher Christian Education, (November 1997). This paper has subsequently been submitted to the South African Computer Journal.