Information Processing Letters 43 (1992) 69–75
North-Holland

24 August 1992

Special subgraphs of weighted visibility graphs *

F.R. Hsu **

Institute of Computer Science and Information Engineering, National Chiao Tung University, Hsinchu 30050, Taiwan, ROC

R.C.T. Lee

National Tsing Hua University, Hsinchu 30043, Taiwan, ROC

R.C. Chang

Institute of Information Science, Academia Sinica, Taipei, Taiwan, ROC and Institute of Computer and Information Science, National Chiao Tung University, Hsinchu 30050, Taiwan, ROC

Communicated by M.J. Atallah
Received 21 August 1991
Revised 13 April 1992

Keywords: Analysis of algorithms, computational geometry

1. Introduction

A *visibility graph* $G = (V, E)$ of a polygon P is defined as follows: Nodes in V are adjacent if and only if the associated vertices in P are mutually visible and the weight of an edge in E is equal to the Euclidian distance between the associated vertices in P. For example, the visibility graph of the polygon in Fig. 1(a) is shown in Fig. 1(b). A *weighted visibility graph* of a polygon is a visibility graph of this polygon and the weight of an edge of this graph is the Euclidian distance between the associated vertices in this polygon.

In this paper, we are interested in finding a subgraph of a weighted visibility graph G with a small number of edges such that any arbitrary shortest path on G is *almost* entirely covered by the subgraph. For example, in Fig. 2, consider a subgraph of the weighted visibility graph of the polygon in Fig. 1(a). The shortest path between a and f is *abdgf* which is fully covered by this subgraph. The shortest path between a and h is *abh*. Only *bh* is not covered by this subgraph. In this paper, we show that for any arbitrary weighted visibility graph G of an n-gon, there exists a subgraph with only a linear number of edges such that for any arbitrary shortest path of G, there are most $8 \log^* n + 5$ edges that do not appear in the subgraph. Note that $\log^* n$ is defined to be the number of applications of the logarithm function required to reduce n to a constant value (say 2). Given an n-gon, we also propose an $O(n)$ time algorithm to find such a subgraph.

Correspondence to: Professor R.C.T. Lee, National Tsing Hua University, Hsinchu 30043, Taiwan, ROC. Email: rctlee@nthu.edu.tw.

* This work was partially supported by a grant from the National Science Council of the Republic of China under grant no. NSC80-0408-E009-07.

** Email: frhsu@twnctu01.bitnet.

2. Preliminaries

In this section, we introduce some preliminary definitions and geometric properties.

2.1. *The polygon cutting theorem and decomposition tree*

Chazelle [1] proved that for any simple polygon with at least four vertices, there is a diagonal that divides this polygon into two subpolygons, roughly of equal size. More precisely, each of the subpolygons P_1 and P_2 generated by this splitting has at least $\lfloor (n+5)/3 \rfloor$ sides. Consider Fig. 3. The diagonal d_4 is such a diagonal. If we apply Chazelle's Theorem recursively, then we obtain a decomposition tree. The decomposition tree for the triangulation shown in Fig. 3 is shown in Fig. 4.

2.2. *Hourglasses*

Let us introduce an interesting structure, the *hourglass* [5]. Suppose that d_i has endpoints A and B, d_j has endpoints C and D and $BACD$ is a subsequence of the vertex sequence of a polygon P. The union of the shortest path between A and C and the shortest path between B and D is the hourglass of $\overline{AB}$ and $\overline{CD}$, denoted $H(\overline{AB}, \overline{CD})$. Note that each of the paths is *inward convex*. There are two types of hourglasses. If the paths are disjoint, we call it *open* and it is composed of two inward convex chains. If the paths are not disjoint, we call it *closed*. See Fig. 5 for examples of hourglasses.

An interesting property of hourglasses is that an hourglass can be obtained by concatenating two hourglasses and only two new edges are introduced. There are of course many possible cases which we shall not describe here. Interested readers may consult [5].

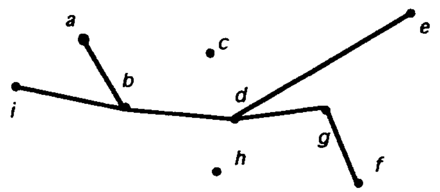

Fig. 2. A subgraph of a weighted visibility graph.

2.3. *The factor graph*

A *factor graph* is constructed out of a decomposition tree [3,5]. Let us consider the decomposition tree in Fig. 4. Note that the subtree of each node is the decomposition tree corresponding to a subpolygon. For instance, the tree under node d_4 corresponds to the entire polygon. For the subtree under node d_2, which is shown in Fig. 6(a), the corresponding subpolygon is shown in Fig. 6(b). Let P_{d_j} denote the subpolygon corresponding to the subtree under node d_j in the decomposition tree. Thus, the subpolygon in Fig. 6(b) will be denoted as P_{d_2}. Now consider P_{d_6}, which is shown in Fig. 7. Diagonal d_6 has three ancestors, namely d_4, d_5, and d_7, in the decomposition tree. Among them, d_5 and d_7 are boundaries of P_{d_6}. Note that for every subpolygon P_{d_j}, except the boundaries of the original polygon, the boundaries of P_{d_j} must be ancestors of d_j in the decomposition tree.

A factor graph is constructed out of a decomposition tree by adding edges from the node

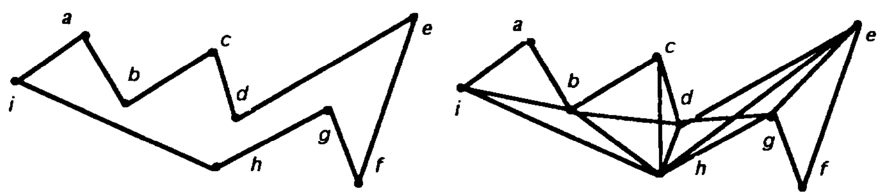

Fig. 1. An example of a visibility graph.

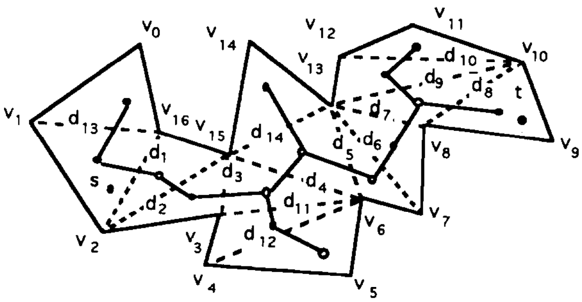

Fig. 3. A triangulated polygon and its dual tree.

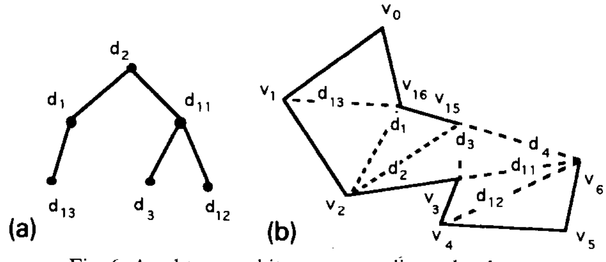

Fig. 6. A subtree and its corresponding subpolygon.

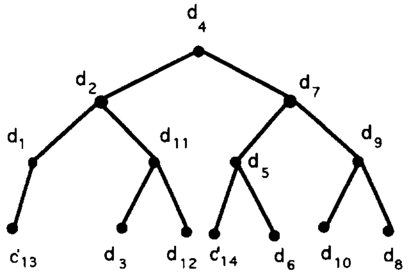

Fig. 4. A decomposition tree.

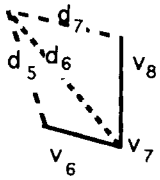

Fig. 7. Subpolygon P_{d_6}.

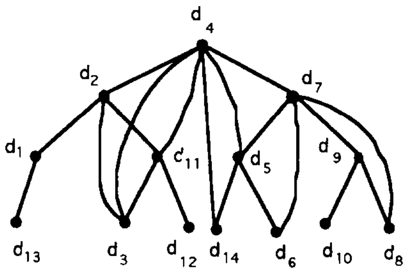

Fig. 8. A factor graph.

corresponding to d_j to all of the nodes corresponding to diagonals bounding P_{d_j} unless the node is already a parent node of the node corresponding to d_j. For the subpolygon in Fig. 7, both d_5 and d_7 are boundaries of P_{d_6}. Thus $\overline{d_5 d_6}$ and $\overline{d_7 d_6}$ are in the factor graph. We do not have to add an edge from d_{13} to d_1 because d_1 is already a parent of d_{13}. The entire factor graph corresponding to the decomposition tree in Fig. 4 is now shown in Fig. 8.

Let us imagine that we want to find the hourglass between d_1 and d_8. From the factor graph shown in Fig. 8, d_4 is again the lowest common ancestor of d_1 and d_8. The path between d_1 and d_4 is $d_1 \to d_2 \to d_4$. We can construct $H(d_1, d_4)$ by constructing $H(d_1, d_2)$ and $H(d_2, d_4)$ and then concatenating them. Note that on the factor graph, there is a path between d_8 and d_4, which is $d_8 \to d_7 \to d_4$. Thus we need only construct the

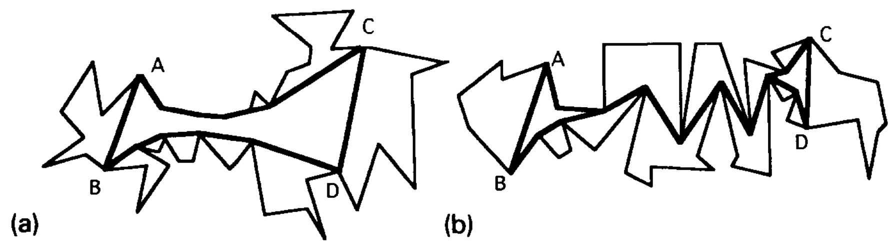

Fig. 5. Examples of hourglasses. (a) An open hourglass. (b) A closed hourglass.

hourglass $H(d_4, d_7)$ and then $H(d_7, d_8)$. From the factor graph, we can also see that $H(d_{12}, d_8)$ will involve the following hourglasses: $H(d_{12}, d_{11})$, $H(d_{11}, d_4)$, $H(d_4, d_7)$ and $H(d_7, d_8)$.

That the factor graph which is sufficient for our purpose can be found in [5].

3. A necessary condition

In this section, we shall show that for any arbitrary weighted visibility graph of an n-gon, there exists a subgraph with only a linear number of edges such that for any arbitrary shortest path there are most $8 \log^* n + 5$ edges that do not appear in the subgraph. Given an n-gon, we also propose an $O(n)$ time algorithm to find such a subgraph.

Now let us define some new terms. The modified factor graph is defined as follows: Let U denote the set of nodes in the decomposition tree, each of which having at least $\log^4 n$ descendants. The modified factor graph is constructed out of the factor graph by adding edges from each node in U to all its descendants that are also in U if there is no edge in the factor graph linking these nodes. In the rest of this paper, let F and S denote the factor graph and the decomposition tree respectively.

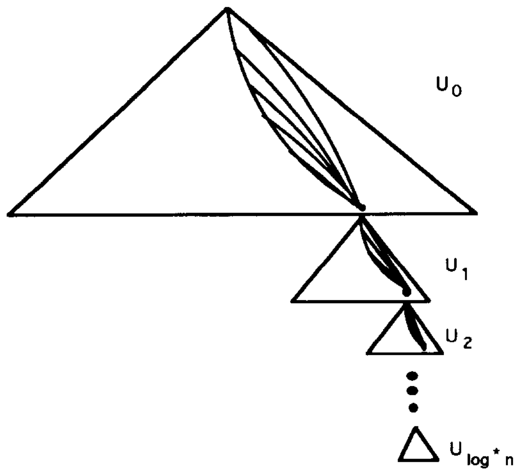

Fig. 9. A leap graph.

A *leap graph* L is a generalization of the modified factor graph. Let U_0 be the set of nodes with at least $\log^4 n$ descendants in the decomposition tree S. For $i = 1, 2, \ldots, \log^* n$, let U_i be the set of nodes in S whose numbers of descendants are less than $(\log^{(i)} n)^4$ and not less than $(\log^{(i+1)} n)^4$. Note that $\log^{(k)} n = \log \log^{(k-1)} n$ and $\log^{(0)} n = n$. These nodes also exist in the factor graph F. A leap graph is constructed out of a factor graph by adding edges from nodes of U_i in F to all their descendants that are also in U_i, for $i = 0, 1, \ldots, \log^* n$. If these edges are already in F, they will not be added into L again. See Fig. 9. We now prove that the size of L is $O(n)$.

Lemma 1. *The number of edges of a leap graph is* $O(n)$.

Proof. Since the decomposition tree S is balanced, there are at most $O(n/(\log^{(i+1)} n)^4)$ nodes in S with at least $O((\log^{(i+1)} n)^4)$ descendants. Thus, the cardinality of U_i is only $O(n/(\log^{(i+1)} n)^4)$. Furthermore, since S is balanced and the number of descendants of any node in U_i is at most $(\log^{(i)} n)^4$, each node in U_i is reached by at most $4 \log^{(i+1)} n$ additional edges from its ancestor. Therefore, there are at most $O(n/(\log^{(i+1)} n)^3)$ edges added from nodes in U_i to all of their descendants in U_i. Thus the total number of edges added to F is

$$\begin{aligned}&\sum_{i=0}^{\log^* n} O\left(n/(\log^{(i+1)} n)^3\right)\\ &= O\left(n/(\log n)^3\right) + O\left(n/(\log\log n)^3\right)\\ &\quad + O\left(n/(\log\log\log n)^3\right) + \cdots\\ &= O(n).\end{aligned}$$

Since the size of the factor graph is $O(n)$, the number of edges of a leap graph is $O(n)$. □

Let us define some new terms before discussing the properties of the leap graph. A diagonal sequence $D = (d_{r_1}, d_{r_2}, \ldots, d_{r_k})$ is *serially concatenable* if $d_{r_{t+1}}$ separates d_{r_t} and $d_{r_{t+2}}$ in the original polygon for $1 \leqslant t \leqslant k-2$. It is not difficult to see that any subsequence of a serially concatenable diagonal sequence is also serially concatenable. The following lemma points out an important property of the leap graph.

Lemma 2. *For every d_i, d_j in the decomposition tree S, there exists a serially concatenable diagonal sequence $(d_{r_1} = d_i, d_{r_2}, \dots, d_{r_t} = d_j)$ such that $t \leqslant 4 \log^* n + 3$ and $\overline{d_{r_l} d_{r_{l+1}}}$ in the leap graph L for $1 \leqslant l \leqslant t-1$.*

Proof. For d_i and $d_j \in S$, let d' denote the lowest common ancestor of d_i and d_j in S. By the definition of S, d' separates d_i and d_j in the original polygon. If $\overline{d_i d'} \in L$ and $\overline{d_j d'} \in L$, then the lemma is true.

Consider that case that either $\overline{d_i d'} \notin L$ or $\overline{d_j d'} \notin L$. Suppose that $\overline{d_i d'} \notin L$. Let $Next(d_p, d_q)$ denote the lowest diagonal among diagonals that are on the path from d_p to d_q in S and separate d_p and d_q in the original polygon. If d_q is the parent of d_p, let $Next(d_p, d_q) = d_q$. Consider the diagonal sequence $DS = (d_{k_1} = d_i, d_{k_2}, \dots, d_{k_s} = d')$ where $d_{k_{l+1}} = Next(d_{k_l}, d')$ for $1 \leqslant l \leqslant s-1$. By the definition of $Next(d_p, d_q)$, it is not difficult to see that DS is serially concatenable. We now further show that $\overline{d_{k_l} d_{k_{l+1}}} \in F$ for $1 \leqslant l \leqslant s-1$.

Consider $\overline{d_{k_l} d_{k_{l+1}}}$. If $d_{k_{l+1}} = d'$, by definition, $\overline{d_{k_l} d_{k_{l+1}}}$ is in F. Consider the case that $d_{k_{l+1}} \neq d'$. By definition, $d_{k_{l+1}}$ separates d_{k_l} and d'. Assume that $\overline{d_{k_{l+1}} d_{kl}}$ is not in F. By the definition of S, $d_{k_{l+1}}$ is a boundary of $P_{LSON(d_{k_{l+1}})}$ and $P_{RSON(d_{k_{l+1}})}$. Since $d_{k_{l+1}}$ is an ancestor of d_{k_l} in S, $P_{d_{k_l}}$ is a subpolygon of $P_{LSON(d_{k_{l+1}})}$ or $P_{RSON(d_{k_{l+1}})}$. Since $\overline{d_{k_{l+1}} d_{k_l}}$ is not in F, $d_{k_{l+1}}$ is not a boundary of $P_{d_{k_l}}$. Therefore, there must be some diagonal, say d_j', which separates $d_{k_{l+1}}$ and d_{k_l} and is on the path from d_{k_l} to $d_{k_{l+1}}$ in S. Since $d_{k_{l+1}}$ separates d' and d_{k_l} and d_j' separates $d_{k_{l+1}}$ and d_{k_l}, d_j' must separate d' and d_{k_l}. Besides, d_j' is younger than $d_{k_{l+1}}$ in S, a contradiction. Thus $\overline{d_{k_{l+1}} d_{k_l}}$ must be in F. Therefore, $\overline{d_{k_l} d_{k_{l+1}}} \in F$ for $1 \leqslant l \leqslant s-1$.

For $0 \leqslant t \leqslant \log^* n$, let $LD(t)$ and $HD(t)$ denote the lowest and highest diagonals in U_t that are also in DS. If no such diagonals exist, let these notations denote empty. Consider the subsequence $DS' = (d_p = d_{f_1}, d_{f_2}, \dots, d_{f_e} = d')$ of DS, where DS' is composed of $LD(t)$ and $HD(t)$ for $0 \leqslant t \leqslant \log^* n$. Consider $\overline{d_{f_l} d_{f_{l+1}}}$. If $d_{f_{l+1}}$ is also the successor of d_{f_l} in DS, then $\overline{d_{f_l} d_{f_{l+1}}} \in F$. If $d_{f_{l+1}}$ is not the successor of d_{f_l} in DS, d_{f_l} and $d_{f_{l+1}}$ must be in the same U_t. Therefore, $\overline{d_{f_l} d_{f_{l+1}}} \in L$. Since DS is serially concatenable, DS' is also serially concatenable. To sum up, for the case that $\overline{d_i d'} \notin L$, there exists a serially concatenable diagonal sequence $DS' = (d_{f_1} = d_i, d_{f_2}, \dots, d_{f_e} = d')$ such that $e \leqslant 2 \log^* n + 2$ and $\overline{d_{f_l} d_{f_{l+1}}} \in L$ for $1 \leqslant l \leqslant e'-1$. If $\overline{d_j d'} \notin L$, we similarly can show that there exists a serially concatenable diagonal sequence $DS_2' = (d_{f_1'} = d', d_{f_2'}, \dots, d_{f_{e'}'} = d_j)$ such that $e' \leqslant 2 \log^* n + 2$ and $\overline{d_{f_l} d_{f_{l+1}}} \in L$ for $1 \leqslant l \leqslant t-1$. Since d' separates d_i and d_j, it follows that the combination of DS' and DS_2' is serially concatenable. Therefore, there exists a serially concatenable diagonal sequence $(d_{r_1} = d_i, d_{r_2}, \dots, d_{r_t} = d_j)$ such that $t \leqslant 4 \log^* n + 3$ and $\overline{d_{r_l} d_{r_{l+1}}} \in L$ for $1 \leqslant l \leqslant t-1$. □

Now we can show a necessary condition of the weighted visibility graph in the following theorem.

Theorem 3. *For any arbitrary weighted visibility graph of an n-gon, there exists a subgraph with only a linear number of edges such that for any arbitrary shortest path, there are most $8 \log^* n + 5$ edges that do not appear in the subgraph. There exists an algorithm to construct such a subgraph in $O(n)$ time.*

Proof. Recall that by definition, an hourglass consists of two shortest paths. Let SL denote the shortest path set defined by the hourglasses corresponding to edges in the leap graph L. Let $US(SL)$ denote the union of SL. We will show that $US(SL)$ satisfies the statement in the theorem.

For any arbitrary vertices p and q of P, let $\pi(p, q)$ denote the shortest path between p and q inside P. Let d_p be a diagonal incident to p if one exists, or otherwise the diagonal closest to p that separates p from q. We define d_q in the similar way. By lemma 2, there exists a serially concatenable diagonal sequence $DS = (d_{r_1} = d_p, d_{r_2}, \ldots, d_{r_t} = d_q)$ such that $t \leqslant 4 \log^* n + 3$ and $\overline{d_{r_l} d_{r_{l+1}}} \in L$ for $1 \leqslant l \leqslant t-1$. Since DS is serially concatenable, $H(d_p, d_q)$ can be constructed by concatenating $H(d_{r_1}, d_{r_2})$, $H(d_{r_2}, d_{r_3}), \ldots, H(d_{r_{t-1}}, d_{r_t})$. It follows that $H(d_p, d_q)$ can be constructed by concatenating at most $4 \log^* n + 2$ hourglasses corresponding to edges in L. Concatenating two hourglasses introduces only two new edges. Thus, it introduces at most $8 \log^* n + 2$ new edges to construct $H(d_p, d_q)$ from $US(SL)$. Since at most three line segments of $\pi(p, q)$ are not covered by $H(d_p, d_q)$, it follows that there are at most $8 \log^* n + 5$ edges of $\pi(p, q)$ not covered by $US(SL)$.

Consider the problem of constructing $US(SL)$. Since $US(SL)$ is the union of shortest paths associated with hourglasses corresponding to edges in L and whenever an hourglass is found, its two associated shortest paths are also found, we can construct $US(SL)$ \by constructing hourglasses corresponding to edges in L. Consider the hourglasses corresponding to edges in L. Let $f(k)$ be the time complexity of triangulating a k-gon. Guibas and Hershberger [5] proposed an $f(n)$ time algorithm to construct hourglasses corresponding to edges in F and these hourglasses can be stored in $O(n)$ space such that for d_i and its descendant d_j, $H(d_i, d_j)$ can be constructed in $O(\log^2 |P_{d_i}|)$ time. Since currently, the best bound for $f(k)$ is $O(k)$ [2], Guibas and Hershberger's algorithm [5] can be run in linear time. Consider hourglasses corresponding to edges in $L \setminus F$. By definition, for any d_k in U_i, $|P_{d_k}|$ is at most $O((\log^{(i)} n)^4)$. Thus, for d_t, d_k in U_i and $\overline{d_t d_k}$ in L, we can construct $H(d_t, d_k)$ in

$$O\left(\log^2 (\log^{(i)} n)^4\right) = O\left((\log^{(i+1)} n)^2\right)$$

time. As shown in the proof of Lemma 1, there are at most $O(n/(\log^{(i+1)} n)^3)$ edges added from nodes in U_i to all of their descendants in U_i. Thus, we can construct hourglasses corresponding to edges added from nodes in U_i to all of their descendants in U_i in

$$O\left(n/(\log^{(i+1)} n)^3\right) O\left((\log^{(i+1)} n)^2\right) = O(n/\log^{(i+1)} n).$$

It follows that we can construct hourglasses corresponding to edges in $L \setminus F$ in

$$\sum_{i=0}^{\log^* n} O(n/\log^{(i+1)} n) = O(n/\log n) + O(n/\log\log n) + O(n/\log\log\log n) + \cdots = O(n).$$

Therefore, the hourglasses corresponding to edges in L can be constructed in $O(n)$ time and space. It follows that $US(SL)$ can be constructed in $O(n)$ time and space. □

4. Conclusion

In this paper, we are interested in some theoretical properties of weighted visibility graphs. We show that for any arbitrary weighted visibility graph of an n-gon, there exists a subgraph with only a linear number of edges such that for any arbitrary shortest path, there are most $8 \log^* n + 5$ edges that do not appear in this subgraph. Given an n-gon, we also propose an $O(n)$ time algorithm to find such a subgraph. It is a challenge to design a fast algorithm to find such a subgraph with minimum weight. It should be noted that we are not suggesting that this subgraph should be used as data structure to find shortest paths. The following problem remains open: for any arbitrary weighted visibility graph of an n-gon, does there exist a subgraph with $O(n)$ edges such that the maximum number of missing edges of any arbitrary shortest path is less than $O(\log^* n)$?

References

[1] B. Chazelle, A theorem on polygon cutting with applications, in: *Proc. 23th Ann. IEEE Symp. on Foundations of Computer Science* (1982) 339–349.

[2] B. Chazelle, Triangulating a simple polygon in linear time, *Discrete Comput. Geom.* **6** (1991) 485–524.

[3] B. Chazelle and L. Guibas, Visibility and intersection problem in plane geometry, *Discrete Comput. Geom.* **4** (1989) 551–581.

[4] L. Guibas, J. Hershberger, D. Leven, M. Sharir and R.E. Tarjan, Linear time algorithms for visibility and shortest path problems inside triangulated simple polygons, *Algorithmica* **2** (1987) 209–233.

[5] L.J. Guibas and J. Hershberger, Optimal shortest path queries in a simple polygon, *J. Comput. System Sci.* **39** (1989) 126–152.

[6] J. O'Rourke, *Art Gallery Theorems and Algorithms* (Oxford University Press, New York, 1987).

[7] M. Overmars and H. van Leeuwen, Maintenance of configurations in the plane, *J. Comput. System Sci.* **23** (1981) 166–204.