Author's version

Published in: IEEE Symposium on Emerging Technologies and Factory Automation, ETFA, 2007, Article number4416800, Pages 434-440

A Tight Lower Bound for Art Gallery Sensor Location Algorithms

Andrea Bottino
Politecnico di Torino
C.so Duca degli abruzzi 24
10129, Torino, Italy
andrea.bottino@polito.it
Aldo Laurentini
Politecnico di Torino
C.so Duca degli abruzzi 24
10129, Torino, Italy
aldo.laurentini@polito.it
Luisa Rosano
Politecnico di Torino
C.so Duca degli abruzzi 24
10129, Torino, Italy
luisa.rosano@polito

Abstract

Locating sensors in 2D can be often modelled as an Art Gallery problem. Tasks such as surveillance require observing or "covering" the interior of a polygon with a minimum number of sensors or "guards". Observing the boundaries of a polygonal environment is sufficient for tasks such as inspection and image based rendering. As interior covering, also Edge Covering (EC) is NP-hard, and no finite algorithm is known for its exact solution. A number of heuristics have been proposed for the approximate solution of this important problem, but their performances with respect to optimality is unknown. Therefore, a polygon specific tight lower bound for the number of sensors is very useful for assessing the performances of these algorithms. In this paper, we propose a new lower bound for the EC problem. It can be computed in reasonable time for environments with up to a few hundreds of edges. To evaluate its closeness to optimality, we compare it with a previously developed lower bound and with the solution provided by a recent incremental EC algorithm. Tests over hundreds of polygons with different number of edges show that the new lower bound is tight and outperforms the previous one.

1. Introduction

Several computer vision and robotics tasks, as surveillance, inspection, image based rendering, constructing environment models, require multiple sensor locations, or the displacement of a sensor in multiple positions for fully exploring an environment or an object.

Sensor placement, or planning, or location, is an important area of research. A recent sensor placement survey [15] refers to tasks as reconstruction and inspection. Several other tasks and techniques were considered in [11] and [16].

Sensor location problems require considering a number of constraints, such as image resolution, field of view of the sensors, feature visibility, lighting, etc. Visibility is clearly the fundamental constraint. An omni directional or rotating sensor is usually modeled as a point. A feature of an object is visible from the sensor if any segment joining a point of the feature and the viewpoint does not intersect the environment or the object itself.

Although the general problem is three-dimensional, in several cases it can be restricted to 2D. This is for instance the case of buildings, which can be modeled as objects obtained by extrusion. The 2D visibility constraint is modeled by the classic Art Gallery problem, which asks to position a minimum set of "guards" able to see, or "cover" a polygonal environment. Tight upper bounds for the cardinality of the set of guards have been found in several cases. The famous Art Gallery Theorem by Chvàtal states that at most $\lfloor n/3 \rfloor$ guards are required for covering any simple polygon with n edges, metaphorically the interior of an art gallery. The upper tight bound $\lfloor (n+h)/3 \rfloor$ holds for polygons with n edges and h polygonal holes. Many variations of the problem have been considered, as for instance particular kind of polygons, restricted positions for the guards, additional constraints. For further details, the reader is referred to the monograph by O'Rourke [12] and to the surveys by Shermer [14] and Urrutia [17].

Unfortunately, the practical problems, that is finding the cardinality of the minimum set of guards and locating these guards in a given polygon, are NP-hard, and no finite exact algorithm, not even exponential, is known for locating a minimum cover. In addition, approximate algorithms polynomial in the worst case and with guaranteed performance are unlikely to exist [5].

Observe that, for tasks as surveillance, sensors are required to observe, or cover, the *interior* of a polygon. Other tasks, such as inspection, a main application of sensor planning according to the survey [16], and image based rendering, only require observing the *boundary*. In this paper, we will deal with observing the *edges* of a polygonal environment. We call this the *Edge Covering (EC)* problem, while the classic problem will be referred to as the *Interior Covering (IC)* problem. The EC problem and its relations with IC have been analyzed in [21]. The Chvàtal bound also holds for EC, but, although any interior cover is also an edge cover, in general an optimum set of IC guards is not an optimum set of EC guards and vice-versa. Examples show that the number of interior guards may be two times, for simple polygons, or O(n) times, for polygons with holes, the edge guards. Then the EC problem is different from IC, but not easier. Actually, also EC is NP-hard [21], and no finite exact algorithm is known for locating a minimum set of EC guards in a given polygon

Even if they are NP-hard, the IC and the EC problems are important in practice, and many approximate sensor positioning algorithms have been proposed. Some approximate polynomial algorithms for IC are reported by Shermer in [14]. Others worst-case polynomial algorithms have been presented later, for instance by Bjorling-Sachs and Souvaine [3] and Elnagar and Lulu [6], [7]. As for the EC problem, some attempt has been made for constructing *practical* sensor positioning algorithms. Kazakakis and Argyros [10] have proposed and implemented a polynomial heuristic that also takes into account the visibility constraint. The randomized approach (Danner and Kavraki [4], Gonzales-Banos and Latombe [8], [9]) attempts to approach the optimal solution by locating at random many sensors. Only a few of these algorithms have been implemented, and in any case no experimental results comparing the cardinalities of the solution provided by these algorithms with the optimal solution have been presented.

Recently, an EC incremental sensor location algorithm has been presented [22]. This algorithm converges toward the optimal solution in an undefined number of steps, and makes use of a lower bound, *specific of the polygon considered*, for the minimum, or optimum, number of guards. The lower bound allows evaluating the quality of the solution obtained at each step, and halting the algorithm if the solution is satisfactory. Experimental results, showing that on the average the algorithm supplies solutions close to the lower bound, are presented in [22].

Clearly, since no known algorithm is able to compute the cardinality of a minimum set of EC guards, a tight lower bound is of great importance for evaluating the quality of sensor positioning algorithm. In this paper, we present and discuss a new, polygon specific, lower bound algorithm. The lower bound computed with this algorithm is equal or larger than that computed with the algorithm described in [22]. The algorithm for computing the lower bound has been implemented and tested for many random polygons of different categories and different number of edge, and compared with the results supplied by the previous lower bound algorithm described in [22]. The tests show that the new lower bound is significantly larger, and thus better, than that provided by the previous algorithm. The algorithm is not polynomial, but its running time allows dealing with polygons with up to a few hundred of edges.

The paper is organized as follows. In Section 2, we describe the new lower bound algorithm. Section 3 provides the experimental results and comparisons. Concluding remarks are reported in Section 4.

2. The Lower Bound Algorithm for EC

2.1. The previous Lower Bound and its shortcomings

Let us first recall the lower bound algorithm described in [22]. It is based on the concept of *weak visibility polygon of an edge.* Two points of a polygon P are visible, or see each other, if the segment joining the points lies completely in P. According to the definition given by Avis and Toussaint [2], a polygon W is weakly visible from an edge e if for each point $w \in W$ there exists at least a point $z \in e$ such that w is visible from z. In other words, the weak visibility polygon $W(\mathrm{e}_i)$ of an edge e_i is the polygon whose points see at least a point of e_i. Observe that points seeing only one vertex of e_i do not belong to $W(e_i)$. Examples of weak visibility polygons are shown in Fig.1. Polynomial algorithms for computing weak visibility polygons of an edge are described in the literature [13]. In our case, however, weak visibility polygons are computed as a by-product of the sensor location algorithm described in [22].

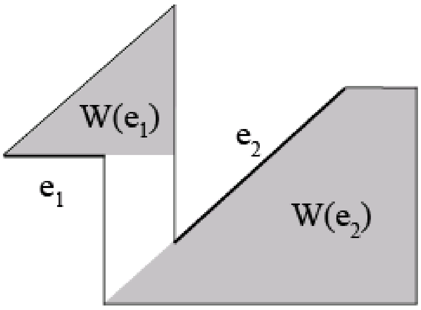

Figure 1. Two weak visibility polygons. Each of these polygons must contain at least one guard.

Weak visibility polygons allow us to determine a lower bound for the number of sensors needed. In fact, it is clear that each weak visibility polygon must contain at least one sensor, otherwise no points of the edge are seen by any sensor. Therefore, a lower bound $LB_W(P)$ for a polygon P is obtained by computing *the cardinality of the maximal subset of disjoint (not intersecting) weak visibility polygons* $W(e_i)$ *of* P.

A simple example is shown in Fig.1. It is easy to verify by inspection that no more than two disjoint weak visibility polygons can be found, for instance W(e_1) and W(e_2), and thus LB_W(P)=2.

Computing LB_W requires solving the *maximum independent set problem* for a graph *G* where each node represents the weak visibility polygon of an edge of *P*, and each edge of *G* connects nodes corresponding to intersecting weak visibility polygons. The problem is equivalent to the *maximum clique problem* for the *complement* graph *G'*. Although this is an NP-complete problem, *exact* branch-and-bound algorithms for these problems have been presented and extensively tested ([17], [18], [19]), showing more than acceptable performances for graphs with hundreds of nodes.

The tests reported in [22] also show that on the average the difference between the LB_W(P) and the cardinality of the solution provided by the sensor location algorithm is small, and both are close to the optimum cardinality that lies in between.

However, the algorithm for computing LB_W fails to produce good results in some simple cases. Consider for example the case in Fig.2, showing a comb polygon of a family used for showing that the Chvàtal upper bound is tight for both IC and EC. Only two not intersecting weak visibility polygons can be found, for instance those shown in Fig.2 (a), and then LB_W(P)=2. However, three EC guards are clearly required, one for each spike. The reason of the bad behavior of the algorithm in this case can be appreciated from Fig.2 (b), where the weak visibility polygon W(e_3) of one of the edges forming the central spike is shown. W(e_3) intersects W(e_1), and likewise W(e_4) intersects W(e_2).

Let us observe that similar arguments show that the lower bound LB_W is 2 for *all polygons* of the comb family: as the number of spikes and guards increases, the gap between the lower bound and the cardinality of the minimum set increases as well.

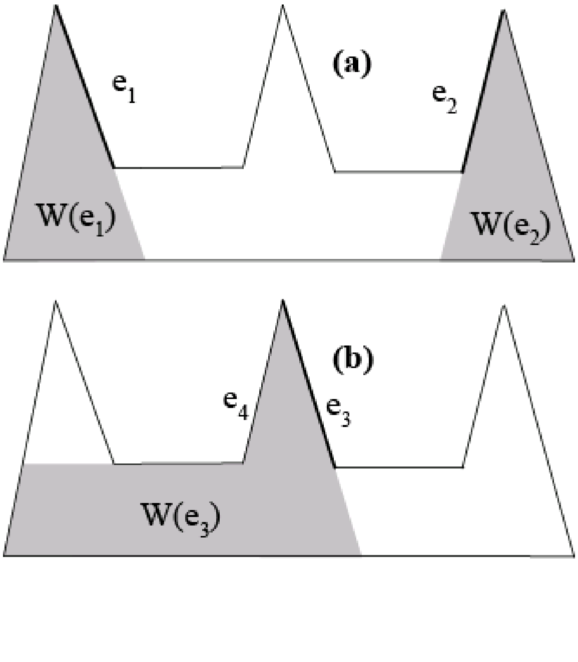

Figure 2. $LB_1(P)$ is two, but EC requires three guards

2.2. The new Lower Bound Algorithm

The previous example suggests considering visibility polygons of parts of the boundary smaller than an edge. Given a polygon *P* and recalling the definition given by O'Rourke [12], the *point visibility polygon VP(x) of a point x* is *the set of points $p \in P$ visible from x.* In particular, we focus our attention on convex vertices of the polygon and thus consider *$VP(v_i)$* of all convex vertices v_i of *P*. We only consider vertices at convex angles because they are able to produce visibility polygons smaller than those of their converging edges.

Consider the cardinality of the maximal subset of not intersecting VPs of convex vertices. It is clear that this cardinality is another lower bound, since each VP must contain at least one guard. If we use this new lower bound, the problem with the comb polygon family is solved, as shown in Fig. 3.

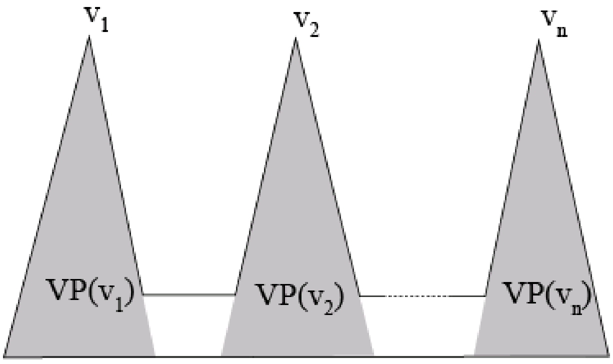

Figure 3. The non intersecting $VP(v_i)$ are as many as the guards.

However, choosing as lower bound the cardinality of the larger set of VPs of convex vertices could be not satisfactory even in relatively simple cases. Consider for instance the polygon in Fig.4. It can be easily verified that no more than four VPs of convex vertices exist, and precisely those of the vertices v_1, v_2, v_3, v_4 (Fig.4(a)). However, five EC sensors are required, located for instance as shown in Fig. 4(b).

The examples discussed suggest to take into account both weak visibility polygons of edges and point visibility polygons of convex vertices.

Then we assume the following new definition of lower bound:

The lower bound $LB_{W\&VP}(P)$ is the cardinality of the maximal subset (or subsets) of not intersecting weak visibility polygons $W(e_i)$ of edges e_i of P, and visibility polygons $VP(v_i)$ of convex vertices v_i of P.

Using this definition, the lower bound for comb polygons is the same as that shown in Fig.3. For the polygon of Fig.4, the new definition supplies five and not four as lower bound. A maximum set of non intersecting visibility polygons is shown in Fig.5. One of them is the weak visibility polygon of the edge *e*; the other polygons can be interpreted either as visibility polygons of convex vertices, or as weak visibility polygons of edges converging in these vertices. Combining polygons as those shown in Fig 2 and 4, we can easily produce examples where the new lower bound is better then those provided by weak visibility polygons and convex vertex visibility polygons separately.

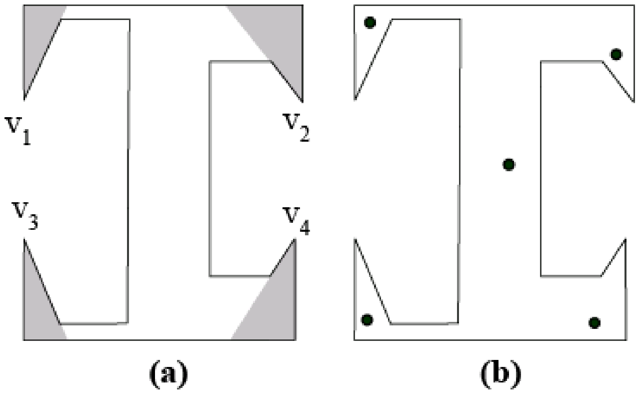

Figure 4. At most, four non intersecting VP of convex vertices can be found (a), but five guards are required (b).

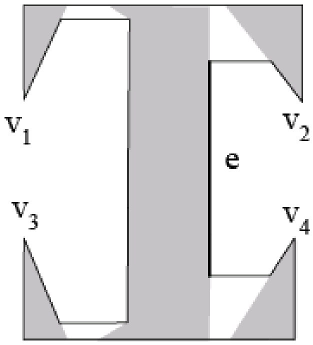

Figure 5. Five non intersecting visibility polygons are found.

In general, it is clear that $LB_{W\&VP}(P) \geq LB_W(P)$ for any P, and then $LB_{W\&VP}(P)$ is a better or equal lower bound.

Polynomial algorithms for computing point visibility polygons of polygons with and without holes can be found in O'Rourke [12]. In addition, for polygons without holes it is possible to compute the point visibility polygon of a convex vertex v_i as the intersection of the weak visibility polygons of the edges converging into v_i. In our case, visibility polygons are computed, again, as a by-product of the sensor location algorithm described in [22].

At a first glance, we could expect a heavier computational burden for the non polynomial part of the algorithm, that is the selection of the maximum independent set of vertices in the associated graph. However, as we will show in the following, this is not the case since an important reduction of the number of nodes of the graph can be performed, since the weak visibility polygons of the edges converging at the convex vertices should not be considered.

3. Experimental results

In this section, we present experimental results showing that, on the average, the new lower bound significantly outperforms the previous.

In order to evaluate the performance of $LB_{W\&VP}$ compared to LB_W, we implemented it within the EC algorithm described in [22]. Thus, two versions of the EC algorithm are considered: one is the original version (described in [22]) computing the lower bound LB_W, while the second computes the $LB_{W\&VP}$ proposed in this article. In the following, results from the original version of the EC algorithm are subscripted with *W*, while results from the new version are subscripted with *W&VP*. Comparing the old and the new LB is not sufficient for a full evaluation. A better insight is provided by the *reduction of the gap* between the lower bound and the final solution, as well as by the *proportional gap*, that is the gap divided by the lower bound, representing the percentage of total guards exceeding the LB estimation. It is also interesting to compare computational times of new and old lower bounds.

Both versions of the EC algorithm were tested over several hundreds of polygons belonging to the following five categories:

(A) generic random polygons, with edges oriented in generic directions;
(B) generic random polygons with one to three holes;
(C) orthogonal random polygons with no holes;
(D) orthogonal random polygons with one to three holes;
(E) generic random polygons with more than a hundred edges.

Four different sets of polygons, with 30, 40, 50 and 60 edges, were constructed for each of the first four categories. For the last category, three sets of polygons with 100, 150 and 200 edges were used. Test results for each category are illustrated through Table 3 to Table 7.

Each record of these tables refers to a set of *no.* polygons with *nedgs* edges used for tests. Data reported in these tables provide the following information averaged over the total number of polygons for each set:

- *LB*, the lower bound computed;
- *C,* the cardinality of the final EC solution. For polygons of the categories (A)-(D) the cardinality is given by the solution of the EC algorithm presented in [22], with four iterations without improvements and a time limit for the execution of 2400s. For polygons of category (E), the cardinality is given by the greedy solution of the EC algorithm;

- *gap (G)*, the absolute distance between the lower bound and the cardinality of the EC solution. Precisely, G_W=C_W-LB_W is the gap estimated for each polygon tested under the original version of the EC algorithm and $G_{W\&VP}$=$C_{W\&VP}$-$L_{W\&VP}$ is the gap estimated under the new version of the algorithm. The smaller is the gap, the better a solution is. Clearly, in the optimal case, the gap is null;
- *G/C*, the *gap per total number of guards* or *proportional gap;* that is, respectively, G_W/C_W and $G_{W\&VP}/C_{W\&VP}$. Relating the gap to the cardinality of a given EC solution is another way of estimating the quality of the lower bound;
- *LBtime*, the total time, in seconds, spent to compute the lower bound computation (see below for further details);
- *G reduction*, the percentage of gap reduction when using $LB_{W\&VP}$ instead of LB_W; gap reduction is defined as 1- $G_{W\&VP}/G_w$;
- *G/C reduction*, the reduction of the proportional gap given by the new lower bound;
- *LBtime reduction*, the percentage of time saved computing the lower bound as $LB_{W\&VP}$ instead of LB_W (negative values stand for extra time spent).

The experiments show that the new lower bound is on the average higher than the old one. It is important to notice that the final solutions of the two implementations are almost identical, as it can be seen from the “C” column. The difference is due to a single case (in Table 4, 50 edges’ set) where a different number of completed iterations, before reaching the time bound, produces solutions with different cardinalities.

Equal solutions, combined with an improvement of the lower bound, lead to a sensible reduction of the gap between LB and the final EC solution and, consequently, of the proportional gap. These improvements are summarized per polygon category in Table 1, where we can see that the gap reduction ranges between 27% and 42% and the proportional gap reduction ranges between 28% and 47%.

As a whole, considering all the experiments, the mean gap reduction is 33.57% and the mean proportional gap reduction is 34.29%. These results assert, beyond the shadow of a doubt, that the new lower bound presented in this article provides a tighter approximation of the optimum.

Regarding the processing time, the total time required to compute the lower bound includes:

- the *data structure time*, that is the time spent to construct the required data structure (weak visibility polygons in the case of LB_W, weak and point visibility polygons in the case of $LB_{W\&VP}$)
- the *max clique time*, which is the time taken to construct the dual graph from the set of visibility polygons and to solve the max clique problem

Polygon category	G reduction	G/C reduction
Random	42.59%	41.30%
Random with holes	27.85%	29.40%
Orthogonal	46.43%	47.84%
Orthogonal with holes	28.93%	28.47%
Random high	29.17%	29.43%
Total	33.57%	34.29%

Table 1. Total gap and proportional gap reduction per polygon category

Table 2. Lower bound reduction times

	Average reduction times		
Polygon categories	Data struct	Max clique	Total
Random	-1.24%	31.40%	10.70%
Random with holes	-3.19%	30.69%	10.72%
Orthogonal	-1.21%	21.39%	5.96%
Orthog. with holes	-15.46%	24.27%	-0.15%
Random high	-51.82%	87.72%	83.76%
Total	-29.42%	86.83%	80.90%

Processing times required for computing the data structure and solving the max clique problem were individually recorded for each polygon tested and then averaged per each polygon category. Time reductions for: 1) constructing the data structure, 2) solving the max clique problem and 3) computing the lower bound as a whole are summarized in Table 2.

The time reduction of a specific task (e.g. constructing the lower bound data structure) is the percentage of time saved by $LB_{W\&VP}$ in performing that task. Thus, positive values stand for time savings while negative values stand for extra time consumed. Table 2 reports some interesting information: first, as expected, the time spent in creating the data structure increases; second, the time spent in evaluating the max clique decreases. The first behaviour is due to the computation of the PVPs required for $LB_{W\&VP}$. The second is due to the fact that the graph combining non intersecting weak visibility polygons and point visibility polygons tends to have fewer nodes compared to the one using weak visibility polygons only. Therefore the maximum independent set problem is easier, and therefore faster, to solve. This is particularly evident for polygons with a very high number of edges. The result of this time saving is that, on the average $LBtime_{W\&VP} < LBtime_{W}$.

Summarizing, the experiments show that:

- the new lower bound is tighter and, therefore, closer to the optimum;
- the computational burden of the evaluation of the lower bound has markedly reduced.

Therefore, the new LB definitely outperforms the previous one.

4. Conclusions

We have studied, implemented and experimented a new lower bound for the minimum number of guards required for solving the edge covering problem, a variation of the art gallery problem.

In order to evaluate its performance, we compared it with a previously proposed lower bound and with the cardinality of the coverage provided by an EC algorithm.

The results collected from a wide range of polygons, with and without holes, show that the new lower bound is on the average higher/tighter than the previous one and the relative gap per total number of guards is reduced on average of almost 34.29% than the respective gap computed with LB_W. Furthermore, despite the additional computation required, the current lower bounds requires less time for its evaluation.

Concluding, the new lower bound is a significant enhancement of the previous one.

References

[1] V. Chvàtal, “A combinatorial theorem in plane geometry”, *Journal of Combinatorial Theory*, Vo.18, pp.39–41, 1975.

[2] Avis and G. T. Toussaint, “An optimal algorithm for determining the visibility polygon from an edge”, *IEEE Trans. Comput.*, Vol. C-30, No. 12, pp. 910-914, 1981.

[3] I. Bjorling-Sachs and D.Souvaine, “An efficient algorithm for guard placement in polygons with holes”, *Discrete and Computational Geometry*, pp 77-109, 1995.

[4] T. Danner and L.E. Kavraki, “Randomized planning for short inspections paths”, *IEEE Int. Conf. On Robotics and Automation*, pp. 971-976, April 2002.

[5] S. Eidenbenz, C. Stamm and P. Widmayer, “Inapproximability results for guarding polygons and terrains”, *Algorithmica*, Vol.31, pp.79-113, 2001.

[6] A. Elnagar and L.Lulu, “An Art Gallery based approach to autonomous robot motion planning in global environments”, *Proc. IEEE Int. Conf. on Intelligent Robotics and Systems*, pp.1367-1372, 2005.

[7] A. Elnagar and L.Lulu, “A comparative study between visibility-based roadmap path planning algorithms”, *Proc. IEEE Int. Conf. on Intelligent Robotics and Systems*, pp.3700-3705, 2005.

[8] H. Gonzales-Banos and J-C. Latombe, “A randomised art gallery algorithm for sensor placement”, *ACM SCG’01*, June 2001.

[9] H. Gonzales-Banos and J-C. Latombe. “Planning robot motion for range-image acquisition and automatic 3D model construction”, *Proc. AAAI Fall Symposyum Series*, 1998.

[10] G. D. Kazakakis, and A.A. Argyros, “Fast positioning of limited visibility guards for inspection of 2D workspaces”, *Proc. 2002 IEEE/RSJ Intl. Conf. On Intelligent Robots and Systems*, pp. 2843-2848, October 2002.

[11] T.S. Newman and A.K. Jain, “A survey of automated visual inspection” *Comput. Vis. Image Understand.*, Vol.61, No.2, pp.231-262, 1995.

[12] J. O'Rourke, *Art gallery theorems and algorithms*, Oxford University Press, New York, 1987.

[13] J.R. Sack and S. Suri, “An optimal algorithm for detecting weak visibility of a polygon”, IEEE Trans. on Computers, Vol. 39, pp. 1213-1219, 1990.

[14] T. Shermer, “Recent results in art galleries”, *IEEE Proc.* Vol. 80, pp. 1384-1399, 1992.

[15] W.R. Scott and G. Roth, “View planning for Automated Three-Dimensional Object Reconstruction and Inspection”, *ACM Comput. Surveys*, Vol. 35, No. 1, pp. 64-96, March 2003.

[16] K.A. Tarabanis, P.K. Allen, and R.Y. Tsai, “A survey of sensor planning in computer vision”, *IEEE Trans. Robot. and Automat.*, Vol. 11, No. 1, pp. 86 –104, 1995

[17] J. Urrutia, *Art Gallery and Illumination Problems*, Handbook on Computational Geometry, Elsevier Science Publishers, J.R. Sack and J. Urrutia eds. pp. 973-1026, 2000.

[18] P. Oestergard, “A fast algorithm for the maximum clique problem”, *Discrete Applied Mathematics*, Vol.120, pp. 197-207, 2002.

[19] D.R. Woods, “An algorithm for finding a maximum clique in a graph”, *Operations Research Letters*, Vol.21, pp. 211-217, 1997.

[20] R. Carraghan and P. Paradalos, “An exact algorithm for the maximum clique problem”, *Operations Research Letters*, 9:375-382, 1990.

[21] A. Laurentini, “Guarding the walls of an art gallery”. *The Visual Computer*, Vol. 15, pp. 265-278, 1999.

[22] A. Bottino, A. Laurentini, “Experimental results show near-optimality of a sensor location algorithm”, *Proceedings of IEEE Int. Conf. on Robotics and Biomimetics*, Kunming, China, December 2006.

nedgs	**no.**	**LB**		**C**		**G**		**G/C**		**LBtime**		**G reduction**	**G/C reduction**	**LBtime reduction**
		W&VP	*W*	*W&VP*	*W*	*W&VP*	*W*	*W&VP*	*W*	*W&VP*	*W*			
30	20	3,95	3,80	4,30	4,30	0,35	0,50	0,078	0,108	0,319	0,292	30,00%	27,91%	-9,09%
40	20	5,20	4,90	5,40	5,40	0,20	0,50	0,035	0,087	0,643	0,598	60,00%	59,84%	-7,58%
50	20	6,25	5,90	6,70	6,70	0,45	0,80	0,066	0,114	1,561	1,731	43,75%	41,88%	9,83%
60	20	7,75	7,40	8,30	8,30	0,55	0,90	0,063	0,102	2,687	3,212	38,89%	38,92%	16,37%

Table 3. Random polygons - (A)

nedgs	**no.**	**LB**		**C**		**G**		**G/C**		**LBtime**		**G reduction**	**G/C reduction**	**LBtime reduction**
		W&VP	*W*	*W&VP*	*W*	*W&VP*	*W*	*W&VP*	*W*	*W&VP*	*W*			
30	20	4,90	4,60	5,20	5,20	0,30	0,60	0,046	0,099	0,460	0,448	50,00%	52,90%	-2,83%
40	20	5,70	5,40	6,45	6,45	0,75	1,05	0,115	0,159	1,674	1,728	28,57%	27,85%	3,12%
50	20	6,75	6,40	7,55	7,45	0,80	1,05	0,106	0,137	2,106	2,345	23,81%	22,92%	10,19%
60	20	7,40	7,15	8,40	8,40	1,00	1,25	0,115	0,146	2,982	3,569	20,00%	21,30%	16,44%

Table 4. Random polygons with 1-3 holes - (B)

nedgs	**no.**	**LB**		**C**		**G**		**G/C**		**LBtime**		**G reduction**	**G/C reduction**	**LBtime reduction**
		W&VP	*W*	*W&VP*	*W*	*W&VP*	*W*	*W&VP*	*W*	*W&VP*	*W*			
30	20	4,25	3,95	4,50	4,50	0,25	0,55	0,053	0,128	0,195	0,173	54,55%	58,17%	-13,06%
40	20	5,70	5,35	6,15	6,15	0,45	0,80	0,073	0,127	0,333	0,329	43,75%	42,02%	-1,16%
50	20	6,60	6,15	7,30	7,30	0,70	1,15	0,095	0,159	2,236	1,055	39,13%	40,12%	-111,85%
60	20	8,00	7,15	8,85	8,85	0,85	1,70	0,091	0,187	1,814	3,311	50,00%	51,29%	45,22%

Table 5. Orthogonal polygons - (C)

nedgs	**no.**	**LB**		**C**		**G**		**G/C**		**LBtime**		**G reduction**	**G/C reduction**	**LBtime reduction**
		W&VP	*W*	*W&VP*	*W*	*W&VP*	*W*	*W&VP*	*W*	*W&VP*	*W*			
30	20	5,45	5,25	6,00	6,00	0,55	0,75	0,464	0,647	0,202	0,170	26,67%	28,33%	-18,85%
40	20	7,70	7,55	8,30	8,30	0,60	0,75	0,069	0,086	0,248	0,182	20,00%	19,62%	-36,38%
50	20	7,85	7,40	9,10	9,10	1,25	1,70	0,142	0,198	0,707	0,733	26,47%	28,22%	3,53%
60	20	9,45	8,50	11,35	11,35	1,90	2,85	0,165	0,243	1,402	1,471	33,33%	32,15%	4,67%

Table 6. Orthogonal polygons with 1-3 holes - (D)

nedgs	**no.**	**LB**		**C**		**G**		**G/C**		**LBtime**		**G reduction**	**G/C reduction**	**LBtime reduction**
		W&VP	*W*	*W&VP*	*W*	*W&VP*	*W*	*W&VP*	*W*	*W&VP*	*W*			
100	15	11,87	11,00	13,73	13,73	1,87	2,73	0,132	0,194	7,637	7,868	31,71%	32,28%	2,93%
150	15	17,07	16,13	19,80	19,80	2,73	3,67	0,136	0,183	32,270	36,090	25,45%	25,65%	10,59%
200	15	21,93	20,47	25,27	25,27	3,33	4,80	0,131	0,188	116,41	918,72	30,56%	30,16%	87,33%

Table 7. Random polygons with a high number of edges - (E)