# Greedy Heuristics for Solving the Weighted Orthogonal Art Gallery Problem

–

## 1 Introduction

*The Orthogonal Art Gallery Problem* (OAGP) is one of many variants of Art Gallery Problem (AGP). AGP asks for a set of points $G$ of minimal cardinality on some polygon $P$ such that for each point $y \in P$ there is $x \in G$ such that $xy \subset P$. Set $G$ is called guard set of $P$ and the points from $G$ as guards. The orthogonal AGP consider on arbitrary polygon but whose angles are $90°$ and $270°$. This problem was stated by Victor Klee in 1973. [7]. The problem is motivated from installing the cameras in a building (or gallery) such that the whole surface of the building is covered. Orthogonality constraint naturally comes out from the orthogonality of the walls in a building.

Kahn et al. in [5] formulated and proofed that $\lfloor \frac{n}{4} \rfloor$ guards are sufficient to cover an orthogonal polygon with $n$ vertices. A variant of the OAGP for which we are interested in this study allows only that guards are positioned at the vertices of polygon $P$. Actually, we are looking for minimum vertex guards needed to cover an orthogonal polygon. This problem is known to be NP- hard in [8,6].

By discretization $D(P)$ the set of points of the polygon $P$, the art gallery problem can be reduced to the well-known Min Set Cover Problem. For each vertex of the polygon $P$, a set of points of visibility is determined. On that way, the problem of determining the minimum number of guards covering the entire polygon is reduced to determining the minimum number of subsets of points, such that each point from $D(P)$ is included in at least one of the chosen subsets.

In [2] an approximate solution of the minimum vertex guard problem, which can be computed in $O(n^4)$ time and the size of the solution is at most $O(\log n)$ times the optimal, is presented for simple polygon with $n$ vertices. Firstly, this algorithm partition the polygonal region into convex components and construct sets consisting of these convex components. After that, on these constructed sets Johnson's approximation algorithm [4] for the Minimum set-covering problem (MSC) is applied to get solution.

An anytime algorithm to compute successively better approximations of the optimum to Minimum Vertex Guard is proposed in [9]. A major idea of this approach is exploring dominance of visibility regions to first detect pieces that are more difficult to guard. The same problem is solved in [10] by applying successive approximations from [9].

Tozoni et a. [11,12] presented an exact Integer linear programming (ILP)-based algorithm, which iteratively generates upper and lower bounds through the resolution of discretized space of the AGP.

More detailed overview of the results on art gallery problems is out the scope of this paper and can be found in a survey paper [3].

Couto et al. presented an exact and efficient algorithm for the Orthogonal Art Gallery Problem in [1]. The algorithm is divided into two phases. In the preprocessing phase an initial discretization is constructed and the Integer Programming model for solving MSC is applied. In the solution phase, the discretized instance is iteratively refined and solved, until the solution becomes viable.

FiXme: TODO: work on literature approaches

## 2 Preliminaries

Solvers which solves IP model for OAGP problem are one of the most efficient techniques to approach this problem. It is known that the problem is related to the known Minimal-Set-Cover (MSC) problem.

Let us suppose we are given a discretization $D(P)$ of the polygon $P$ (with a family of rectangles). Then we relate the OAGP with the known MSC problem. Family $\mathcal{F} \subseteq D(P)$ of nonempty sets is given as: $S_i \in \mathcal{F}$ iff it includes any point from $D(P)$ which is visible from guard $i \in V$. Note that set $S_i$ includes a point $p_i$ which can also included by some other guard $j \in V$, $i \neq j$. So, the task of OAPG becomes finding a minimal cardinalty cover $\mathcal{C} \subseteq \{S_1, ..., S_n\}$ of set $D(P)$, that is

$$\bigcup_{c \in \mathcal{C}} c = D(P).$$

The Integer Programming model for the Set Cover problem is known, and is as follows:

$$\sum_{i=1}^{n} x_i \longrightarrow \min \tag{1}$$

$$\text{s.t.} \tag{2}$$

$$\sum_{j \in V} a_{ij} x_j \geq 1 \ (\forall p_i \in D(P)) \tag{3}$$

$$x_j \in \{0,1\}, j \in V, \tag{4}$$

where $a_{ij} = \begin{cases} 1, p_i \in V(j), \\ 0, \text{otherwise} \end{cases}$ and $x_i = \begin{cases} 1, \text{ if the point } i \in \mathcal{C}, \\ 0, \text{otherwise}, \end{cases}$ presents a MIP model to solve OAGP where $V(j)$ is the set of all points from $D(P)$ that are visible from $j$-th vertex of $P$. Set $Z = \{j \in V \mid x_i = 1\}$ represents a solution of the problem. Constraint (3) enforces that any point $p_i \in D(P)$ will be visible from at least one guard from $Z$.

If we want to add weights into the OAGP problem such that all guard have no equal weight (of 1), we obtained the weighted variant of OAGP, that is the WOAGP problem. This can be augmented that the prices of cameras do not

**Algorithm 1** Greedy Heuristic

**Input:** an instance of a problem
**Output:** A (feasible) non-expandable solution (or reporting that no feasible solution)
$s^P \leftarrow ()$ // partial solution set to empty solution
**while** Extend($s^P$) $\neq \emptyset$ **do**
    Select component $e \in$ Extend($s^P$) // w.r.t. some criterion
    Extend $s^P$ by $e$
**end while**

need to be equal and these prices might be different due to the quality (of the range) of the cameras that are installed at some specific corners. Let us suppose that for any guard $p_i$ we assign a price $w_i$, $i = 1, ..., n$. Then, the above model is may be extended to its weighted version where the objective function has to be replaced by

$$\sum_i x_i w_i,$$

and all other constraints remain the same.

In order to solve this model, we propose to use a general purpose solver Cplex.

## 3 Algorithmic Approaches for WOAGP

Since the WOAGP can be seen as the Weighted Set Cover problem (WSCP) and since the best known heuristics to solve WSCP is an enhanced greedy heuristic, our idea is to apply greedy algorithms to solve WOAGP. These algorithms produce a solution of reasonable quality within a short interval of time. Efficient of greedy heuristics is related to a greedy criterion utilized to expand current (non-complete, that is partial) solution to complete one. Among all candidates ( solution components for expansion, that is not-yet-considered guards, we choose one with the smallest greedy value etc. until the solution is complete (i.e., cover all regions of the polygon). Pseudocode of Greedy heuristic is given in Algorithm 1

### 3.1 Greedy Heuristic based on Price-per-Unit

Our greedy criterion to expand current partial solution $s^P$ is based on the WOAGP characteristics, that is considering not-yet-covered regions of polygon $P$, that is, its discretisation $D(P)$. For each not yet considered guards $p_i$, we assign the region he is able to see by $S_{p_i}$. As the next candidate to extend $s^P$, we choose not-yet-considered guard $p^*$ which is able to cover his part $S_{p_i}$ by smallest price per unit, among the other not considered guards. More precisely, greedy criterion $g$ is given as:

$$g(s^p, p_i) = \frac{w_{p_i}}{S_{p_i}}, \tag{5}$$

and $p^*$ that minimizes $g(s^p, p^*)$ value is chosen as an extension of the current partial solution. Afterwards the region $S_{p^*}$ is further then dropped off from $P$, i.e., $P = P \setminus S_{p^*}$, and update $s^p = s^p \cup \{p^*\}$. These steps are repeated until $s^P$ is complete, which means $P = \emptyset$ (polygon $P$ is covered by the guards from $s^P$).

The best WSC problem greedy heuristic known from the literature was based on the following greedy heuristic:

$$g(s^p, p_i) = \frac{w_{p_i}}{f(s^p \cup \{p_i\}) - f(s^p)}, \tag{6}$$

where $f(s^p) = |\bigcup_{s \in s^p} s|$.

### 3.2 An Alternative Greedy Heuristic

FiXme: TODO: work in progress... Greedy criterion:

- prefer those guards with a smaller cost $w_i$;
- If there are more guards with the smallest cost, prefer one with the largest $S_{p_i}$.

FiXme: TODO: work in progress... Another approach DRAGAN:

- introduce a penalty function
- introduce a term "incorrect point". For a point from D(P) we say that it is incorrect if it is not covered by any guard from a current partial solution $s^{ps}$.
- let $incorrect_{total}$ be the total number of incorrect points from discretisazion $D(P)$
- let $w_{total}$ be the total sum of all weights among all verteces
- involve the penalty function in the objective function

  $$obj(s^{ps}; v) = \sum_{i \in s^{ps} \cup \{v\}} w_i + incorrect_{total}$$

  or

  $$obj(s^{ps}; v) = \frac{\sum_{i \in s^{ps} \cup \{v\}} w_i}{w_{total}} + \frac{incorrect_{total}}{|D(P)|}$$

  FiXme: moguce i skaliranje ove dvije vrijednosti?
- start Greedy with empty solution
- in each iteration add such a guard to $s^{ps}$ for which the obj value is minimal
- end when $incorrect_{total}$ becomes 0

### 3.3 An improvement of Greedy Solution

Local search and Large Neighborhood search FiXme: TODO... Local search (LS) technique is a basis of the LS–based metaheuristics. Let us define the structure of neighborhood. We define them over the set of guards. The first configuration of neighbourhood of some partial solution is operation of swapping; that means

**Algorithm 2** Local Search + Greedy

**Input:** and Instance of the problem; $pct \in (0, 1)$:percentage
**Output:** a complete solution $s$
$s_{greedy} \leftarrow \emptyset$
**while** $s_{greedy}$ is not complete **do**
  $s_{greedy} \leftarrow$ call Greedy to expend $s_{greedy}$ for (up to) $\lfloor n \cdot pct \rfloor$ guards
  $s_{greedy} \leftarrow$ LS($s_{greedy}$)
**end while**

from a given partial solution $s^P$, we choose one guard from $s^P$ and then pick another guard from $V \setminus s^p$ w.r.t. value $obj(;)$.

Another criterion for the neighbourhood's configuration ($k$-th) would be taking arbitrary $k$ guards $\{p_1, ..., p_d\}$ from $s^p$, then drop them from $s^p$. Afterwords, for each $p_i$ which are dropped off, we take the guard which is the nearest by distance from $p_i$ and add them into $s^p$.

### 3.4 Hybrid of the Greed Heuristic and Local Search

In order to improve the results of the Greedy heuristic, we propose a hybridization with the LS method. The pseudocode is given in Algorithm 2.

## 4 Computational Results

We used the instance of a specific OAGP and assigned the weights to each vertex of polygons. We have generated three kind of benchmarks:

- *random benchmarks.* For each vertex $i$ of polygon $P$ we take a random value $w_i \in \{X_1, ..., X_q\}$ FiXme: da li je domen ovdje OK? as its weight ($X_i$ are some random values for prices), $q \in \mathbb{N}$.
- *topological-type benchmarks.* For each vertex $i$ of polygon $P$ let us denote by $l_i$ and $l_{i+1}$ the lengths of edges that comes out of vertex $i$. Then, $w_i := \frac{l_i + l_{i+1}}{2}$. This can be augmented by the fact that if the the arithmetic length of both edges that comes out of vertex $i$ is longer, it is expected that vertex is a guard can see a larger pieces of polygon $P$. This implies that the range of camera $i$ has to be larger, which again means that it has to be of a higher price.
- ...

## 5 Conclusions and Future Work

## References

1. M. C. Couto, C. C. De Souza, and P. J. De Rezende. An exact and efficient algorithm for the orthogonal art gallery problem. In *XX Brazilian Symposium on Computer Graphics and Image Processing (SIBGRAPI 2007)*, pages 87–94. IEEE, 2007.

**Algorithm 3** Local Search

**Input:** a (partial) solution $s_{greedy} = \{v_1, ..., v_k\}$, $iter_{allow} > 0.$, probability $p$
**Output:** and improved solution $s$
**while** $iter < iter_{allow}$ **do**
    $D(P)_s$ the number of points covered by $s_{greedy}$
    **if** $|D(P)_s| == |D(P)|$ **then** // a complete solution
        return $s_{greedy}$
    **end if**
    choose random vertex $v_{i'}$ from $s_{greedy}$
    $v_{i''}, v_{i'''} \leftarrow$ two neighbor vertices of vertex $v_{i'}$
    $s^1_{greedy} \leftarrow s_{greedy} \cup \{v_{i''}\} \setminus \{v_{i'}\}$
    $s^2_{greedy} \leftarrow s_{greedy} \cup \{v_{i'''}\} \setminus \{v_{i'}\}$
    **if** $|D(P)_{s^1_{greedy}}| > |D(P)_{s_{greedy}}|$ **then**
        $s_{greedy} \leftarrow s^1_{greedy}$
    **end if**
    **if** $|D(P)_{s^1_{greedy}}| == |D(P)_{s_{greedy}}|$ **then**
        $s_{greedy} \leftarrow s^1_{greedy}$ with probability $p$
    **end if**
    **if** $|D(P)_{s^2_{greedy}}| > |D(P)_{s_{greedy}}|$ **then**
        $s_{greedy} \leftarrow s^2_{greedy}$
    **end if**
    **if** $|D(P)_{s^2_{greedy}}| == |D(P)_{s_{greedy}}|$ **then**
        $s_{greedy} \leftarrow s^2_{greedy}$ with probability $p$
    **end if**
    $iter \leftarrow iter + 1$
**end while**
return $s_{greedy}$


2. S. K. Ghosh. Approximation algorithms for art gallery problems in polygons. *Discrete Applied Mathematics*, 158(6):718–722, 2010.
3. S. K. Ghosh. Approximation algorithms for art gallery problems in polygons and terrains. In *International Workshop on Algorithms and Computation*, pages 21–34. Springer, 2010.
4. D. S. Johnson. Approximation algorithms for combinatorial problems. *Journal of computer and system sciences*, 9(3):256–278, 1974.
5. J. Kahn, M. Klawe, and D. Kleitman. Traditional galleries require fewer watchmen. *SIAM Journal on Algebraic Discrete Methods*, 4(2):194–206, 1983.
6. M. J. Katz and G. S. Roisman. On guarding the vertices of rectilinear domains. *Computational Geometry*, 39(3):219–228, 2008.
7. J. O'rourke. *Art gallery theorems and algorithms*, volume 57. Oxford University Press Oxford, 1987.
8. D. Schuchardt and H.-D. Hecker. Two np-hard art-gallery problems for ortho-polygons. *Mathematical Logic Quarterly*, 41(2):261–267, 1995.
9. A. P. Tomás, A. L. Bajuelos, and F. Marques. Approximation algorithms to minimum vertex cover problems on polygons and terrains. In *International Conference on Computational Science*, pages 869–878. Springer, 2003.

10. A. P. Tomás, A. L. Bajuelos, and F. Marques. On visibility problems in the plane–solving minimum vertex guard problems by successive approximations. In *ISAIM*, 2006.
11. D. C. Tozoni, P. J. de Rezende, and C. C. de Souza. A practical iterative algorithm for the art gallery problem using integer linear programming. *Optimization Online*, pages 1–21, 2013.
12. D. C. Tozoni, P. J. D. Rezende, and C. C. D. Souza. Algorithm 966: a practical iterative algorithm for the art gallery problem using integer linear programming. *ACM Transactions on Mathematical Software (TOMS)*, 43(2):1–27, 2016.