Illuminating the x-Axis by α-Floodlights

Bengt J. Nilsson
Department of Computer Science and Media Technology, Malmö University, Sweden

David Orden
Physics and Mathematics Department, Universidad Alcalá, Spain

Leonidas Palios
Department of Computer Science and Engineering, University of Ioannina, Greece

Carlos Seara
Departament of Mathematics, Universitat Politècnica de Catalunya, Barcelona, Spain

Paweł Żyliński
Institute of Informatics, University of Gdańsk, Poland

Abstract

Given a set S of regions with piece-wise linear boundary and a positive angle $\alpha < 90°$, we consider the problem of computing the locations and orientations of the minimum number of α-floodlights positioned at points in S which suffice to illuminate the entire x-axis. We show that the problem can be solved in $O(n \log n)$ time and $O(n)$ space, where n is the number of vertices of the set S.

2012 ACM Subject Classification Theory of computation → Computational geometry

Keywords and phrases Computational Geometry, Visibility, Art Gallery Problems, Floodlights

Digital Object Identifier 10.4230/LIPIcs.ISAAC.2021.11

Funding *Bengt J. Nilsson*: Grant 2018-04001 from the Swedish Research Council.
David Orden: H2020-MSCA-RISE project 734922-CONNECT and project PID2019-104129GB-I00 funded by MCIN/ AEI/ 10.13039/501100011033.
Carlos Seara: H2020-MSCA-RISE project 734922-CONNECT and projects PID2019-104129GB-I00/ AEI/ 10.13039/501100011033 and Gen. Cat. DGR 2017SGR1640.

1 Introduction

An α-*floodlight* is a two-dimensional floodlight whose illumination cone angle is equal to a positive angle α. We are interested in using the minimum number of α-floodlights positioned at points of a given set S in the plane in order to illuminate the entire x-axis; in particular, we consider that S is a collection of regions with piece-wise linear boundary which may degenerate into a point. We assume that no point of S lies on the x-axis (otherwise, at most two floodlights would suffice for any value of α) and that the entire S lies in the halfplane above the x-axis (any point of S below the x-axis can be equivalently reflected about the x-axis into the halfplane above the x-axis). Next, regarding the angle α of the α-floodlights, we consider that $\alpha < 90°$ because for $\alpha \geq 90°$ the problem admits a trivial solution: if $90° \leq \alpha < 180°$ then two floodlights are necessary and sufficient to illuminate the entire x-axis, and if $\alpha \geq 180°$ then one floodlight is necessary and sufficient. Thus, in this paper we focus on the following problem.

The Axis α-Illumination Problem

Given a set S of regions with piece-wise linear boundary above the x-axis and a positive angle $\alpha < 90°$, compute the locations and orientations of the minimum number of α-floodlights positioned at points in S which suffice to illuminate the entire x-axis.

© Bengt J. Nilsson, David Orden, Leonidas Palios, Carlos Seara, and Paweł Żyliński;
licensed under Creative Commons License CC-BY 4.0
32nd International Symposium on Algorithms and Computation (ISAAC 2021).
Editors: Hee-Kap Ahn and Kunihiko Sadakane; Article No. 11; pp. 11:1–11:12
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

As the number of required α-floodlights can be very large even for a set S of small descriptive size, we designate that a solution to an instance of the Axis α-floodlight Problem is a set of pairs (t_i, R_i) where $t_i \in S$ and R_i is a maximal continuous range of the x-axis to be illuminated by α-floodlights positioned at t_i such that (i) the union of all the ranges R_i is equal to the x-axis and (ii) over all pairs, the sum $\sum_i \left\lceil \frac{angle(R_i)}{\alpha} \right\rceil$ is minimized where $angle(R_i)$ is the angle subtended by the range R_i from t_i; this sum is precisely the total number of α-floodlights needed to illuminate the entire x-axis. We note that there may be more than one pair associated with a location $t_i \in S$ but if it is so, then the corresponding x-axis ranges do not intersect; see Figure 1(right). As we show (Corollary 12), the size of such a solution is at most linear in the number of vertices of the given set S.

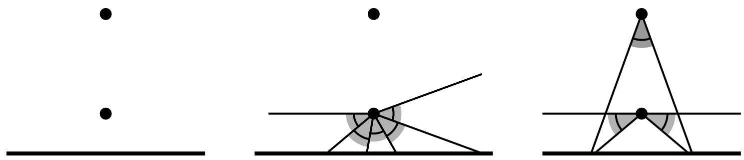

Figure 1 (left) An instance of the Axis α-Illumination Problem for two floodlight locations and $\alpha = 40°$; (middle) A solution with five floodlights; (right) A solution with three floodlights.

Clearly, any instance of the Axis α-floodlight Problem admits a solution since a single point in the set S can illuminate the entire x-axis by using $\left\lceil \frac{180°}{\alpha} \right\rceil$ α-floodlights; see Figure 1(middle) where five 40°-floodlights can be used to illuminate the x-axis. Yet, the minimum number of needed floodlights may be much smaller; in Figure 1(right), for the given set S containing two locations, three 40°-floodlights suffice to illuminate the entire x-axis.

Our Contribution. In this paper, we present an algorithm to solve the Axis α-Illumination Problem. Our algorithm runs in $O(n \log n)$ time where n is the number of vertices of the given set S of potential floodlight locations. Our algorithm can be used to illuminate arbitrary lines as well as line segments.

Related Work. Floodlight illumination problems are considered a prominent class in Computational Geometry [14, 20] and find applications in the field of directional sensor networks [19]. The seminal *Stage Illumination Problem* [3] was posed by Urrutia in 1992 and later proved to be NP-complete even with some restrictions by Ito et al. [11]. This problem takes as inputs a line segment and a set of floodlights F_i with angles α_i and apexes at predetermined locations p_i on the same side of the segment, the goal being to rotate the floodlights around their positions in such a way that the segment is completely illuminated. Even more related to our work is the problem of the *Optimal Floodlight Illumination of the Real Line* [5], which takes as inputs a line and a set S of n points, with the goal being to determine a finite set of floodlights F_j of arbitrary angles α_j and with apexes at $p_j \in S$ (more than one floodlight can have the same point as apex) such that the line is illuminated and the sum of the angles α_j is the smallest possible. This problem was shown to be solvable in $\Theta(n \log n)$ time [5]. Further, the *Optimal Floodlight Illumination of a Stage*, similar to the previous one but considering a segment instead of a line, has also been considered and solved in $\Theta(n \log n)$ time even if no more than one floodlight are allowed to have the same point as apex [7]. On the other hand, the problem of whether a polygon can be illuminated by a given number of α-floodlights is NP-hard and APX-hard [1].

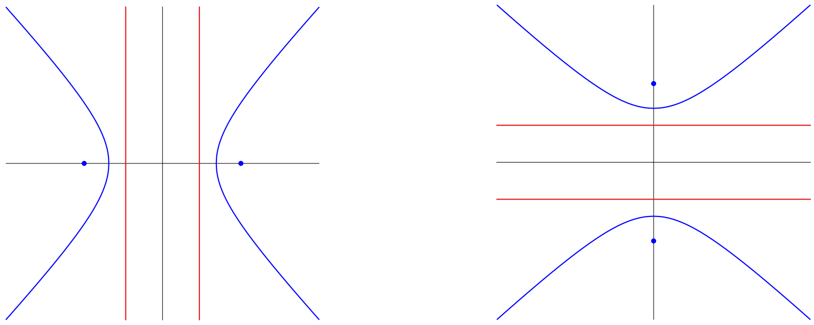

Figure 2 (left) A hyperbola with its foci (shown in blue) on a horizontal line; (right) A hyperbola with its foci (shown in blue) on a vertical line. The red lines are the directrices of the hyperbola.

Finally, a wider perspective locates our problem as a variant of the Art Gallery Problem, originally posed by Klee in 1973 as the question of determining the minimum number of guards sufficient to see every point of the interior of a simple polygon; for more details, see the book by O'Rourke [13], the survey article by Shermer [15], and the book chapter by Urrutia [20]. Among all these variants, we only mention just a few, chronologically, those the most related to our problem: the searchlight problem in polygons [18], floodlight illumination of the plane [3], floodlight illumination of polygons [8], the two-floodlight illumination problem [9], floodlight illumination of wedges [17], continuous surveillance of points by rotating floodlights [2], and monitoring the plane with rotating radars [4].

2 Preliminaries

Our algorithm relies on the use of hyperbola arcs and of the farthest-point Voronoi diagram of a point set in the plane. So, we present the definition and useful properties of these two notions.

Hyperbola. A *hyperbola* with *foci* f_1 and f_2 is the locus of the points in the plane such that the absolute value of the difference of their distances from f_1 and f_2 is constant (and less than the distance of the foci) [21]; see Figure 2(left). Most commonly, the foci are located at $(x_0 - \mathtt{c}, y_0)$ and $(x_0 + \mathtt{c}, y_0)$, where $\mathtt{c} > 0$. Then, if the absolute value of the difference of the distances from the foci is equal to $2\mathtt{a}$ with $0 < \mathtt{a} < \mathtt{c}$, the expression of the hyperbola is

$$\frac{(x - x_0)^2}{\mathtt{a}^2} - \frac{(y - y_0)^2}{\mathtt{c}^2 - \mathtt{a}^2} = 1. \tag{1}$$

Such a hyperbola consists of two branches separated by any vertical line $x = x_0 + \delta$ where $-\mathtt{a} < \delta < \mathtt{a}$. In fact, these branches can also be defined in terms of the two *directrices* of the hyperbola which, in this case, are vertical lines; the directrices are symmetrically positioned about the *center* (x_0, y_0) of the hyperbola at distance $\mathtt{a}^2/\mathtt{c}$ from it. Then, each *hyperbola branch* is the locus of points whose distance from one of the foci divided by the (perpendicular) distance from the corresponding directrix is greater than 1; for the hyperbola satisfying Equation (1), this ratio of distances is equal to $\mathtt{c}/\mathtt{a} > 1$.

If the foci are located at $(x_0, y_0 - \mathtt{c})$ and $(x_0, y_0 + \mathtt{c})$ (i.e., the foci are on a *vertical* line), we have a symmetric case by "exchanging" the x- and y-axis, as shown in Figure 2(right).

Farthest-point Voronoi diagram. The farthest-point Voronoi diagram of a planar point set will also be exploited in our algorithm. For a given set of points $S = \{p_1, p_2, ..., p_n\}$ in the plane, typically called *sites*, the *farthest-point Voronoi diagram* of S divides the plane into cells such that each cell contains all the points of the plane with the same farthest site among the sites in S [6]. It is well known that:

▶ **Lemma 1** ([6]).

(i) *A point of a point-set S has a cell in the farthest-point Voronoi diagram of S if and only if it is a vertex of the convex hull of S.*

(ii) *The farthest-point Voronoi diagram of n points in the plane has $O(n)$ vertices, edges, and cells.*

As a result of Lemma 1(i), all cells of a planar farthest-point Voronoi diagram are unbounded and its vertices and edges form a tree-like structure (Figure 9). Additionally, the definition of the farthest-point Voronoi diagram readily implies the following corollary.

▶ **Corollary 2.** *Let $fVD(S)$ be the farthest-point Voronoi diagram of a planar point-set S and let $V(p)$ be the cell of $p \in S$ in $fVD(S)$. Then, a point r belongs to the closure of $V(p)$ if and only if the entire S is enclosed by the circle with center r and radius the distance of r to p.*

3 Illuminating unbounded and bounded ranges of the x-axis

In this section, we present how to efficiently illuminate unbounded and bounded ranges of the x-axis and we introduce some useful notation. We first consider unbounded ranges of the x-axis, i.e., a range $(-\infty, \chi]$ or a range $[\chi, +\infty)$. Then, the definition of an α-floodlight implies the following observation.

▶ **Observation 3.** *Let t be a two-dimensional point above the x-axis. The illumination cone of any α-floodlight positioned at t and illuminating the maximum range $(-\infty, \chi]$ of the x-axis (i.e., χ is maximized) is delimited by the t-originating leftward-pointing horizontal ray and the t-originating downward-pointing ray that forms an angle equal to α with the positive x-axis; see Figure 3. Symmetrically, the illumination cone of any α-floodlight positioned at t and illuminating the maximum range $[\chi', +\infty)$ of the x-axis (i.e., χ' is minimized) is delimited by the t-originating rightward-pointing horizontal ray and the t-originating downward-pointing ray that forms an angle equal to $180° - \alpha$ with the positive x-axis.*

In light of this observation, in the next lemma we show how to efficiently illuminate the unbounded ranges of the x-axis.

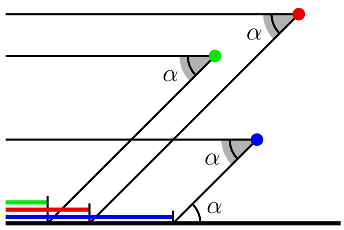

Figure 3 Illuminating a maximum x-axis range $(-\infty, \chi]$.

▶ **Lemma 4.** *For the given set S of potential floodlight locations, let $t \in S$ lie on the line supporting S from below and forming angle α with the positive x-axis. Then, the maximum range $(-\infty, \chi_t]$ of the x-axis illuminated by an α-floodlight positioned at t is no smaller than the maximum range $(-\infty, \chi]$ of the x-axis illuminated by an α-floodlight positioned at any other point in S; see Figure 3. A symmetric result holds for the illumination of a range $[\chi', +\infty)$ of the x-axis where the points in S that maximize the illuminated range are those that lie on the line supporting S from below and forming angle $180° - \alpha$ with the positive x-axis.*

Next, let $(-\infty, \chi_{left}]$ and $[\chi_{right}, +\infty)$ be the maximum such ranges of the x-axis illuminated by an α-floodlight positioned at any point in the given location set S; the values χ_{left} and χ_{right} can be computed as described in Lemma 4. Note that $\chi_{left} < \chi_{right}$ because $\alpha < 90°$.

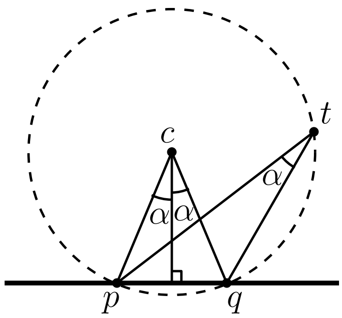

Figure 4 Illuminating a bounded range of the x-axis with an α-floodlight positioned at t.

Now, let us consider the illumination of bounded ranges of the x-axis. If an α-floodlight positioned at a point t above the x-axis illuminates the range $[\chi_L, \chi_R]$ of the x-axis with $-\infty < \chi_L < \chi_{right}$, we write that $illum(t, \chi_L) = [\chi_L, \chi_R]$; the range $illum(t, \chi_L)$ is uniquely defined as the range of the x-axis swept by the counterclockwise rotation by an angle α of the t-originating ray that goes through the point $(\chi_L, 0)$ on the x-axis. In addition, we extend this notation and use $illum(t, -\infty)$ to denote the maximum range of the form $(-\infty, \chi]$ illuminated by an α-floodlight at t; see Observation 3.

Assuming that $illum(t, \chi_L) = [\chi_L, \chi_R]$, let p, q be the points $p = (\chi_L, 0)$ and $q = (\chi_R, 0)$ and let r be the radius of the circle through t, p, q (see Figure 4). Then, the distance of the center c of this circle from the x-axis is $r \cos \alpha$, that is, the ratio of its distance to t over its distance to the x-axis is $1/\cos \alpha$, which is a constant greater than 1 for any fixed $\alpha < 90°$. Therefore, from Section 2, the center c belongs to a hyperbola branch which we formally define next.

▶ **Definition 5.** *For any positive angle $\alpha < 90°$ and any point t above the x-axis, we define the* hyperbola branch H_t *(with the point t as focus and the x-axis as directrix) that is the locus of points whose distance from the point t divided by the (perpendicular) distance from the x-axis is equal to $1/\cos \alpha > 1$.*

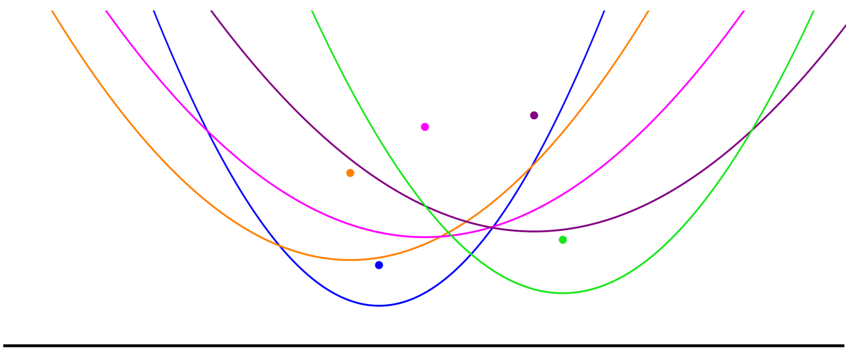

Figure 5 A set of five point locations and the corresponding hyperbola branches H_i for $\alpha = 10°$.

Figure 5 shows the hyperbola branches H_t for a set of five points and $\alpha = 10°$. Then, from the above discussion and from Figure 4, we have:

▶ **Lemma 6.** *Let t be a point above the x-axis and α a positive angle where $\alpha < 90°$. Consider an α-floodlight positioned at t which illuminates a range $[\chi_L, \chi_R]$ of the x-axis, and let p, q be the points $p = (\chi_L, 0)$ and $q = (\chi_R, 0)$ on the x-axis. If C is the circle defined by t, p, q, then:*

(i) *The center c of the circle C lies on the hyperbola branch H_t (see Definition 5).*

(ii) *The line through p and c forms a fixed angle equal to $90° - \alpha$ with the positive x-axis. Symmetrically, the line through q and c forms a fixed angle equal to $90° + \alpha$ with the positive x-axis.*

As we are interested in minimizing the total number of α-floodlights used, it is important to use floodlights positioned at points of the given set S of locations such that the illuminated range is maximized. Thus, for $-\infty \leq \chi < \chi_{right}$, we denote by $loc\_max(\chi) \subseteq S$ the set of locations $t \in S$ such that $illum(t, \chi)$ is maximized; Lemma 4 implies that if $\chi = -\infty$, all these locations belong to a line forming angle α with the positive x-axis whereas Figure 4 implies that for $-\infty < \chi < \chi_{right}$, they belong to an arc of a circle with endpoints being the endpoints of $illum(t, \chi)$.

In order to determine the points in $loc\_max(\chi)$ for χ such that $\chi_{left} \leq \chi < \chi_{right}$, we use the properties stated in the following lemma.

▶ **Lemma 7.** *Let t be a point in the location set S. For any real number χ such that $\chi_{left} \leq \chi < \chi_{right}$, assume that $illum(t, \chi) = [\chi, \chi_R]$ and let p, q be the points $p = (\chi, 0)$ and $q = (\chi_R, 0)$ on the x-axis. If c is the center of the circle C through t, p, q, then:*

(i) *The point t belongs to $loc\_max(\chi)$ if and only if the circle C encloses the entire set S.*

(ii) *The point t belongs to $loc\_max(\chi)$ if and only if c belongs to the upper envelope of the hyperbola branches H_u for all $u \in S$.*

(iii) *If t belongs to $loc\_max(\chi)$, then no range $[\chi'_L, \chi'_R]$ with $\chi'_L \leq \chi$ and $\chi'_R > \chi_R$ can be illuminated by an α-floodlight located at any point of S.*

4 The Algorithm

First, we show that the Axis α-Illumination Problem admits solutions in which we use one floodlight to illuminate the range $(-\infty, \chi_{left}]$ and another one to illuminate the range $[\chi_{right}, +\infty)$ (Figure 6) where $\chi_{left}, \chi_{right}$ are as defined in Section 3.

▶ **Lemma 8.** *Let S be a given set of locations and let $\alpha < 90°$. Then, there exists a solution of the Axis α-Illumination Problem on S in which one floodlight is used to illuminate the range $(-\infty, \chi_{left}]$ and another one to illuminate the range $[\chi_{right}, +\infty)$ of the x-axis.*

Therefore, next, we concentrate on the illumination of the range $[\chi_{left}, \chi_{right}]$ of the x-axis. First, we note that, as is the case in [5], any instance of the Axis α-Illumination Problem on a (possibly continuous) location set S can be reduced into an instance on the set of the convex hull vertices of S, that is, on a discrete location set.

▶ **Lemma 9.** *The Axis α-Illumination Problem on a set S of planar regions with piece-wise linear boundary has the same solution as the Axis α-Illumination Problem on the vertices of the convex hull $CH(S)$ of S.*

Because of Lemma 9, in the following, we can assume that S is a discrete set, with $|S| = n$.

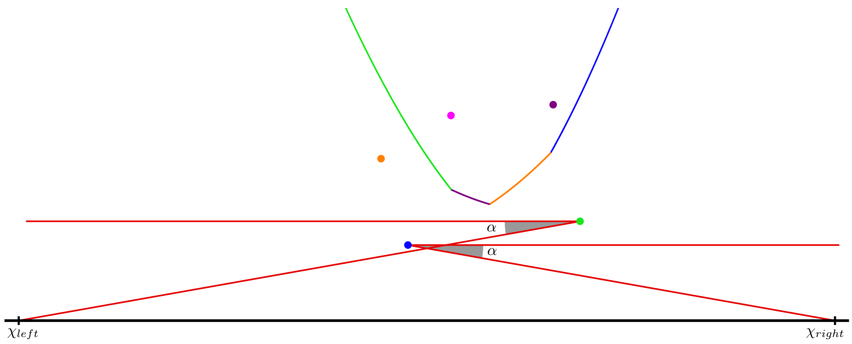

Figure 6 The upper envelope H of the hyperbola branches in Figure 5 and the α-floodlights (shown in red) illuminating the ranges $(-\infty, \chi_{left}]$ and $[\chi_{right}, +\infty)$ of the x-axis.

For the illumination of the range $[\chi_{left}, \chi_{right}]$ of the x-axis, Lemma 7(*iii*) suggests that it is advantageous to use locations that belong to the corresponding $loc\_max(\cdot)$ set. This is what we do in First Algorithm. The algorithm determines a location in the current $loc\_max(\chi)$ set (for $\chi_{left} \leq \chi < \chi_{right}$) from the intersection of the upper envelope H of the hyperbola branches with a line forming angle $90° - \alpha$ with the positive x-axis that goes through point $(\chi, 0)$ on the x-axis (see Lemma 6(*ii*)); any such line intersects a hyperbola branch H_t at exactly one point.

First Algorithm

Input: a positive angle $\alpha < 90°$, a set S of regions with piece-wise linear boundary above the x-axis, and the range $[\chi_{left}, \chi_{right}]$ of the x-axis to be illuminated
Output: a set F of α-floodlights at points in S illuminating the entire $[\chi_{left}, \chi_{right}]$, and the corresponding illuminated ranges

---

1. $F \leftarrow \emptyset$; $\{F$ *will store a solution*$\}$
  $current\_\chi \leftarrow \chi_{left}$;
  while $current\_\chi < \chi_{right}$ **do**
    $p \leftarrow$ the point $(current\_\chi, 0)$ on the x-axis;
    $v \leftarrow$ a vertex in $loc\_max(current\_\chi)$;
    $q \leftarrow$ point on the x-axis to the right of p such that the angle $\widehat{pvq}$ is equal to α;
    $\chi' \leftarrow$ the x-coordinate of q;
    $F \leftarrow F \cup \{ (v, [current\_\chi, \chi']) \}$;
    $current\_\chi \leftarrow \chi'$;
  return the resulting F;

Clearly, the algorithm can be used to illuminate any subset of the range $[\chi_{left}, \chi_{right}]$ of the x-axis, and in general any line segment in the plane. First Algorithm places an α-floodlight at a point in $loc\_max(\chi_{left})$, which is computed from the arc in the upper envelope H that is intersected by the line through the point $(\chi_{left}, 0)$ and forming angle $90° - \alpha$ with the positive x-axis (Lemma 6 and Lemma 7(*ii*)), and if the illuminated range is $[\chi_{left}, \chi_1]$, in a similar fashion, places an α-floodlight at a point in $loc\_max(\chi_1)$, and if the new illuminated range is $[\chi_1, \chi_2]$, it places an α-floodlight at a point in $loc\_max(\chi_2)$, and so on so forth until χ_{right} gets illuminated. Figure 7 shows how Step 2 of the First Algorithm works in order to illuminate the range $[\chi_{left}, \chi_{right}]$ of the x-axis for the set of five point locations of Figure 5.

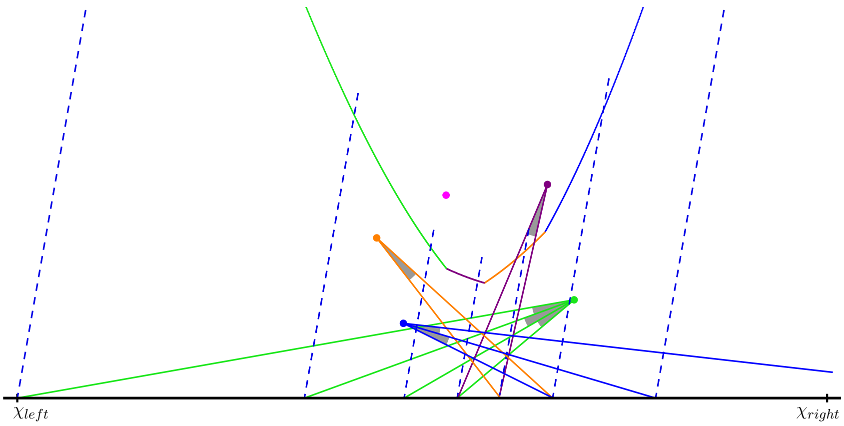

Figure 7 Illustration of the operation of First Algorithm on the hyperbola branches of Figure 5 and the set of floodlights produced by the algorithm to optimally illuminate the range $[\chi_{left}, \chi_{right}]$ of the x-axis.

If the upper envelope H of the hyperbola branches H_t of the vertices of the convex hull $CH(S)$ of the location set S is given as a left-to-right sequence of hyperbola arcs along with the associated vertices of $CH(S)$, First Algorithm takes $O(n+k)$ time where k is the number of α-floodlights that are eventually used. If the output consists of a list of the required floodlights, this algorithm is output-size sensitive. However, the number k may be very large even for a location set S of small descriptive size.

In the following, we propose a modified approach which, in one fell swoop, computes a number of illuminated ranges corresponding to many α-floodlights placed at the same location in S; once we reach an arc A of the upper envelope H, we determine all consecutive α-floodlights that need to be placed at the convex-hull vertex corresponding to the arc A by using the right endpoint of A. This approach is used in Step 2 of Second Algorithm for the general Axis α-Illumination Problem, which we give below.

Second Algorithm

Input: a positive angle $\alpha < 90°$ and a set S of regions with piece-wise linear boundary above the x-axis
Output: a set F of α-floodlights located at points in S illuminating the entire x-axis, and the corresponding illuminated ranges on the x-axis

---

1. compute the convex hull $CH(S)$ of the given location set S;
compute the upper envelope H of the hyperbola branches (each defined by a vertex in $CH(S)$ and the x-axis as directrix) and store it as a left-to-right sequence of hyperbola arcs, each associated with the corresponding vertex in $CH(S)$;
$v_{first} \leftarrow$ a vertex of $CH(S)$ as described in Lemma 4 in order to place an α-floodlight to illuminate the range $(-\infty, \chi_{left}]$ of the x-axis;
$v_{last} \leftarrow$ a vertex of $CH(S)$ as described in Lemma 4 in order to place an α-floodlight to illuminate the range $[\chi_{right}, +\infty)$ of the x-axis;
$F \leftarrow \{ (v_{first}, (-\infty, \chi_{left}]) \}$; $\{F$ *will store a solution*$\}$

continued on next page...

Second Algorithm *(continued)*

2. $current\_\chi \leftarrow \chi_{left}$;
while $current\_\chi < \chi_{right}$ **do**
  $p \leftarrow$ the point $(current\_\chi, 0)$ on the x-axis;
  $L \leftarrow$ the line through p forming angle $90^\circ - \alpha$ with the positive x-axis;
  $A \leftarrow$ the hyperbola arc (in the upper envelope H) intersected by the line L;
  $v \leftarrow$ the vertex in $CH(S)$ associated with A;
  if the arc A has a right endpoint in H **then**
    $L' \leftarrow$ the line through the right endpoint of A forming angle $90^\circ + \alpha$ with the positive x-axis;
    $q' \leftarrow$ the point of intersection of the line L' with the x-axis;
    $k \leftarrow \lfloor \widehat{pvq'}/\alpha \rfloor$; {*we do not go past the right endpoint of A*}
  else {*the arc A is the rightmost arc in H*}
    $q' \leftarrow$ the point $(\chi_{right}, 0)$ on the x-axis;
    $k \leftarrow \lceil \widehat{pvq'}/\alpha \rceil$; {*we reach the point $(\chi_{right}, 0)$*}
  $q \leftarrow$ the point on the x-axis such that the angle $\widehat{pvq}$ is equal to $k \cdot \alpha$;
  $\chi' \leftarrow$ the x-coordinate of q;
  $F \leftarrow F \cup \{ (v, [current\_\chi, \chi']) \}$;
  $current\_\chi \leftarrow \chi'$;
$F \leftarrow F \cup \{ (v_{last}, [\chi_{right}, +\infty)) \}$;

3. in the solution F, merge any pairs associated with the same vertex and with touching ranges of the x-axis;
return the resulting F;

An example of how Step 2 of Second Algorithm works to illuminate the range $[\chi_{left}, \chi_{right}]$ of the x-axis is presented in Figure 8. It is important to note that a single iteration of the `while` loop in Step 2 of the Second Algorithm corresponds to several iterations of the `while` loop in Step 2 of the First Algorithm for the same floodlight location and for touching illumination ranges; to see this, observe that in the `while` loop in Step 2 of the Second Algorithm, if s is the point of intersection of the hyperbola branch H_v with a line through q that forms the angle $90^\circ + \alpha$ with the positive x-axis and if χ'' is the x-coordinate of the point of intersection of the x-axis with the line through s that forms angle $90^\circ - \alpha$ with the positive x-axis, then, for each $\chi \in [current\_\chi, \chi'']$, v belongs to $loc\_max(\chi)$ (see also Lemma 6(ii) and (iii)). The correctness of the algorithm follows from the previous observation, the correctness of the First Algorithm, the x-monotonicity of the upper envelope H, and from Lemmas 4, 7(ii), 8, and 9, respectively.

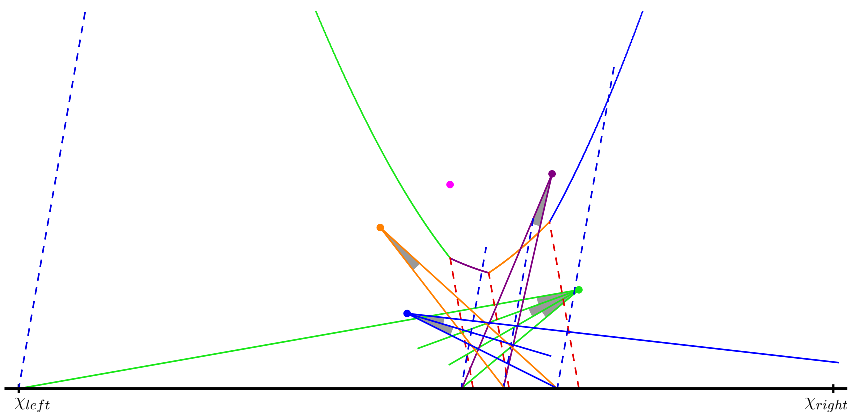

Figure 8 Illustration of the operation of the Second Algorithm on the set of five points shown in Figure 5 for $\alpha = 10^\circ$ to illuminate the range $[\chi_{left}, \chi_{right}]$ of the x-axis and the resulting floodlights.

4.1 Complexity of Second Algorithm

The upper envelope H of the hyperbola branches H_t for $t \in S$ can be efficiently computed by using the farthest-point Voronoi diagram $fVD(S)$ of the vertices of the convex hull of the set S, which coincides with the farthest point Voronoi diagram of the vertices of S (see Lemma 1(i)). The following lemma and the corollary give the relationship of the $fVD(S)$ with the arcs in the upper envelope H.

▶ **Lemma 10.** *Let H be the upper envelope of the hyperbola branches of all the vertices in the convex hull $CH(S)$ of S and let $fVD(S)$ be the farthest-point Voronoi diagram of the vertices of $CH(S)$. Moreover, let H_v be the hyperbola branch of a vertex v of the convex hull of S and let t be a point of H_v. Then, the point t belongs to H if and only if t belongs to the closure of the cell of v in $fVD(S)$.*

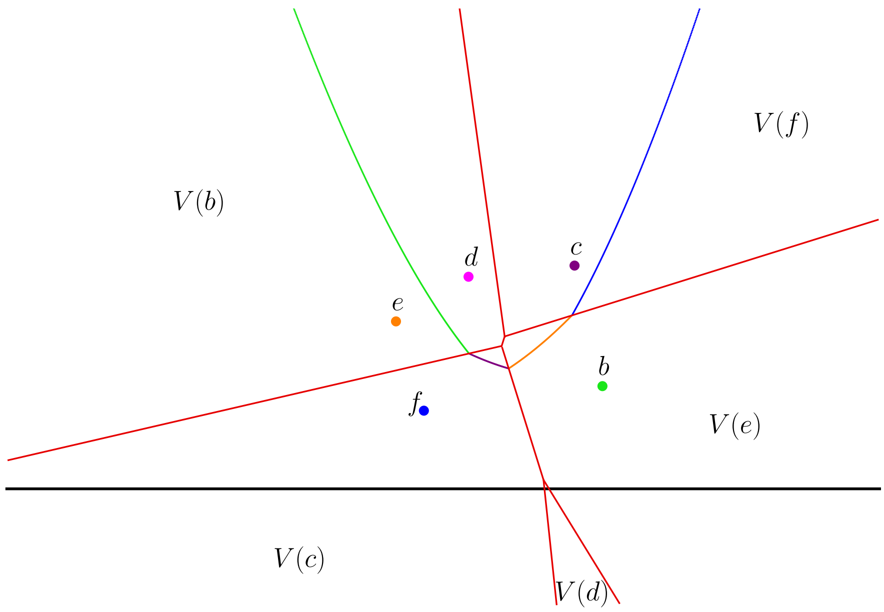

Figure 9 The upper envelope H of the five points shown in Figure 5 and their farthest-point Voronoi diagram (shown in red).

Lemma 10 implies the following corollary; see Figure 9.

▶ **Corollary 11.** *Let H, $fVD(S)$, v, and H_v be as in Lemma 10. Then, the following hold.*

(i) *The part of the hyperbola branch H_v of a vertex $v \in CH(S)$ that belongs to H is precisely the intersection of H_v with the cell of v in $fVD(S)$.*

(ii) *A point $t \in H$ is a vertex of H if and only if t either lies on an edge or is a vertex of $fVD(S)$.*

(iii) *The size (number of vertices or hyperbola arcs) of the upper envelope H of the hyperbola branches of all the vertices in the convex hull of the set S is $O(|CH(S)|)$.*

Since each arc of the upper envelope H produces at most one pair in the solution F in Step 2 of the Second Algorithm (see Figure 8), Corollary 11(iii) implies that:

▶ **Corollary 12.** *The size of the solution computed by the Second Algorithm is $O(n)$ where n is the total number of vertices of the location set S. Hence, the same holds for any solution to the Axis α-Illumination Problem as it is described in Section 1.*

Now we are ready to estimate the complexity of the Second Algorithm. Let n be the number of vertices of the set S. The computation of the convex hull of the vertices of S, which coincides with the convex hull of S, takes $O(n \log n)$ time [6]. The computation of the upper envelope of H can be done by computing first the farthest-point Voronoi diagram of the convex hull vertices of S, and then the vertices of H by taking advantage of Corollary 11(ii), the proof of Corollary 11(iii), and the appropriate hyperbola arcs of H in accordance with Corollary 11(i); the farthest point Voronoi diagram of $O(n)$ points can be computed in $O(n \log n)$ time [16] and has $O(n)$ size (see Lemma 1(ii)), while the remaining work can be done in $O(n)$ time. Finally, computing v_{first} and v_{last} takes $O(n)$ time and the initialization of F takes constant time. In total, Step 1 can be completed in $O(n \log n)$ time.

Each iteration of the `while` loop in Step 2 of the algorithm requires constant time for everything but determining the arc A intersected by the line L. Finding the arc A can be done in $O(\log n)$ time by using binary search on the x-monotone upper envelope H; in fact, all the arcs A needed in the different iterations of the `while` loop can be found in $O(n)$ total time by walking along H from left to right as needed by the algorithm. Because each iteration of the `while` loop involves a different arc in H and the total number of arcs is $O(n)$ (Corollary 11(iii)), the total number of iterations is $O(n)$. In addition to the `while` loop, Step 2 contains operations that require constant total time; thus, Step 2 can be completed in $O(n \log n)$ time. Step 3 takes $O(n)$ time because we can efficiently merge the appropriate pairs in the set F computed after Step 2 by processing them in the order they are produced.

Overall, the algorithm takes $O(n \log n)$ time. The space for computing and storing the upper envelope H of the hyperbola branches, the farthest-point Voronoi diagram (Lemma 1(ii)), and the solution F (Corollary 12) is $O(n)$. Thus, we conclude with the following theorem.

▶ **Theorem 13.** *The Second Algorithm correctly computes a solution to the Axis α-Illumination Problem and requires $O(n \log n)$ time and $O(n)$ space, where n is the number of vertices of the given location set.*

5 Concluding Remarks

We proved that the Axis α-Illumination Problem admits an $O(n \log n)$-time and $O(n)$-space algorithm where n is the number of vertices of the given location set. The obvious open question is whether there exists a matching lower bound for that problem or if not, to find a faster algorithm.

A natural extension of our Axis α-Illumination Problem is the case of a set S of regions (with piece-wise linear boundary) lying in a polygon and we want to illuminate the boundary of that polygon. This problem may be thought of as a variant of the Art Gallery polygon where the purpose is to guard only the boundary of the input polygon [12]. We believe that the Second Algorithm can be extended to the case where S lies in a circle or in a convex polygon and we want to illuminate the boundary of that circle/polygon. However, in the case of a simple polygon, the problem is NP-hard [13] and APX-hard [10].

References

1 Ahmed Abdelkader, Ahmed Saeed, Khaled A. Harras, and Amr Mohamed. The inapproximability of illuminating polygons by α-floodlights. In *Proceedings of the 27th Canadian Conference on Computational Geometry, CCCG 2015, Kingston, Ontario, Canada, August 10-12, 2015*. Queen's University, Ontario, Canada, 2015. URL: `http://research.cs.queensu.ca/cccg2015/CCCG15-papers/25.pdf`.

2 Sergey Bereg, José Miguel Díaz-Báñez, Marta Fort, Mario Alberto López, Pablo Pérez-Lantero, and Jorge Urrutia. Continuous surveillance of points by rotating floodlights. *Int. J. Comput. Geom. Appl.*, 24(3):183–196, 2014. `doi:10.1142/S0218195914600024`.

3 Prosenjit Bose, Leonidas J. Guibas, Anna Lubiw, Mark H. Overmars, Diane L. Souvaine, and Jorge Urrutia. The floodlight problem. *Int. J. Comput. Geom. Appl.*, 7(1/2):153–163, 1997. doi:10.1142/S0218195997000090.

4 Jurek Czyzowicz, Stefan Dobrev, Benson L. Joeris, Evangelos Kranakis, Danny Krizanc, Ján Manuch, Oscar Morales-Ponce, Jaroslav Opatrny, Ladislav Stacho, and Jorge Urrutia. Monitoring the plane with rotating radars. *Graphs Comb.*, 31(2):393–405, 2015. doi:10.1007/s00373-015-1543-4.

5 Jurek Czyzowicz, Eduardo Rivera-Campo, and Jorge Urrutia. Optimal floodlight illumination of stages. In *Proceedings of the 5th Canadian Conference on Computational Geometry, Waterloo, Ontario, Canada, August 1993*, pages 393–398. University of Waterloo, 1993.

6 Mark de Berg, Otfried Cheong, Marc J. van Kreveld, and Mark H. Overmars. *Computational Geometry: Algorithms and Applications, 3rd Edition.* Springer, 2008. URL: https://www.worldcat.org/oclc/227584184.

7 Jana Dietel, Hans-Dietrich Hecker, and Andreas Spillner. A note on optimal floodlight illumination of stages. *Inf. Process. Lett.*, 105(4):121–123, 2008. doi:10.1016/j.ipl.2007.08.009.

8 Vladimir Estivill-Castro, Joseph O'Rourke, Jorge Urrutia, and Dianna Xu. Illumination of polygons with vertex lights. *Inf. Process. Lett.*, 56(1):9–13, 1995. doi:10.1016/0020-0190(95)00129-Z.

9 Vladimir Estivill-Castro and Jorge Urrutia. Two-floodlight illumination of convex polygons. In Selim G. Akl, Frank K. H. A. Dehne, Jörg-Rüdiger Sack, and Nicola Santoro, editors, *Algorithms and Data Structures, 4th International Workshop, WADS '95, Kingston, Ontario, Canada, August 16-18, 1995, Proceedings*, volume 955 of *Lecture Notes in Computer Science*, pages 62–73. Springer, 1995. doi:10.1007/3-540-60220-8_51.

10 Christodoulos Fragoudakis, Euripides Markou, and Stathis Zachos. Maximizing the guarded boundary of an art gallery is apx-complete. *Comput. Geom.*, 38(3):170–180, 2007. doi:10.1016/j.comgeo.2006.12.001.

11 Hiro Ito, Hideyuki Uehara, and Mitsuo Yokoyama. Np-completeness of stage illumination problems. In Jin Akiyama, Mikio Kano, and Masatsugu Urabe, editors, *Discrete and Computational Geometry, Japanese Conference, JCDCG'98, Tokyo, Japan, December 9-12, 1998, Revised Papers*, volume 1763 of *Lecture Notes in Computer Science*, pages 158–165. Springer, 1998. doi:10.1007/978-3-540-46515-7_12.

12 Aldo Laurentini. Guarding the walls of an art gallery. *Vis. Comput.*, 15(6):265–278, 1999. doi:10.1007/s003710050177.

13 Joseph O'Rourke. *Art Gallery Theorems and Algorithms.* Oxford University Press, Inc., USA, 1987.

14 Joseph O'Rourke. Visibility. In Jacob E. Goodman and Joseph O'Rourke, editors, *Handbook of Discrete and Computational Geometry, Second Edition*, pages 643–663. Chapman and Hall/CRC, 2004. doi:10.1201/9781420035315.ch28.

15 T.C. Shermer. Recent results in art galleries (geometry). *Proceedings of the IEEE*, 80(9):1384–1399, 1992. doi:10.1109/5.163407.

16 Sven Skyum. A simple algorithm for computing the smallest enclosing circle. *Inf. Process. Lett.*, 37(3):121–125, 1991. doi:10.1016/0020-0190(91)90030-L.

17 William L. Steiger and Ileana Streinu. Illumination by floodlights. *Comput. Geom.*, 10(1):57–70, 1998. doi:10.1016/S0925-7721(97)00027-8.

18 Kazuo Sugihara, Ichiro Suzuki, and Masafumi Yamashita. The searchlight scheduling problem. *SIAM J. Comput.*, 19(6):1024–1040, 1990. doi:10.1137/0219070.

19 Dan Tao and Tin-Yu Wu. A survey on barrier coverage problem in directional sensor networks. *IEEE Sensors Journal*, 15(2):876–885, 2015. doi:10.1109/JSEN.2014.2310180.

20 Jorge Urrutia. Art gallery and illumination problems. In Jörg-Rüdiger Sack and Jorge Urrutia, editors, *Handbook of Computational Geometry*, pages 973–1027. North Holland / Elsevier, 2000. doi:10.1016/b978-044482537-7/50023-1.

21 Izu Vaisman. *Analytical Geometry.* World Scientific, 1997. doi:10.1142/3494.