On the Inverse Beacon Attraction Region of a Point

Bahram Kouhestani David Rappaport Kai Salomaa*

Abstract

Motivated by routing in sensor networks, Biro et al. [2] introduced the notion of beacon attraction and inverse attraction as a new variant of visibility in a simple polygon. A beacon b is a point inside a polygon P that can induce an attraction that moves a target point p greedily towards it in a trajectory that always reduces distance from p to b. The trajectory of p may require sliding p along the boundary of an obstacle. The attraction region of b is the set of all points that eventually reach b. The inverse attraction region of p is the set of points that can attract p. We present algorithms to efficiently compute the inverse attraction region of a point for simple, monotone, and terrain polygons with respective time complexities $O(n^3)$, $O(n \log n)$ and $O(n)$.

1 Introduction

Biro et al. [2] introduced a novel variation of the art gallery problem motivated by geographical greedy routing in sensor networks. A guard is a fixed point, called a beacon, that induces a force of attraction within the environment. The attraction of a beacon moves objects (represented by points) greedily towards the beacon. A point is attracted (covered) by a beacon if it eventually reaches the beacon. It is a common practice in sensor networks that message sending is performed by greedy routing where a node sends or passes the message to its neighbour that is closest to the destination. Depending on the geometry of the network and the location of the sender and receiver, greedy routing may fail. This introduces the interesting problem to determine whether messages can be exchanged between sender and receiver using greedy routing.

Biro et al. [2] studied the combinatorics of guarding a polygon with beacons and showed that $\left\lfloor \frac{n}{2} \right\rfloor - 1$ beacons are sometimes necessary and always sufficient to route between any pair of points in a simple polygon. They also proved that it is NP-hard to find a minimum cardinality set of beacons to cover a simple polygon. In 2013, Biro et al. [3] presented a polynomial time algorithm for routing between two fixed points using a discrete set of candidate beacons in a simple polygon and a 2-approximation algorithm where the beacons are placed with no restrictions. For polygons with holes, Biro et al. [4] showed that $\left\lfloor \frac{n}{2} \right\rfloor - h - 1$ beacons are sometimes necessary and $\left\lfloor \frac{n}{2} \right\rfloor + h - 1$ beacons are always sufficient to guard a polygon with h holes. For other results on beacons see [1].

In this paper we present algorithms to compute the inverse attraction region of a point inside an n-gon. We show that the inverse attraction region of a point can be computed in $O(n^3)$ time in a simple polygon. For monotone polygons we present a simple $O(n \log n)$ time algorithm to compute the inverse attraction region, and for terrain polygons we can further reduce the complexity to $O(n)$ time.

2 Preliminaries

Let P be simple polygon in the plane with the vertices $v_1, v_2, ..., v_n$ in counter-clockwise order. P is *monotone* with respect to the line L if every line orthogonal to L intersects P in at most one connected component. Throughout this paper, without loss of generality, we assume that L is the x-axis. Let u and v be the first and last vertices of the monotone polygon M in lexicographic order. The upper (lower) chain of M is the ordered set of edges from u to v in clockwise (counter-clockwise) order. We define a *terrain polygon*[1] as a monotone polygon with one of its chains consisting of a single line segment.

Let p and q be two points inside P. The Euclidean shortest path (geodesic path) between p and q, $SP(p, q)$ is a path inside P that connects p and q and among all such paths it has the smallest length. The union of Euclidean shortest paths from p to all vertices of P is called the shortest path tree of p and is denoted by $SPT(p)$. Guibas et al. presented a linear time algorithm to compute $SPT(p)$ [6]. It is worth mentioning that $SP(p, q)$ turns only at reflex vertices of P and the angle facing the exterior of P at a turn is convex (the outward convex property of the shortest path). The parent of a node $u \neq p$ in $SPT(p)$ is the last reflex vertex on $SP(p, u)$ which is not u. For proofs and details on shortest paths, see [7, Ch. 3].

*School of Computing, Queen's University, Kingston, Ontario, Canada. {kouhesta,daver,ksalomaa}@cs.queensu.ca

[1]A terrain polygon is sometimes called a "monotone mountain".

A *beacon* is a stationary point inside a simple polygon P that can induce a force of attraction within P. When beacon b is activated, points in P move greedily towards b and monotonically decrease their Euclidean distance to b. Furthermore, points are allowed to slide on the boundary of the environment in order to get closer to b, thus, the movement of a point alternates between moving straight towards b and sliding on the boundary of P (Fig. 1). Let e be an edge of P and let L be the supporting line of e. Let h be the orthogonal projection of b on L (Fig. 2a). As h is the point with the shortest distance to b among all points on L, sliding on e is always towards h. If h is located on e a point sliding on e will reach h and remain on h. Otherwise, it slides all the way to an endpoint of e. Then the point will move straight towards b if that is possible. Otherwise, depending on the location of the orthogonal projection of b on the supporting line of the adjacent edge, the point either slides on the new edge or remains stationary on the endpoint (Fig. 2).

Eventually a moving point either reaches b or becomes stuck on a boundary point of P. The path from the original position of a point p to its final position is called the *attraction trajectory* of p. A point in P is *attracted* by b if its Euclidean distance to b is eventually decreased to 0. The *attraction region* of a beacon b is the set of all points in P that b can attract and can be computed in linear time [1]. In the case that the point does not reach b, its final location is called a *dead point*. The *dead region* relative to a dead point d is the set of all points that end up on d. The boundary between the attraction region and a dead region or two dead regions is called a *split edge*. We denote a split edge that separates the attraction region of the beacon from a dead region as a *separation edge*. In contrast to conventional visibility, beacon attraction is not symmetric. For example in Fig. 1 a beacon located on p cannot attract a point on b. The *inverse attraction region* of a point q is defined as the set of beacon locations in P that attract q.

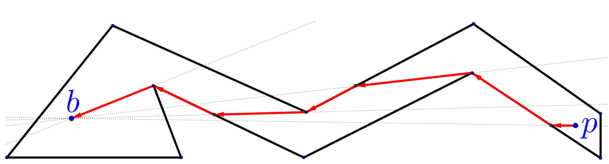

Figure 1: The movement of a point alternates between moving straight towards the beacon and sliding on the boundary.

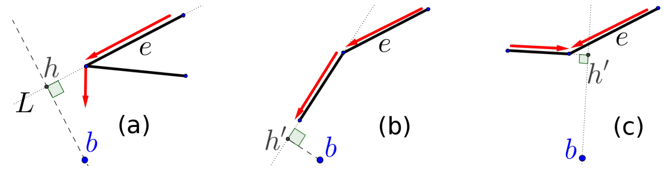

Figure 2: Three cases when a point slides to an endpoint of e. (a) It moves straight towards b. (b) It slides on the adjacent edge. (c) It get stuck on the endpoint. Here h' is the orthogonal projection of b on the supporting line of the adjacent edge of e.

Let r be a reflex vertex incident to edges e_1 and e_2. Let H_1 and H_2 be half-planes perpendicular to e_1 and e_2 emanating from r which include the outside of P in a small neighbourhood of r. The *dead wedge* of r is defined as the intersection of H_1 and H_2 (Fig 3). Let b be a beacon inside the dead wedge of r and to the left of r. Consider h, the orthogonal projection of b on the supporting line of e_2. Note that a point on e_2 close to r will slide away from r. Let Γ be the ray from r and in the direction of $\overrightarrow{br}$ and let s be the line segment between r and the first intersection of Γ with the boundary of P. The attraction of b to a point just to the right of s moves the point to e_2 and slides it towards h, while a point just to the left of s avoids e_2 and passes r. In other words the final destination of those two points will be different and therefore s is a split edge of b and we have the following lemma.

Lemma 1 *A reflex vertex r introduces a split edge for the beacon b if and only if b is inside the dead wedge of r.*

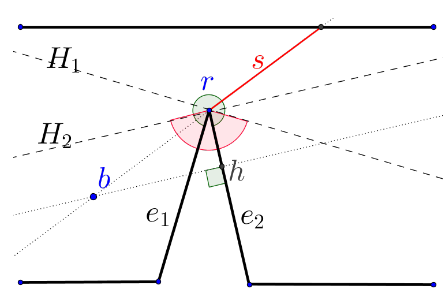

Figure 3: The dead wedge of a reflex vertex r is the intersection of half planes H_1 and H_2 designated by the red arc.

Next we address the problem of computing the inverse attraction region, that is:
Given a simple polygon P and a point q inside P, find the set of all beacon locations in P that attract q.

3 Inverse attraction region in simple polygons

Biro presented an algorithm for computing the inverse attraction region of a point in a simple polygon [1]. Unfortunately his $O(n^2)$ time and space algorithm has a flaw. The algorithm begins by constructing an arrangement A_P of lines to partition P with the idea that for any two points inside a particular region, either both or none attract q.

The arrangement A_P contains three types of lines: 1) lines through edges of P, 2) lines through a reflex vertex and perpendicular to one of the edges incident to this reflex vertex (i.e lines supporting edges of the dead wedge of reflex vertices), and 3) lines through q and each reflex vertex of P.

As far as we know Biro's proof [1] of the following property for A_P, is correct.

Property 1: If b_1 and b_2 belong to the same region of A_P and the reflex vertex r is a split vertex relative to b_1 (i.e. r introduces a split edge for b_1) then r is also a split vertex relative to b_2 [1].

Biro used Property 1 to conclude that all points in a particular region behave the same with respect to q (all or none attract q). The example in Fig. 4 illustrates that property 1 is not sufficient to guarantee that points in the same region have the same attraction behaviour with respect to q. Consider the line L going through the reflex vertices r_1 and r_2 and let s and t be two points close to and on opposite sides of L. Even though r_2 introduces a split edge for both s and t, it is easy to see that s cannot attract q while t can. This example suggests that additional lines need to be added to the arrangement.

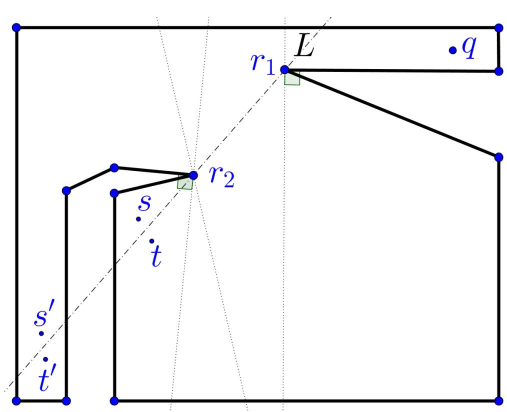

Figure 4: An example where the arrangement in [1] does not work. Point s cannot attract q while t can. Also observe that point s' can attract q while t' cannot.

The example in Fig. 4 implies that it is necessary to add some of the lines going through pairs of reflex vertices of P to the arrangement. As a polygon may have $O(n)$ reflex vertices, this adds an additional $O(n^2)$ lines with an arrangement of $O(n^4)$ regions.

We construct a new arrangement of $O(n^2)$ complexity which correctly groups together points in P.

The arrangement A_P uses three types of lines:
1) Lines through edges of $SPT(q)$.
2) Lines through the edges of the dead wedge of a reflex vertex of p.
3) Lines through edges of the polygon.

Note that lines of the third type are added to A_P to distinguish points that are inside or outside of P.

Lemma 2 *If b_1 and b_2 belong to the same region of A_P then either both or neither attract q.*

Proof. For the sake of contradiction and without loss of generality assume b_1 attracts q while b_2 does not attract q. Let r be the split vertex that separates q from the attraction region of b_2 (i.e. r introduces a split edge for b_2 that separates q from the attraction region of b_2). Without loss of generality let us assume that q is to the left of this split edge, s_2 (Fig. 5). As b_1 and b_2 are in the same region of A_p, b_1 is also in the dead wedge of r and r introduces a split edge for b_1. As b_1 attracts q, q lies to the right of this split edge s_1.

We have two cases:
1) Both s_1 and s_2 have an (upper) endpoint on a common edge e (Fig. 5a). In this case q lies in the triangle formed by s_1, s_2 and e. This triangle is contained in P, and therefore q sees r and the line segment connecting r and q is in $SPT(q)$. Therefore, the line $\overline{qr}$ forces b_1 and b_2 to be in two different regions of A_P, a contradiction.
2) Edges s_1 and s_2 have (upper) endpoints on different edges of P (Fig. 5b). Let the endpoint of s_1 lie on e and the endpoint of s_2 lie on e'. Notice that the left endpoint of e' is located between s_1 and s_2. Now consider the shortest path between q and r. If q and r see each other directly then the supporting line of the line segment $\overline{qr}$ belongs to A_P and similar to the previous case we have a contradiction. If q and r cannot see each other directly then there exists a reflex vertex r' between s_1 and s_2 such that the shortest path between q and r passes through r'. Now by the construction the line $\overline{rr'}$ is in A_P which forces b_1 and b_2 to be located in two different regions, a contradiction. □

Theorem 3 *The inverse attraction region of a point in a simple polygon can be computed in $O(n^3)$ time and $O(n^2)$ space.*

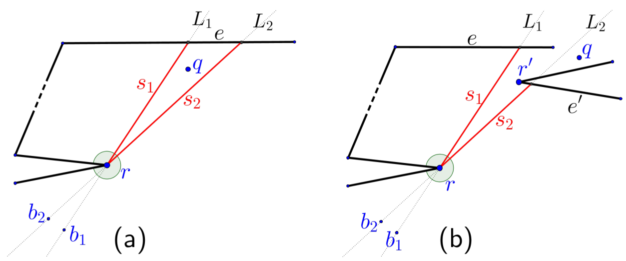

Figure 5: Split edges of b_1 and b_2.

Proof. There are $O(n)$ lines in the arrangement. Therefore the number of regions in the arrangement is $O(n^2)$ and for each region we can check whether a candidate point can attract q in linear time, resulting in the O(n^3) time complexity. □

4 Inverse attraction region in a monotone polygon

In the previous section we showed that lines passing through edges of SPT(q) and through edges of dead wedges form the boundaries between regions that attract q and those that don't. For the case of monotone polygons we show that a much smaller subset of these boundary edges suffice.

Let M be a monotone polygon and let q be a point in M. We begin by studying the effect of a single reflex vertex on the inverse attraction region of q. Let v be a reflex vertex of M with e_l and e_r the left and right adjacent edges of v, respectively. Let $q \in M$ be a point to the right of v. Our goal is to distinguish all beacon placements to the left of v that do not attract q because they are blocked by an edge incident to v. To do so, first we assume that there are no reflex vertices between q and v (i.e no reflex vertex exists simultaneously to the left of q and to the right of v) and find points to the left of v that cannot move q past a vertical line through v.

We show that a ray passing through v can be used to bound a subpolygon of M so that any beacon placed within that subpolygon cannot attract the point q. This ray can be defined in one of two ways yielding what we call a *blocking ray*. The two cases of blocking rays are described as follows:

Case 1 blocking ray: $q_1 \in M$ is a point to the right of the reflex vertex v and below the line L_1 orthogonal to e_r at v. Observe that L_1 passes through the left edge of the dead wedge of v. According to attraction properties (Fig. 2), a beacon in M below L_1 and to the left of v cannot attract q_1 past the vertical line through v, and therefore it does not attract q_1. Thus we can express the effect of v by a ray Γ_1 emanating from v extending to the left along L_1. No point below Γ_1 can attract q_1. We call Γ_1 the *blocking ray* of v relative to q_1.

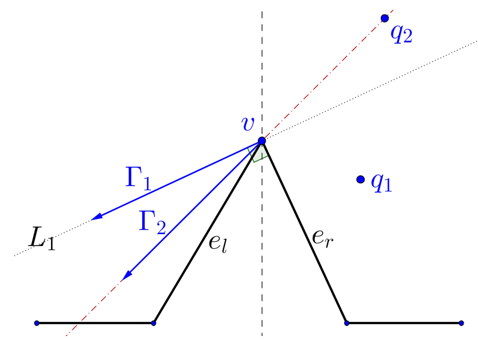

Figure 6: The effect of a single reflex vertex on the inverse attraction region of a point. Case 1 blocking ray: q_1 lies below L_1: beacons below Γ_1 cannot attract q_1. Case 2 blocking ray: q_2 lies above L_1 and and below L_2: beacons below Γ_2 cannot attract q_2.

Case 2 blocking ray: $q_2 \in M$ is a point to the right of v and above L_1. Let Γ_2 be the ray emanating from v extending to the left along the line $\overline{q_2 v}$. Note that Γ_2 is in the dead wedge of v. Consider a beacon b to the left of v. If b is to the right of Γ_2 then the attraction path of q_2 will intersect e_r and by considering the orthogonal projection of b on the supporting line of e_r, we see that b cannot pass q_2 over v. Now assume b is to the left of Γ_2. Here the line segment $\overline{q_2 b}$ will not intersect e_r and therefore b can move q past over v. Here Γ_2 is the blocking ray of v relative to q_2.

We define the *blocking region* of a reflex vertex v relative to q as points of M which are below the blocking ray of v relative to q. Informally, the blocking region of v is the set of beacon locations that cannot attract q due to v. Note that a point in the blocking region of v (in both cases) is in the dead wedge of v.

We can now present an algorithm to compute the inverse attraction region of a point in a monotone polygon.

Algorithm InverseAttractionRegion
Input. Monotone polygon M and a point $q \in M$.
Output. Inverse attraction region of q, that is, beacon locations in P that attract q.

```
1: Compute SPT(q), the shortest path tree from q to
   each vertex of M.
2: for each reflex vertex r that sees q do
3:     Discard points in the blocking region of r relative
       to q
4: end for
```

for each pair of consecutive reflex vertices v, v' in $SPT(q)$ ($v = \text{parent}(v')$) **do**
  Discard points in the blocking region of v' relative to v.
end for
return The remaining polygon

Theorem 4 *Algorithm InverseAttractionRegion correctly computes the inverse attraction region of an input point q in a monotone polygon.*

Proof. Suppose p is discarded by the algorithm due to the edge $s = vv'$, where $v = parent(v')$ in SPT(q). We claim that p cannot attract any point on s (see appendix). We show that p cannot attract q as well. We consider two cases:

1) v and v' lie on different chains of M. Here, s partitions M into two sub-polygons and p and q are in different sub-polygons. Let π be the attraction trajectory of q to p. As p and q are on different sides of s, π crosses s. Let x be the intersection of π and s. As p cannot attract x, we conclude that it cannot attract q.

2) v and v' are on the same monotone chains (Fig. 8). Let w be the first intersection point of the ray $\overrightarrow{v'v}$ with M to the right of v. Note that as the shortest path is outward convex, the parent of v in $SPT(q)$ lies in the sub-polygon to the right of the line segment $\overline{vw}$. Therefore, $\overline{vw}$ partitions M into two sub-polygons where p and q are in different sub-polygons. As p cannot attract v, we can show that it cannot attract any point on $\overline{vw}$ (see appendix). If p attracts q then the attraction trajectory must intersect $\overline{vw}$ which is a contradiction.

Now suppose p is a point that cannot attract q. Let t be the separation edge of the attraction region of p such that p and q are in different sides of t. Let v' be the reflex vertex that introduces t and M_1 be the sub-polygon that contains q (Fig. 7). Observe that $v = \text{parent}(v')$ in SPT(q) is in M_1 because the shortest path is outward convex. Therefore, p does not attract v and p lies in the blocking region of v' relative to v. With our construction when the pair $(v, v') \in SPT(q)$ is processed, p will be discarded. □

We use a result of Hershberger [5] that computes the upper envelope of a set S of n non-vertical line segments in $O(n \log n)$ time. The upper envelope of S is defined as the portion of the segments in S visible from $y = +\infty$. The lower envelope is defined symmetrically.

Lemma 5 *The time complexity of the Algorithm InverseAttractionRegion is $O(n \log n)$.*

Proof. In order to achieve an $O(n \log n)$ time complexity, we first collect all blocking rays and then discard

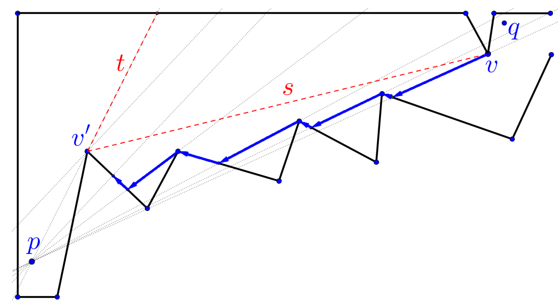

Figure 7: Attraction trajectory of v. Here, p cannot attract v.

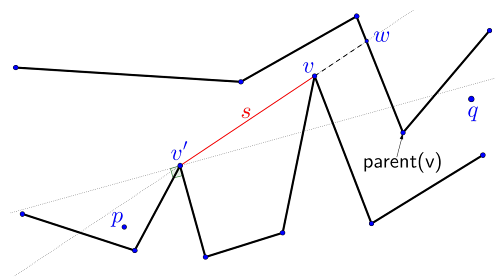

Figure 8: No point on $\overline{vw}$ can be attracted by p. Therefore p cannot attract q.

points in blocking regions. Let B be an axis aligned bounding box of M. By intersecting the blocking rays with B (and adding the top and bottom edges of B) we have a collection of blocking line segments. If the blocking line segment originated from a reflex vertex of the lower (upper) chain, then we need to discard points of M that are vertically below (above) this line segment. Using Hershberger's algorithm [5], we construct the upper (lower) envelope of blocking line segments of vertices of the lower (upper) chain in $O(n \log n)$ time and obtain two monotone polygons. The intersection of these two polygons with M is the set of points below all upper chain blocking rays and above all lower chain blocking rays. As the intersection of monotone polygons can be computed in linear time, the total complexity is $O(n \log n)$. □

5 Inverse attraction in a terrain polygon

Let M be a terrain polygon and let L be a vertical line through q. L partitions M into two terrain polygons. We consider each of these polygons separately and discard points that cannot attract q in each polygon. Here we explain how this is done for M_1, the polygon to the left of L. Let R_1 be all rays of R that extend from left to right. We present a linear time algorithm to discard points below the rays in R_1. The algorithm starts by traversing M_1 from right to left. Events are reflex vertices with a blocking ray that extends to the left. The algorithm preserves the invariant that at each event point the computed polygon to the right is the set of points in M_1 vertically above all current blocking rays $\Gamma_1, \Gamma_2, ..., \Gamma_i$. Furthermore, the algorithm stores and updates a convex set C which is the upper envelope of current rays intersected by a bounding box of the polygon.

Algorithm DiscardingBelowRays
Input. A terrain polygon M_1. A set $R = \Gamma_1, \Gamma_2, ..., \Gamma_m$ of blocking rays all extending to the left.
Output. A polygon P obtained by discarding points in M_1 vertically below the rays in R.

1: Order R such that Γ_i is the blocking ray of the reflex vertex r_i and r_i is to the left of r_{i+1} for all $i = 1, 2, ..., m-1$.
2: Let C be an axis aligned bounding box of M_1.
3: Let V_i be the vertical line through r_i and H_i be the half-plane to the left of V_i.
4: Let polygon P be the subset of M_1 between V_1 and a vertical line through q.
5: **for** $i = 1$ to m **do**
6: $\quad C = C \cap H_i$
7: $\quad$ **if** r_i is in C **then**
8: $\quad\quad$ Intersect C with the half-plane above the supporting line of Γ_i by traversing the lower edges of C and finding the first edge of C that intersects Γ_i.
9: $\quad\quad$ Add to P all points of M_1 between V_i and V_{i+1} that are also in C
10: $\quad$ **end if**
11: **end for**
12: **return** P

The algorithm computes the upper envelope of rays $\Gamma_1, \Gamma_2, ..., \Gamma_i$ between V_i and V_{i+1} and intersects the result with the portion of M_1 between V_i and V_{i+1} (Fig. 9). Therefore, the output are points of M above all blocking rays. Before we analyze the time complexity of the algorithm, we show that it is safe to ignore rays of reflex vertices that start below some current blocking regions (step 6).

Lemma 6 *If $r_i \notin C$ then Γ_i does not contribute to P.*

Proof. See appendix. □

Lemma 7 *Algorithm DiscardingBelowRays runs in $O(n)$ time.*

Proof. We use a sequential search in both step 7 intersecting $C \cap H_i$ and in step 9 intersecting the upper envelope of Γ_i with C. In each case once we step over an edge we eliminate it forever. Thus the overall complexity of the algorithm is $O(n)$. □

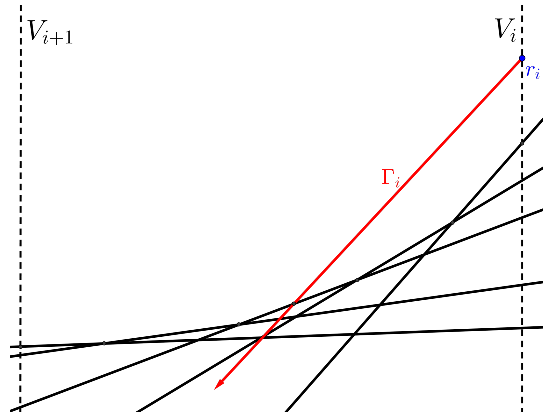

Figure 9: Discarding points below rays.

6 Conclusion

In this paper, we presented algorithms to efficiently compute the inverse attraction region of a point for simple, monotone, and terrain polygons. Currently we are developing a more efficient algorithm for simple polygons using the ideas of chapter 4. We believe that we can design an $O(n \log n)$ time algorithm which can be shown to be optimal.

References

[1] M. Biro. Beacon-based Routing and Guarding. PhD Dissertation, Stony Brook University, 2013.

[2] M. Biro and J. Gao, J. Iwerks, I. Kostitsyna and J. S. B. Mitchell. Beacon-based routing and coverage. Proceedings of the 21st Fall Workshop on Computational Geometry, 2011.

[3] M. Biro, J. Iwerks, I. Kostitsyna and J. S. B. Mitchell. Beacon-Based Algorithms for Geometric Routing. Proceedings of the 13th International Symposium on Algorithms and Data Structures, WADS, 2013.

[4] M. Biro, J. Gao, J. Iwerks, I. Kostitsyna and J. S. B. Mitchell. Combinatorics of beacon routing and coverage. Proceedings of the 25th Canadian Conference on Computational Geometry, 2013.

[5] J. Hershberger. Finding the upper envelope of n line segments in $O(n \log n)$ time. Information Processing Letters, 33(4):169-174, 1989.

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

[7] S. Ghosh. Visibility Algorithms in the Plane. Cambridge University Press, 2007.

Appendix

Although our goal is to compute the inverse attraction region of a fixed point, it is useful to compare the blocking regions of two points relative to a particular reflex vertex.

Lemma 8 *Let v be a reflex vertex of M. Let q and q' be two points of M such that q' is on the open line segment $\overline{qv}$. If there are no reflex vertices between v and q, then the blocking regions of q and q' (relative to v) are equal.*

Proof. Consider the cases in Fig. 6. It is easy to verify that when q' lies on the (open) line segment $\overline{qv}$, the blocking rays of q and q' are the same. Therefore, their blocking regions are equal. □

Lemma 9 *Let q be a point close to the left edge of v. Consider the clockwise rotation of q around v to a vertical position. During the rotation, the blocking region of v relative to q never increases.*

Proof. During the rotation, as long as q is below L_1 (see Fig. 6), the blocking region of q remains the same. While q is rotated from L_1 to a vertical position, the blocking ray of v relative to q will rotate clockwise from L_1 to a vertical downward ray. During this time the blocking region of q (i.e. points in M below the blocking ray) monotonically gets smaller until it is empty. Therefore, during the rotation the blocking region of q is non increasing. □

Next we consider the effect of two reflex vertices on the inverse attraction region of a point. Let v and v' be the only two reflex vertices of M. If a point $q \in M$ is located between v and v' then any attraction trajectory of q can at most have one of v and v' on its path and therefore the effect of v and v' can be considered separately. Therefore we focus on the inverse attraction region of the point q which lies to the right of both reflex vertices. Without loss of generality assume v is on the lower chain and v' is on the upper chain of M.

Case 1) If q is visible to both v and v', we claim that any attraction trajectory of q can at most pass through one of these reflex vertices. The attraction trajectory of q to a beacon b passes through v only if b is below the ray $\overrightarrow{qv}$ and passes through v' only if b is above the ray $\overrightarrow{qv'}$ (Fig. 10). As q sees both v and v' there does not exist a beacon both below $\overrightarrow{qv}$ and above $\overrightarrow{qv'}$. Therefore at most one reflex vertex can affect the attraction trajectory and in the computation of the inverse attraction v and v' are considered separately. We conclude that a point inside the blocking regions of v or v' cannot attract q.

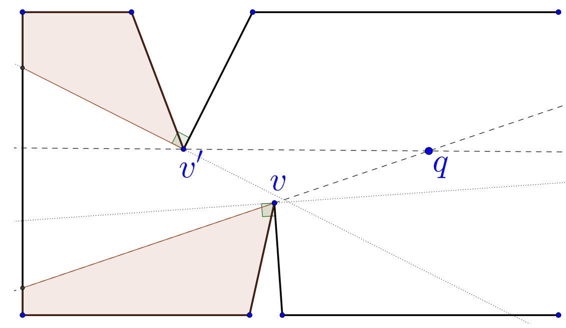

Figure 10: If q sees both v and v', no attraction trajectory of q can intersect both v and v' and in the computation of the inverse attraction region, v and v' are considered separately. Here points that cannot attract q are shaded.

Case 2) Otherwise, without loss of generality assume that q can see v but not v' (Fig. 11). We classify the points to the left of v' into two groups: *i*) points above the ray $\overrightarrow{qv}$ and *ii*) points below $\overrightarrow{qv}$. Let p be a point in group *i*. Consider π the attraction trajectory of q in the attraction of p. As p is located above $\overrightarrow{qv}$, π does not intersect the adjacent edges of v. We conclude that p can attract q if and only if p is not in the blocking region of v' (relative to q). Now assume that p is a point in group *ii*. In this case π will intersect v or the right edge of v. Therefore, p attracts q if and only if p can move q from its initial position to v (i.e. p is above the blocking ray of v relative to q) and p can attract v (i.e p is below the blocking ray of v' relative to v).

Next we show how to combine the two groups of case 2.

Lemma 10 *If q sees v but not v' then points in the blocking region of v relative to q and points in the blocking region of v' relative to v are the only points that cannot attract q.*

Proof. It is obvious that a point in the blocking region of v relative to q does not attract q, because it cannot move q past over v. So we only need to argue about points to the left of v'. Let p be a point in group *ii* (i.e. p is a point to left of v' and below $\overrightarrow{qv}$). By the previous argument p attracts q if and only if p can move q from its initial position to v and p can attract v. Therefore, p cannot lie in the blocking region of v (relative to q) and it cannot lie in the blocking region of v' relative to v and so the lemma follows.

Now let p be a point in group i (i.e. p is to the left of v' and above $\overrightarrow{pv}$). Note that as q does not see v', p also lies above the line $\overline{vv'}$ (see Fig. 11). Recall our case analysis in Fig. 6. If the relative position of v with respect to v' lies in case 1 (which is the case in Fig. 11), then the blocking region of v' relative to v is all points in the left side of the dead wedge of v'. The attraction trajectory of q in the attraction of a point in group i intersects the right edge of v'. Therefore a point in group i can attract q if it is not located on the left side of the dead wedge of v'. This is precisely the blocking region of v' relative to v.

Now assume that the relative position of v to v' lies in case 2 of Fig. 6. Recall that the blocking ray of v' relative to v is the ray from v' in the direction of the vector $\overrightarrow{vv'}$. As points above the $\overline{qv}$ are also above $\overline{vv'}$, all points of group i reside in the blocking region of v' relative to v and lemma follows. □

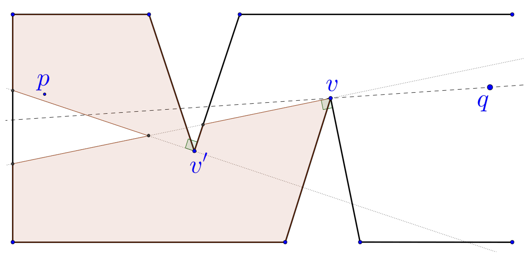

Figure 11: Points that cannot attract q are shaded.

Theorem 4 *The algorithm InverseAttractionRegion correctly computes the inverse attraction region of a given point in a monotone polygon.*

Proof. We use proof by contradiction. First that assume p is a point that can attract q and is discarded by the algorithm. Without loss of generality we assume p is to the left of q. If p is discarded in step 3 of the algorithm then let v be the rightmost reflex vertex responsible for discarding p. Note that q and v see each other, and p is in the blocking region of v therefore it is also in the dead wedge of v. As p is also to the left of v, p cannot attract any points on the right adjacent edge of v. Since p attracts q, the attraction trajectory of q to p must pass above v. Here in order for q to pass above v, there must exist an edge e between v and q such that q slides on e and moves above the line $\overline{pv}$. This implies that e blocks the visibility of v and q, which is a contradiction.

Assume p is discarded in step 6 due to s, where s is the directed open edge of SPT(q) from v to v'. Note that due to the monotonicity of M both v and v' are to the right of p and to the left of q. Consider π_{pv} the attraction trajectory of v to p (Fig. 7). As p is discarded when the pair (vv') is processed, in the absence of other reflex vertices p cannot attract v. Since v and v' are visible, no attraction trajectory (towards p) can slide through s. By lemma 8 the blocking region of all points on s are equal and by lemma 9 no points below s can have a blocking region smaller than the blocking region of v. Therefore (even in the presence of other reflex vertices) no points on π_{pv} can be attracted by p and thus p does not attract v. Now we show that p cannot attract q as well. We consider two cases:

1) v and v' lie on different chains of M. Here, s partitions M into two sub-polygons and p and q are in different sub-polygons. Note that by lemma 8 the blocking region of v relative to any point on s is precisely the blocking region of v relative to v'. This implies that p cannot attract any point on s. Let π be the attraction trajectory of q to p. As p and q are on different sides of s, π crosses s. Let x be the intersection of π and s. As p cannot attract x, we conclude that it cannot attract q.

2) v and v' are on the same monotone chains. Let w be the first intersection point of the ray $\overrightarrow{vv'}$ with M to the right of v (Fig. 8). Note that as the shortest path is outward convex, the parent of v in $SPT(q)$ lies in the sub-polygon to the right of the line segment $\overline{vw}$. Therefore, $\overline{vw}$ partitions M into two sub-polygons where p and q are in different sub-polygons. By lemma 8 the relative blocking region of v' relative to any point on $\overline{vw}$ is exactly the blocking region of v' relative to v. As p cannot attract v, it cannot attract any point on $\overline{vw}$. If p attracts q then the attraction trajectory must intersect uw which is a contradiction.

Now suppose p is a point that cannot attract q and is not discarded by the algorithm. Let t be the separation edge of the attraction region of p such that p and q are in different sides of t. Let v' be the reflex vertex that introduces t and M_1 be the sub-polygon that contains q (Fig. 7). Observe that $v = \text{parent}(v')$ in SPT(q) is in M_1 because the shortest path is outward convex. Therefore, p does not attract v and p lies in the blocking region of v' relative to v. With our construction when the pair $(v, v') \in SPT(q)$ is processed, p will be discarded. □

Lemma 6 *If $r_i \notin C$ then Γ_i does not contribute to P.*

Proof. Let Γ_i be the blocking ray of r_i and $r_i \notin C$. Let Γ_j $(j < i)$ be the leftmost ray above r_i. Consider the parent of r_i in $SPT(q)$. If r_j is the parent of r_i then the blocking ray of r_i relative to r_j will be on or under the ray $r_j r_i$, therefore all points in the blocking region of r_i are also in the blocking region of r_j. Now assume $w \neq r_j$ is the parent of r_i and therefore w lies above the ray $r_i r_j$. Consider the blocking ray of r_i relative to w. It lies on or below the line $r_i w$ and so below the line $r_i r_j$. Therefore in both cases the blocking region of r_i can be ignored. □