

Algorithms for computing diffuse reflection paths in polygons¹

Subir Kumar Ghosh², Partha Pratim Goswami³, Anil Maheshwari⁴,
Subhas Chandra Nandy⁵, Sudebkumar Prasant Pal⁶, Swami Sarvattomananda⁷

Abstract

Let s be a point source of light inside a polygon P of n vertices. A polygonal path from s to some point t inside P is called a *diffuse reflection path* if the turning points of the path lie on polygonal edges of P . We present three different algorithms for computing diffuse reflection paths from s to t inside P . For constructing such a path, the first algorithm uses a greedy method, the second algorithm uses a transformation of a minimum link path, and the third algorithm uses the edge-edge visibility graph of P . The first two algorithms are for polygons without holes, and they run in $O(n + k \log n)$ time, where k denotes the number of reflections in the path. The third algorithm is for both polygons with or without holes, and it runs in $O(n^2)$ time. The number of reflections in the path produced by this algorithm can be at most 3 times that of an optimal diffuse reflection path. The problem of computing a diffuse reflection path between two points inside a polygon has not been considered in the past.

1 Introduction

In the last four decades, the problems of *direct* visibility have been investigated extensively [8]. Two points inside a polygon P are called *visible* (directly) if the line segment joining them lie totally inside P . The region of P directly visible from a point light source s inside P is called the *visibility polygon* of P from s (see Figure 1). Several efficient algorithms exist for computing visibility polygons under different conditions [8]. Here, we consider a problem of computing indirect visibility in P of n vertices, that arises due to multiple reflections inside P .

Assume that all edges of P reflect light like mirrors. It can be seen that some points of P , that are not directly visible or illuminated from s , may still become visible due to one or more

¹A part of the work was done when authors visited Tata Institute of Fundamental Research in July, 2008.

²School of Technology & Computer Science, Tata Institute of Fundamental Research, Mumbai 400005, India (ghosh@tifr.res.in).

³Institute of Radiophysics and Electronics, University of Calcutta, Kolkata 700009, India (ppg.rpe@caluniv.ac.in).

⁴School of Computer Science, Carleton University, Ottawa, Canada K1S 5B6 (anil@scs.carleton.ca).

⁵Advanced Computing and Microelectronics Unit, Indian Statistical Institute, Kolkata 700108, India (nandysc@isical.ac.in).

⁶Department of Computer Science and Engineering, Indian Institute of Technology, Kharagpur 721302, India (spp@cse.iitkgp.ernet.in).

⁷School of Mathematical Sciences, Ramakrishna Mission Vivekananda University, Belur, India 711202, (shreesh@rkmvu.ac.in).

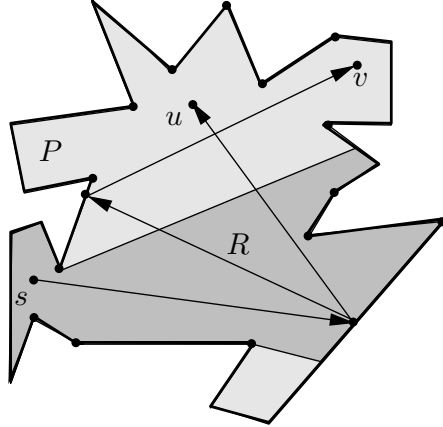


Figure 1: The region R is directly visible from s . A ray from s reaches u after one specular reflection. A ray from s reaches v after two diffuse reflections.

reflections on the edges of P (see Figure 1). As per the standard law of reflection, reflection of a light ray at a point is called *specular* if the angle of incidence is the same as the angle of reflection. There is another type of reflection of light called *diffuse* reflection, where a light ray incident at a point is reflected in all possible interior directions. We assume that the light ray incident at a vertex is absorbed and not reflected.

Visibility with multiple reflections arises naturally in three dimensional scenarios where pixels of a screen are rendered to generate a realistic image. This is achieved by tracing the path of light backwards from each pixel through multiple reflections until a source of light is reached [6]. Clearly, light sources that can be reached through a smaller number of reflections would contribute more intensely, thereby making paths reachable by the minimum number of reflections is more important in illumination modeling. It is therefore worthwhile computing paths through which light arrives from a light source by the minimum number of reflections. Our concern in this paper is the computation of such a path between two points s and t within a polygon P with the minimum number of diffuse reflections. Whether this is an NP-hard problem remains an open question.

Let us mention the previous results on visibility with multiple reflections. In [2], Aronov et al. investigated the region visible from a point source in a simple n -vertex polygon bounded by edges reflecting inwards, when at most one specular (or diffuse) reflection is permitted. For both specular as well as diffuse reflections, they established a tight $\Theta(n^2)$ worst-case geometric complexity bound and also designed an $O(n^2 \log^2 n)$ time algorithm for computing the region visible after at most one reflection. In [1], Aronov et al. addressed the more general problem where at most $k \geq 2$ specular reflections are permitted. They derived an $O(n^{2k})$ upper bound and an $\Omega((n/k)^{2k})$ worst-case lower bound on the geometric complexity of the region visible after at most a constant number k of *specular* reflections. They also designed an algorithm with

$O(n^{2k} \log n)$ running time, for $k > 1$.

In contrast to the result of [1], Prasad et al. showed in [17], that the upper bound on the number of edges and vertices of the region visible due to at most k reflections improves to $O(n^{2^{\lceil (k+1)/2 \rceil + 1}})$ when specular reflections are replaced by diffuse reflections. They also designed an $O(n^{2^{\lceil (k+1)/2 \rceil + 1} \log n})$ time algorithm for computing the visible region. They conjectured in [17] that the complexity of the region visible by at most k diffuse reflections is $\Theta(n^2)$. Note that this region contains blind spots or holes as shown in [16]. Recently, Aronov et al. [3] showed that the complexity of this region visible has an upper bound as low as $O(n^9)$. Bridging the gap between the $O(n^9)$ upper bound of Aronov et al. [3], and the $\Omega(n^2)$ lower bound in [17] remains an outstanding open problem.

In this paper, we present three different algorithms for computing diffuse reflection paths from s to t inside P . For constructing such a path, the first algorithm uses a greedy method, the second algorithm uses a transformation of a minimum link path, and the third algorithm uses the edge-edge visibility graph of P . The first two algorithms are for polygons without holes, and they run in $O(n + k \log n)$ time, where k denotes the number of reflections in the path. The third algorithm is for both polygons with or without holes, and it runs in $O(n^2)$ time. The number of turns in the path produced by this algorithm can be at most 3 times that of an optimal path, where an optimal path is a diffuse reflection path between s and t having the minimum number of turning points. In the next three sections, we present these algorithms. In Section 5, we conclude the paper with a few remarks.

2 Computing the greedy diffuse reflection path

In this section, we present an algorithm for computing a diffuse reflection path from s to t (denoted as $drp(s, t)$) inside a simple polygon P using greedy method. The algorithm runs in $O(n + k \log n)$ time, where k is the number of turning points in $drp(s, t)$.

Let $SP(u, v)$ denote the Euclidean shortest path between two points u and v inside P . Let $SP(s, t) = (u_0, u_1, \dots, u_j)$, where $s = u_0$ and $t = u_j$. Extend the first edge u_0u_1 from u_1 till it meets the boundary of P at some point w_1 (see Figure 2). If w_1 is directly visible from t , then the diffuse reflection path from s to t consists of two links sw_1 and w_1t . Otherwise, treating w_1 as s , compute the next link w_1w_2 of $drp(s, t)$ by extending the first edge of $SP(w_1, t)$ to the boundary of P . Repeat this process till w_k is computed such that w_k is directly visible from t . It can be seen that the greedy path $(sw_1, w_1w_2, \dots, w_{k-1}w_k, w_kt)$ is $drp(s, t)$. The correctness of the algorithm follows from the following lemma.

LEMMA 2.1 The greedy diffuse reflection path $(sw_1, w_1w_2, \dots, w_{k-1}w_k, w_kt)$ is a simple path.

Proof: Observe that (i) every link $w_{i-1}w_i$ of the greedy path intersects $SP(s, t)$ at some point z_i , (ii) $w_{i-1}w_i$ passes through a distinct vertex v_i of P , where w_iv_i is the first edge of $SP(w_i, t)$, and (iii) for every z_i , the next intersection point z_{i+1} lies on $SP(z_i, t)$ (see Figure 2). Hence, the greedy path is simple and the number of links in the path can be at most n .

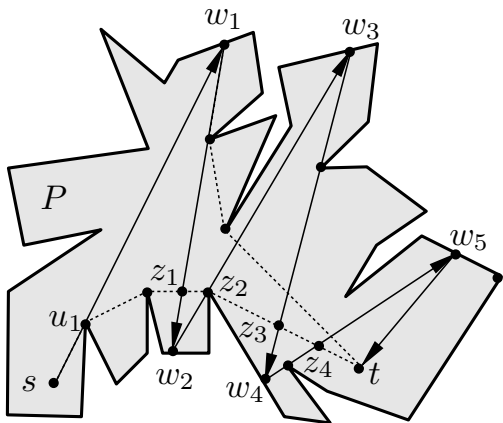


Figure 2: The greedy diffuse path from s to t inside P .

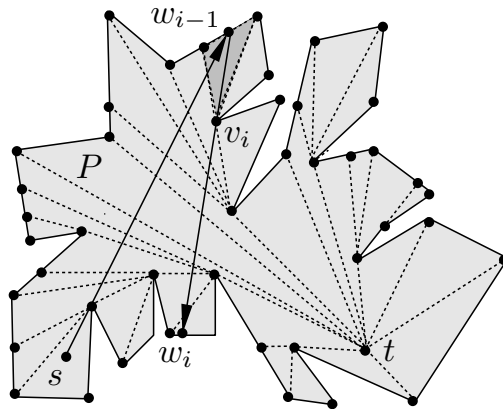


Figure 3: The next vertex v_i of w_{i-1} on $SP(w_{i-1}, t)$ belongs to the triangle containing w_{i-1} in the shortest path map.

Let us analyze the time complexity of the algorithm. In order to compute $w_{i-1}w_i$, the algorithm finds the next vertex v_i of w_{i-1} in $SP(w_{i-1}, t)$ and then extends $w_{i-1}v_i$ from v_i meeting the boundary of P at a point w_i . The vertex v_i can be located by computing $SP(w_{i-1}, t)$, which can be done in $O(n)$ time by the algorithm of Lee and Preparata [13]. Then the point w_i can also be located in $O(n)$ time by traversing through the triangles in the triangulation of P [4]. Since each link $w_{i-1}w_i$ can be computed in $O(n)$ time and there can be at most n links, the entire greedy path can be computed in $O(n^2)$ time.

Instead of computing shortest paths repeatedly for locating the next vertex, the algorithm computes the shortest path tree rooted at t (denoted as $SPT(t)$) in $O(n)$ time by the algorithm of Hershberger [11], and then constructs the shortest path map by extending the edges of $SPT(t)$. It can be seen that the next vertex v_i of w_{i-1} is a vertex of the triangle in the shortest path map which contains w_{i-1} (see Figure 3). Once the triangle containing w_{i-1} is located, the next vertex v_i is also located. Hence, the next vertex v_i can be located for each w_{i-1} in $O(\log n)$ time. Then the point w_i can be located by shooting a ray from w_{i-1} along $w_{i-1}v_i$, which takes $O(\log n)$ time after $O(n)$ time preprocessing [5]. Since the number of rays is bounded by the number of links in $drp(s, t)$, the greedy path can be computed in $O(n + k \log n)$ time.

Let us calculate the bound on the number of links of the greedy diffuse reflection path from s to t . Figure 4 shows that except the last two links $w_{k-1}w_k$ and $w_k t$, every link $w_i w_{i+1}$ of the greedy path passes through a vertex of P . Note that a diffuse reflection path by definition cannot take $w_i t$ as the last link if both w_i and t lie on the same polygonal edge. Since the greedy path does not pass through three vertices of P and the last two links do not pass through vertices of P , the number of links in the greedy path can be at most $n - 1$. On the other hand, the optimal path takes two links sz and zt to reach from s to t . Hence, the number of links in the greedy path can be at most $(n - 1)/2$ times that of an optimal path. We state

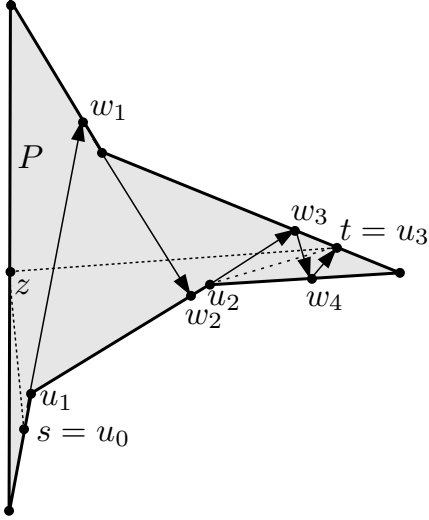


Figure 4: Links sz and zt form an optimal path, whereas the greedy diffuse reflection path turns on all edges of P except three edges.

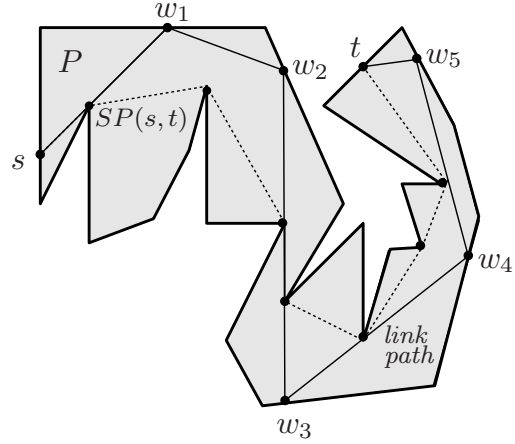


Figure 5: The minimum link path $(sw_1, w_1w_2, w_2w_3, w_3w_4, w_4w_5, w_5t)$ is also a diffuse reflection path as w_1, w_2, w_3, w_4 are boundary points.

the result in the following theorem.

THEOREM 2.2 The greedy diffuse reflection path from s to t can be computed in $O(n + k \log n)$ time, where k is the number of turning points in the path. The number of links in the path can be at most $(n - 1)/2$ times that of an optimal path.

3 Computing a diffuse reflection path using a minimum link path

In this section, we present an algorithm for transforming a minimum link path from s to t inside a simple polygon P into a diffuse reflection path $(sw_1, w_1w_2, \dots, w_{k-1}w_k, w_k t)$ in $O(n + k \log n)$ time. A minimum link path between two points s and t (denoted as $mlp(s, t)$) is a path inside P having the minimum number of segments or links. We have the following observations.

LEMMA 3.1 Between s and t , the number of reflections k in any diffuse reflection path in P cannot be smaller than the number of turns m in any minimum link path.

LEMMA 3.2 By Lemma 3.1, any diffuse reflection path between s and t can be at most k/m times the optimal diffuse reflection path.

The algorithm first constructs $mlp(s, t)$ in $O(n)$ time using the algorithm of Ghosh [7]. If all turning points of $mlp(s, t)$ lie on edges of P , then $mlp(s, t)$ is $drp(s, t)$ (see Figure 5). Moreover, $drp(s, t)$ is an optimal path as it has the minimum number of turns or reflections. We have the following lemma.

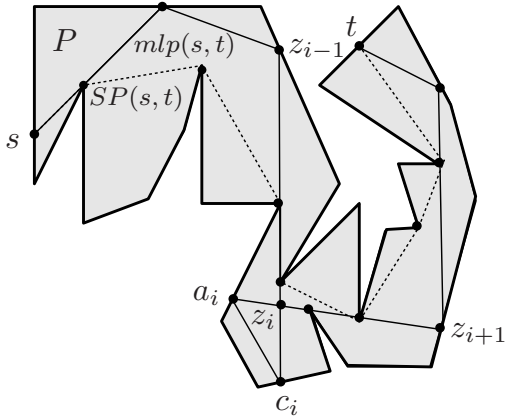


Figure 6: Two links $z_{i-1}z_i$ and $z_i z_{i+1}$ in $mlp(s, t)$ are replaced by three links $z_{i-1}c_i$, $c_i a_i$ and $a_i z_{i+1}$.

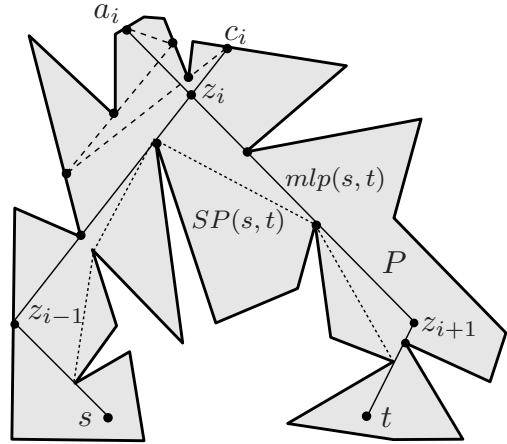


Figure 7: The points a_i and c_i are connected by the greedy diffuse reflection path.

LEMMA 3.3 If all turning points of a minimum link path lie on edges of P , then the path is an optimal diffuse reflection path.

Let $(sz_1, z_1z_2, \dots, z_{m-1}z_m, z_mt)$ be a minimum link path, where z_1, z_2, \dots, z_m are turning points. Consider the case when at least one turning point (say, z_i) is not lying on any edge of P (see Figure 6). Extend $z_i z_{i+1}$ from z_i to the boundary of P meeting it at a point a_i . Similarly, extend $z_i z_{i-1}$ from z_i to the boundary of P meeting it at a point c_i . If the segment $a_i c_i$ lies inside P , then $(sz_1, z_1z_2, \dots, z_{i-1}c_i, c_i a_i, a_i z_{i+1}, \dots, z_{m-1}z_m, z_mt)$ is $drp(s, t)$. Otherwise, a_i and c_i are connected by a greedy link path as stated in the previous section to construct a diffuse reflection path (see Figure 7). If the minimum link path has turning points that are not lying on edges of P , then the above method is used for each such turning point to transform a minimum link path into a diffuse reflection path. Note that the greedy diffuse reflection paths are computed into disjoint regions of P , and are bounded by the links of the minimum link path [7, 8].

Let us calculate the bound on the number of links in $drp(s, t)$. Let u_i and u_l be the next and previous vertices of s and t respectively on $SP(s, t)$ (see Figure 8). Let v_i and v_l be the next clockwise and counterclockwise vertices of u_i and u_l respectively on the boundary of P . Note that if s (or, t) is a vertex of P , then s (respectively, t) is denoted as v_i (respectively, v_l). It can be seen that all turning points of the greedy diffuse reflection path connecting c_i with a_i , for all i , lie on the clockwise boundary of P from v_i to v_l . As shown in Figure 8, all links in the greedy path from c_i to a_i , except the last two links, can pass through vertices of polygonal edges containing its turning points. These observations help us in calculating the bound for $drp(s, t)$ as follows.

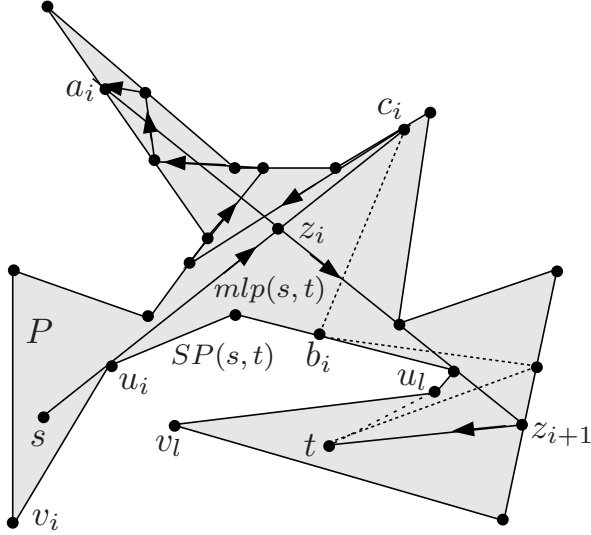


Figure 8: An optimal diffuse reflection path takes four links (or, three turns) to reach t from s .

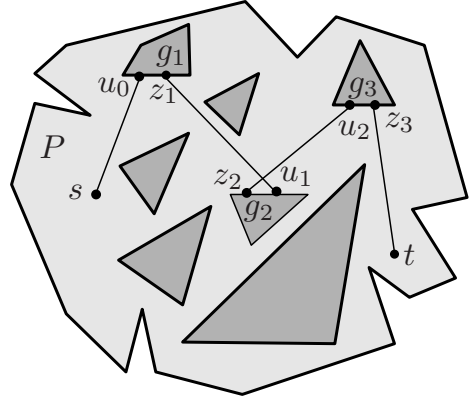


Figure 9: A sequence of links has been constructed connecting pairs of weakly visible edges of P .

Let m' be the number of turning points of $mlp(s, t)$ not lying on the boundary of P . Observe that the number of links in the greedy path can be at most $n - 2m' + 2m' - m - 1$ because (i) the greedy link path from c_i to a_i does not pass through one vertex of the edge containing c_i and another vertex of the edge containing a_i , (ii) the last two links for each of the m' greedy paths do not pass through vertices of P , and (iii) the number of vertices in the clockwise boundary of P from v_i to v_l (including v_i and v_l) must be at least $2 + m - 1$ as $m - 1$ links of $mlp(s, t)$ pass through distinct vertices of $SP(s, t)$ (see Figure 8). Therefore, $drp(s, t)$ can have at most $n - m - 1 + m = n - 1$ links. On the other hand, since the optimal $drp(s, t)$ must take at least one additional link $c_i b_i$ to cross $mlp(a_i, t)$ for every turning point of $mlp(s, t)$ not on the boundary of P , the optimal path must have at least $m + m'$ links. So, the number of links in $drp(s, t)$ can be at most $(n - 1)/(m + m')$ times that of an optimal path. We state the result in the following theorem.

THEOREM 3.4 A minimum link path between s and t can be transformed into a diffuse reflection path from s to t in $O(n + k \log n)$ time, and the number of links in the path can be at most $(n - 1)/(m + m')$ times that of an optimal path, where (i) k is the number of reflections in the diffuse reflection path, (ii) m is the number of links in the minimum link path, and (iii) m' is the number of turning points of the minimum link path not lying on the boundary of P .

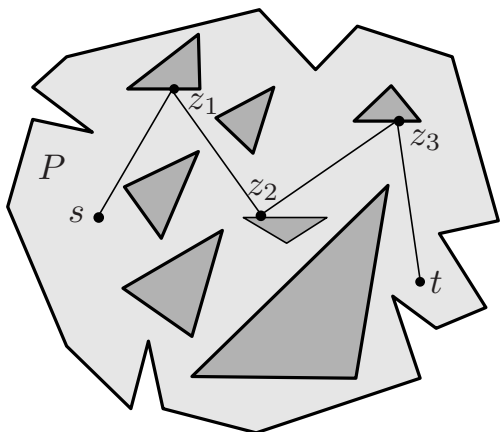


Figure 10: The links connecting pairs of weakly visible edges of P has formed a diffuse reflection path.

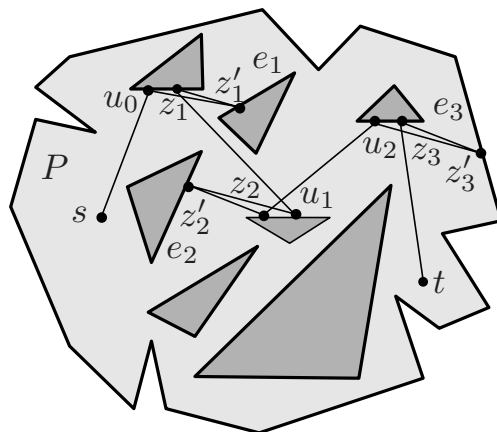


Figure 11: Two additional links are constructed for every $z_i \neq u_{i-1}$ to form a diffuse reflection path.

4 Computing a diffuse reflection path using the visibility graph

In this section, we present an $O(n^2)$ time algorithm for computing a diffuse reflection path from s to t inside a polygon P with or without holes using the edge-edge visibility graph of P . The number of reflections in the path produced by the algorithm can be at most 3 times than that of an optimal $drp(s, t)$. The algorithm first finds a sequence of edges of P using BFS and then constructs a diffuse reflection path which reflects on these edges.

Let V_e denote the set of all edges of P . Two edges of P are said to be weakly visible if some internal point of one edge is visible from an internal point of the other edge. The edge-edge visibility graph G_e of P is a graph with nodes V_e and arcs between nodes that correspond to a weakly visible pair of edges in P [8, 15]. The algorithm starts by constructing the edge-edge visibility graph G_e of P . Then two nodes representing s and t are added in V_e . The node s (or, t) is connected by arcs in G_e to those nodes in V_e whose corresponding edges in P are partially or totally visible from s (respectively, t). We have the following observation.

LEMMA 4.1 Between s and t , the number of reflections in any diffuse reflection path in P cannot be smaller than the number of edges of P in the shortest path between s and t in G_e .

Compute the shortest path from s to t in G_e using BFS. Let g_1, g_2, \dots, g_{k-1} be the sequence of edges of P corresponding to the nodes of V_e in the shortest path from s to t in G_e . Since g_i is weakly visible from g_{i+1} , for all i (see Figure 9), locate a pair of internal points $z_i \in g_i$ and $u_i \in g_{i+1}$, for all i , such that the segment $z_i u_i$ lies inside P . Let u_0 be a point in g_1 visible from s . Let z_{k-1} be a point in g_{k-1} visible from t . So, a sequence of links $su_0, z_1 u_1, \dots, z_{k-2} u_{k-2}, z_{k-1} t$ has been constructed. If $z_i = u_{i-1}$, for all i (see Figure 10), then we have a diffuse reflection path $sz_1, z_1 z_2, \dots, z_{k-1} t$ with the minimum number of reflections. Otherwise, for every $z_i \neq u_{i-1}$,

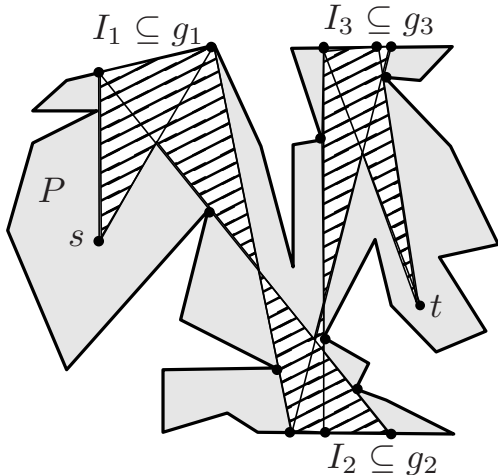


Figure 12: The intervals I_1, I_2, I_3 on edges are not empty.

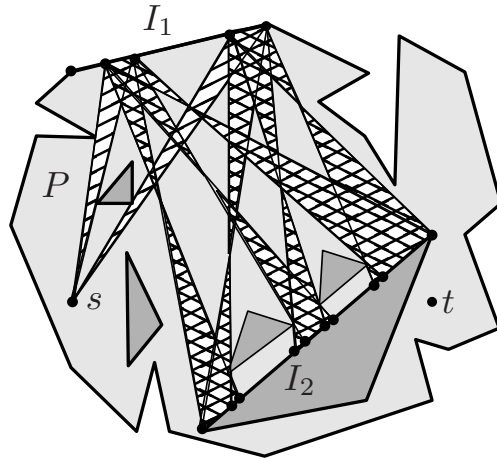


Figure 13: There are two sub-intervals in I_1 due to one hole, and six sub-intervals in I_2 due to two holes.

locate a point z'_i on an edge e_i of P such that all points of g_i are visible from z'_i , and then add two links $u_{i-1}z'_i$ and z'_iz_i to connect u_{i-1} with z_i (see Figure 11). The point z'_i can be located by extending the edge g_i to the nearest polygonal edge e_i and then choosing a point arbitrary close to the intersection point. Hence, $(su_0, u_0z'_1, z'_1z_1, z_1u_1, \dots, u_{k-2}z'_{k-1}, z'_{k-1}z_{k-1}, z_{k-1}t)$ is $drp(s, t)$.

Observe that since the path can have at most $3k$ links and any optimal diffusion reflection path from s to t must have at least k links by Lemma 4.1, the number of reflections in the path can be at most 3 times than that of an optimal path.

Let us analyze the time complexity of the algorithm. The algorithm locates all pairs of weakly visible edges in P as follows. Using the algorithm of Ghosh and Mount [9], the algorithm first computes all visible pairs of vertices (say, E) of P in $O(n \log n + E)$ time. During the process of computation, the algorithm also constructs *funnel sequences* with edges as bases of the funnels. By traversing these funnel sequences, all pairs of weakly visible edges of P can be located. In addition, links connecting pairs of weakly visible edges of P can also be constructed using funnel sequences. The entire computation takes $O(n^2)$ time. By traversing through the funnel sequences again, edges g_1, g_2, \dots, g_{k-1} can be extended to the respective nearest edges in P to locate points $z'_1, z'_2, \dots, z'_{k-1}$ respectively. These points can be located in $O(n^2)$ time. Hence, the entire diffusion reflection path can be computed in $O(n^2)$ time. We summarize the result in the following theorem.

THEOREM 4.2 Using the edge-edge visibility graph of P , a diffuse reflection path from s to t can be computed in $O(n^2)$ time, and the number of reflections in the path can be at most three times than that of an optimal diffusion reflection path.

Let us discuss the problem of choosing an appropriate point $z_i \in g_i$ on edges $g_1, g_2, \dots,$

g_{k-1} such that the segments $sz_1, z_1z_2, \dots, z_{k-1}t$ lie inside P . Consider the case when P is a polygon without holes. Let I_1 be the set of all points of g_1 that are visible from s (see Figure 12). Similarly, let I_2 be the set of all points of g_2 that are visible from some point of I_1 . If the interval I_2 is not empty, then the next interval I_3 is again defined to be the set of all points of g_3 that are weakly visible from I_2 . If all intervals I_1, I_2, \dots, I_{k-1} are non-empty, and there exists a point $z_{k-1} \in I_{k-1}$ such that z_{k-1} is visible from t , then for all i , there exists a point $z_i \in I_i$ that is visible from z_{i+1} . Therefore, locate a point $z_i \in I_i$ from z_{i+1} , for all i , such that $z_i z_{i+1}$ is a segment lying inside P . By Lemma 4.1, the path $(sz_1, z_1z_2, \dots, z_{k-1}t)$ is an optimal diffuse reflection path from s to t . It can be seen that I_{i+1} can be computed from I_i in $O(n)$ time by computing the *hourglass* between g_i and g_{i+1} [8, 10]. Therefore, all intervals can be computed in $O(n^2)$ time. Hence, $(sz_1, z_1z_2, \dots, z_{k-1}t)$ can be computed in $O(n^2)$ time. We have the following theorem.

THEOREM 4.3 Given a sequence of k edges of a polygon P without holes such that (i) the first and last edges are partially or totally visible from points s and t respectively, and (ii) every pair of consecutive edges in the sequence are weakly visible, a diffuse reflection path of k reflections using this sequence of edges from s to t can be computed, if such a path exists, in $O(n^2)$ time.

COROLLARY 4.4 If k is the smallest such sequence of edges in P , then the diffuse reflection path computed by the algorithm is optimal.

Let us consider the other case when P is a polygon containing holes. Let I_1 be the set of all points of g_1 that are visible from s (see Figure 13). Since P contains holes, I_1 may consist of two or more disjoint sub-intervals. Again, let I_2 be the set of all points of g_2 that are visible from some point of sub-intervals of I_1 . Observe that the number of sub-intervals in I_2 can be more than the number of sub-intervals in I_1 as P contains holes (see Figure 13). If I_2 has at least one sub-interval, compute sub-intervals of I_3 . This process is repeated till sub-intervals of I_{k-1} on g_{k-1} are computed. Let b_1, b_2, \dots, b_{k-1} be a sequence of sub-intervals such that $b_i \in I_i$, for all i , and every point of b_i is visible from some point of b_{i-1} . So, a diffuse reflection path $(sz_1, z_1z_2, \dots, z_{k-1}t)$, where $z_i \in b_i$ for all i , can be computed as stated earlier. However, the total number of sub-intervals on all edges in the sequence computed by the algorithm can be exponential. Therefore, the method of computing sub-intervals explicitly on all edges of a given sequence does not lead to any polynomial time algorithm for polygons with holes. Even after the union of overlapping sub-intervals is taken for every edge in the sequence, the total number of disjoint sub-intervals on all edges may still become exponential.

5 Concluding remarks

We have presented algorithms for computing diffuse reflection paths from a light source s to a target point t inside P . As stated in the introduction, there are two types of reflections of light: diffuse and specular. So, it is natural to ask for a specular reflection path from s to t inside P . There is no known algorithm for this problem. Note that unlike diffuse reflection, it has been shown that a path of specular reflections may not always exist for all polygons and for all positions of s inside a polygon [12, 18].

Let g_1, g_2, \dots, g_{k-1} be a sequence of edges of P without holes such that g_i is weakly visible from g_{i+1} , for all i , and g_1 and g_{k-1} are visible from s and t respectively. Given a sequence of such edges, it is possible to find a specular reflection path (if it exists) from s to t passing through these edges in the given order as follows. Compute the interval $I_1 \subseteq g_1$ visible from s . Let s_1 be the position of the virtual source of s with respect to g_1 . Let I_2 be the set of all points of g_2 such that for any point $z \in I_2$, (i) the line segment zs_1 intersects I_1 , and (ii) zs_1 does not intersect the sides of the hourglass in P between g_1 and g_2 . If the interval I_2 is empty, then there is no specular reflection path from s to t passing through the given sequence of edges. So, we assume that I_2 is not empty. Analogously, compute the corresponding intervals I_3 of g_3 , I_4 of g_4, \dots, I_{k-1} of g_{k-1} . Then compute the interval $I'_{k-1} \subseteq I_{k-1}$ visible from t . Now a specular reflection path can be computed from s to t using these intervals in the reverse order. The algorithm runs in $O(n^2)$ time as all hourglasses can be computed in $O(n^2)$ time by the algorithm of Ghosh and Mount [9]. It can be seen that this result on specular reflections is analogous to Theorem 4.3.

Let us consider the problem of computing a minimum link path between two given points s and t inside a polygon P with holes [8, 14]. It can be seen from the last section that if segments connecting z_i with u_{i-1} are added, for all i , then the path $(su_0, u_0z_1, z_1u_1, \dots, u_{k-2}z_{k-1}, z_{k-1}t)$ becomes a link path. Note that this sub-optimal algorithm is simpler than the optimal algorithm given by Mitchell et al. [14] which involves computing arrangements of line segments.

References

- [1] B. Aronov, A. Davis, T. Dey, S. P. Pal, and D. Prasad. Visibility with multiple reflections. *Discrete & Computational Geometry*, 20:61–78, 1998.
- [2] B. Aronov, A. Davis, T. Dey, S. P. Pal, and D. Prasad. Visibility with one reflection. *Discrete & Computational Geometry*, 19:553–574, 1998.
- [3] B. Aronov, A. R. Davis, I. Iacono, and A. S. C. Yu. The complexity of diffuse reflections in a simple polygon. In *Proceedings of the 7th Latin American Symposium on Theoretical Informatics*, Lecture Notes in Computer Science, volume 3887, pages 93–104. Springer, Germany, 2006.
- [4] B. Chazelle. Triangulating a simple polygon in linear time. *Discrete & Computational Geometry*, 6:485–524, 1991.
- [5] B. Chazelle, H. Edelsbrunner, M. Grigni, L. J. Guibas, J. Hershberger, M. Sharir, and J. Snoeyink. Ray shooting in polygons using geodesic triangulations. *Algorithmica*, 12:54–68, 1994.
- [6] J. Foley, A. van Dam, S. Feiner, J. Hughes, and R. Phillips. *Introduction to Computer Graphics*. Addison-Wesley, Reading, MA, 1994.
- [7] S. K. Ghosh. Computing visibility polygon from a convex set and related problems. *Journal of Algorithms*, 12:75–95, 1991.

- [8] S. K. Ghosh. *Visibility algorithms in the plane*. Cambridge University Press, Cambridge, UK, 2007.
- [9] S. K. Ghosh and D. M. Mount. An output-sensitive algorithm for computing visibility graphs. *SIAM Journal on Computing*, 20:888–910, 1991.
- [10] L. J. Guibas and J. Hershberger. Optimal shortest path queries in a simple polygon. *Journal of Computer and System Science*, 39:126–152, 1989.
- [11] J. Hershberger. Finding the visibility graph of a polygon in time proportional to its size. *Algorithmica*, 4:141–155, 1989.
- [12] V. Klee. Is every polygonal region illuminable from some point? *American Mathematical Monthly*, 76:180, 1969.
- [13] D. T. Lee and F. P. Preparata. Euclidean shortest paths in the presence of rectilinear barriers. *Networks*, 14:393–415, 1984.
- [14] J. S. B. Mitchell, G. Rote, and G. Woeginger. Minimum-link paths among obstacles in the plane. *Algorithmica*, 8:431–459, 1992.
- [15] J. O’Rourke and I. Streinu. The vertex edge visibility graph of a polygon. *Computational Geometry: Theory and Applications*, 10:105–120, 1998.
- [16] S. P. Pal, S. Brahma, and D. Sarkar. A linear worst-case lower bound on the number of holes in regions visible due to multiple diffuse reflections. *Journal of Geometry*, 81:5–14, 2004.
- [17] D. Prasad, S. P. Pal, and T. Dey. Visibility with multiple diffuse reflections. *Computational Geometry: Theory and Applications*, 10:187–196, 1998.
- [18] G. T. Tokarsky. Polygonal rooms not illuminable from every point. *American Mathematical Monthly*, 102:867–879, 1995.