RECONSTRUCTION OF PARALLEL LINE SEGMENTS FROM ENDPOINT VISIBILITY INFORMATION

S. K. WISMATH*

Abstract. In general, visibility reconstruction problems involve determining a set of objects in the plane that exhibit a specified set of visibility constraints. In this paper, an algorithm for reconstructing a set of parallel line segments is presented, from specified visibility information contained in an extended endpoint visibility graph. The algorithm runs in polynomial time and relies on simple vector arithmetic to generate a system of linear inequalities.

1. Introduction. There are many problems in computer science that are directly or indirectly concerned with the visibilities inherent among a collection of objects in the plane. Such problems arise in graphics, motion planning, computational geometry, and VLSI design, for example. Although the type of objects and the definition of visibility frequently vary, most results that deal explicitly with visibility issues focus on either the computational or structural properties of visibility.

Given a set S of n disjoint line segments in the plane, the **endpoint visibility graph** $G_e(S)$ consists of $2n$ vertices corresponding to the endpoints of S and an edge set representing pairs of visible endpoints: vertices u and v of $G_e(S)$ are joined by an edge if and only if the corresponding endpoints of S are **visible** (i.e. connectable by a line segment that intersects no other segment of S). Traditionally, this edge set is augmented by the segments themselves, i.e. if u and v are endpoints of the same segment of S, then (u, v) is included as an edge. Research on visibility graphs has focused on the following main problems; see O'Rourke [5] for a more complete survey.

- Construction: Given a set of objects determine the associated visibility graph.
- Recognition: Given a graph, determine whether it represents the visibility graph of some set of objects. The general problem is not known to be in NP, but Everett [3] has shown that for line segments forming a polygon, the problem is in P-space.
- Characterization: Can the class of visibility graphs be nicely characterized? Several properties of visibility graphs have been identified, particularly for special classes of polygons.
- Reconstruction: Given a visibility graph G, create a layout of objects consistent with G.

In this paper, it is the reconstruction problem that is solved for a set S of line segments to be embedded on specified parallel tracks when an extension of a visibility graph is given. There are very few reconstruction results of this form in the literature. For the case of line segments, related results can be found in [1], [7], and [8]. Variants of the *polygon* reconstruction problem are discussed in [2], and [4].

* Department of Mathematics and Computer Science, University of Lethbridge, 4401 University Drive, Lethbridge, AB, Canada, T1K 3M4; E-mail: wismath@hg.uleth.ca

Let $G_e(S)$ denote the embedded endpoint visibility graph of a set S of line segments as defined above. Then for each endpoint v of S, the circular ordering of the endpoints visible to v is known. For an endpoint u visible to v, define *Stab* (u,v) as the name of the first segment of S encountered by the ray from v in the direction of $\bar{uv}$. The symbol "∞" will be used, if this ray encounters no segment of S. Define $G_e^{\pi}(S)$ as the graph $G_e(S)$ supplemented with this *stab* information, namely for each edge (u,v) of $G_e(S)$ the two stabs *stab* (u,v) and *stab* (v,u) are stored at vertices v and u respectively, inserted in sorted order. Thus, the cyclic ordering about each vertex v of $G_e^{\pi}(S)$ consists of pairs of "visibility vectors", spaced π radians apart. Given $G_e^{\pi}(S)$, the "view" from a particular endpoint v can be computed by a single rotational sweep about v. This view is slightly weaker than the traditional *visibility polygon* which is normally specified as an ordered collection of explicit corner points on the segments of S. If u and v are endpoints of the same line segment, and S consists of vertical line segments, then the stabbing information can be used to determine those neighbours of u that lie strictly to the left of u and those that lie to the right of u. Denote by $L(u)$ (respectively $R(u)$), the clockwise ordered set of neighbours of vertex u to the left (respectively right) of u in G_e^{π}.

2. Reconstruction from visibility information. The general reconstruction problem is to determine a set $S = \{s_1, s_2, ...s_n\}$ of non-intersecting line segments in the plane, whose visibility graph is equivalent to a given target visibility graph. In this paper, we consider the restricted problem in which the segments are to be embedded on specified vertical tracks, defined by the lines $X = t_1,\ X = t_2,\ \dots, X = t_n,\ (t_i \neq t_j)$ and the target visibility graph is the extended visibility graph G_e^{π} described in the previous section. Initially, assume that the lengths of each segment $l_1, l_2, \dots\ l_n$ are also known, and then only the bottoms of the segments, denoted by $b_1, b_2, \dots b_n$ must be determined, and segment s_i will have bottom coordinates (b_i, t_i) and top coordinates $(b_i + l_i, t_i)$. Finally, the track values must also be available for endpoints - denote by $t(p)$ the track value for endpoint p.

Now, given a target visibility graph G_e^{π}, the coordinates of the endpoints of S are determined as follows. The visibility information of G_e^{π} must be respected in two ways. In the embedding of S, visible endpoints must be visible, and non-visible endpoints must be blocked by some segment. It is the stabbing information available in G_e^{π} that will be used to constrain the locations of the endpoints to ensure visibility. Each pair of visible endpoints in G_e^{π}, generates two stabs - one in each direction. The *forward stab* will be the vector pointing in the positive X direction, and the *backward stab* in the negative direction. Consider a *stab* of the form in figure 1, in which $stab(b_i, b_j)$ is s_k. The forward *stab* generates two inequalities restricting the layout of segment s_k, namely that the bottom of b_k be below the *stab* line and that the top be above it. For this particular case, these two inequalities can be expressed in terms of vectors as:

$$b_k < b_i + \frac{t_k - t_i}{t_j - t_i}(b_j - b_i)$$

$$b_k + l_k > b_i + \frac{t_k - t_i}{t_j - t_i}(b_j - b_i)$$

The backward *stab* can be expressed in a similar fashion as a restriction on s_h as follows:

$$b_h < b_j + \frac{t_h - t_j}{t_i - t_j}(b_i - b_j)$$

and

$$b_h + l_h > b_j + \frac{t_h - t_j}{t_i - t_j}(b_i - b_j)$$

Stabs to infinity are ignored. Label the inequalities produced in this manner, Type 1 inequalities.

It is also imperative that visibilities not be blocked in the reconstructed layout and thus for each visibility, inequalities must be generated to ensure that line segments are constructed on the appropriate side of the visibility (i.e. above or below it). Let p and q be a pair of visible endpoints, with $t(p) < t(q)$ and with *stab* $(p,q) = r$ and *stab* $(q,p) = o$. Refer to figure 2.

Then it is necessary to ensure that all the intervening segments between $t(o)$ and $t(r)$ do not intersect the line $l(p,q)$. Determining all such segments, for each visibility, would be prohibitively expensive and unnecessarily redundant. Instead, exactly six critical segments will be determined and constrained.
Let p^+ be the first vertex clockwise from o in $L(p)$.
Let p^- be the first vertex counterclockwise from o in $L(p)$.
Let q^+ be the first vertex counterclockwise from r in $R(q)$.
Let q^- be the first vertex clockwise from r in $R(q)$.
By constraining p^+ and q^+ to lie above the line $l(p,q)$ and p^- and q^- to be below it, the two stabbing rays are guaranteed to be unblocked to o and r.

The bars that lie between $t(p)$ and $t(q)$ must also be considered. Let $p^\uparrow$ be the first vertex counter-clockwise from q in $R(p)$, and let $p^\downarrow$ be the first vertex clockwise from q in $R(p)$.
Constraining $p^\uparrow$ to lie above line $l(p,q)$ and $p^\downarrow$ to be below it ensures that the visibility between p and q is not obstructed in the reconstruction. Note that $p^\uparrow$ and $p^\downarrow$ do not necessarily lie ***between*** p and q, however any segments between and above p and q must also be above the visibility segment $p, p^\uparrow$, and will ultimately be so constrained. Also, note that any of $p^\downarrow, p^\uparrow, p^+, p^-, q^+, q^-$ may be undefined, in which case no inequality is created. These six constraints (for each visibility pair) are easily expressed as inequalities and are labelled Type 2 inequalities. For example, q^+ produces the inequality:

$$q^+ > p + \frac{t(q^+) - t(p)}{t(q) - t(p)}(q - p)$$

In total, if there are E pairs of visible endpoints in G_e^π, then $10E$ linear inequalities are generated (two for each of the $2E$ Type 1 inequalities and $6E$ of the second type). However, obtaining a solution to this set of linear inequalities is not trivial. One technique is to create a fictitious objective function and solve the associated linear programming problem. Since linear programming is known to be in the complexity class P, thus the reconstruction problem also has a polynomial time solution. Moreover since techniques, such as the simplex method, work efficiently in practice, a solution to these inequalities can, in fact, be obtained (on average) quickly.

Alternately, it is possible to convert the system of linear inequalities to a system of linear *equalities* by introducing a slack variable into each inequality. Each slack variable must be strictly greater than zero and has a geometric significance, namely the vertical distance of the constrained endpoint from the associated visibility stab or line. Solving the resulting system of equalities (by Gaussian Elimination, for example), yields a parametric solution to the given reconstruction problem which represents the set of *all* valid layouts. Unfortunately, determining even a single solution *with all slack variables strictly greater than zero*, is, in general, difficult (although solvable in polynomial time).

Finally, note that the inequalities generated are of a highly restricted form - each contains exactly three unknowns, and thus there may exist more efficient techniques for obtaining solutions directly.

2.1. Proof of Correctness. It is clear that the $10E$ linear inequalities generated by the algorithm are necessary. Let $\vec{b}'$ be a particular solution to the set of linear inequalities and let S' be the associated set of line segments. We must show that $G_e^\pi(S')$ is equivalent to the given target visibility graph G_e^π. Let S be any set of segments whose visibility graph is indeed equivalent to the target graph. Note that since S' is embedded on parallel tracks, no pair of segments can intersect.

LEMMA 2.1. : *If p and q are visible endpoints in S then p', q' are visible endpoints in S'.*

Proof: Let p and q be a pair of visible endpoints in S such that p' and q' are not visible in S'. Let the leftmost segment in S' that blocks the visibility of p' and q', have top and bottom endpoints u' and v'. Assume, without loss of generality, that in S, the segment (u, v) lies above the (p, q) visibility. Then there are two cases to consider.

- $p^\dagger = v$, in which case there is an inequality forcing v to be above the line (p, q).
- $p^\dagger \neq v$, in which case, $p^\dagger$ is an endpoint constrained to be above the line (p, q). Since p and $p^\dagger$ are a pair of visible endpoints, v must lie above the line $(p, p^\dagger)$ in S. Iterating this argument, proves the existence of a sequence of inequalities which ultimately restrict v to be above the line segment (p, q). And hence v' is similarly constrained to be above the line segment (p', q') in S' and therefore, the segment u', v' does not block the visibility of p' and q'.

□

LEMMA 2.2. : *If p and q are not visible endpoints in S, then p', q' are not visible endpoints in S'.*

Proof Sketch: Assume $t(p) < t(q)$. The proof involves a "shadow" argument. Imagine a light source at p. Since q is not visible to p, it lies in the umbra of p along the line $X = t(q)$. It can be argued that the shadows on this line must be consistent in S' or the visibility graph information will be violated. □

3. Extensions and Open Problems. A first extension to the previous result is to simply note that the heights of each bar are also linear in the resulting inequalities and thus could be considered as unknowns (with the restriction that $l_i > 0$).

A second observation is that if the *stab*bing information, $stab(u, v)$ explicitly denotes a particular point on a segment, rather than simply the name of the segment, then the type 1 constraints become equalities; however the type 2 constraints remain as inequalities. Thus, the complexity of the problem is not obviously lowered, even if this explicit *stab*bing information is specified in G_e^π for each endpoint. In particular, this extension of G_e^π is constructable (in linear time) if the visibility polygons of each endpoint are specified.

The technique used in this paper does not easily generalize. The reconstruction problem is still open for the following two related cases:

- for parallel line segments without specified track information.
- for line segments on non-parallel tracks (for example on two sets of parallel tracks).

Finally, it should be noted that the general reconstruction problem has some relation to oriented matroids and arrangements of pseudolines. In particular, there is a fairly straightforward reduction from the Pseudoline Stretchability problem (shown by Shor [6] to be NP-Complete) to the line segment reconstruction problem.

REFERENCES

[1] T. Andreae, Some Results on Visibility Graphs, *Discrete Applied Mathematics*, Vol. 40, No. 1, (1992), 5-18.

[2] C. Coullard, A. Lubiw, Distance Visibility Graphs, *J. of Computational Geometry and Applications*, Vol. 2, No. 4 (1992) 349-362.

[3] H. Everett, D. Corneil, Recognizing Visibility Graphs of Spiral Polygons, *J. of Algorithms* 11 (1990), 1-26.

[4] H. Everett, A. Lubiw, J. O'Rourke, Recovery of Convex Hulls from External Visibility Graphs, *Proceedings of the 5th Canadian Conference on Computational Geometry*, Waterloo, (1993), 309-314.

[5] J. O'Rourke, *Computational Geometry Column 18*, SIGACT News, Vol.24, No. 1 (1993), pp.20-25.

[6] P. W. Shor, Stretchability of Pseudolines is NP-Hard, DIMACS Series in Discrete Mathematics and Theoretical Computer Science, Vol. 4, 1991, pp. 531-554.

[7] R. Tamassia, I. Tollis, A Unified Approach to Visibility Representations of Planar Graphs, *J. of Discrete and Computational Geometry* 1 (1986), 321-341.

[8] S. K. Wismath, Characterizing Bar Line-of-Sight Graphs, *Proc. 1st ACM Symposium on Computational Geometry*, Baltimore (1985), 147-152.

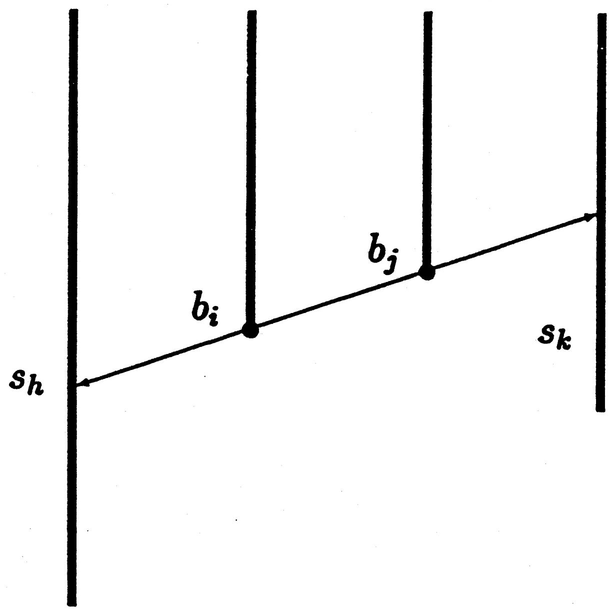

Figure 1

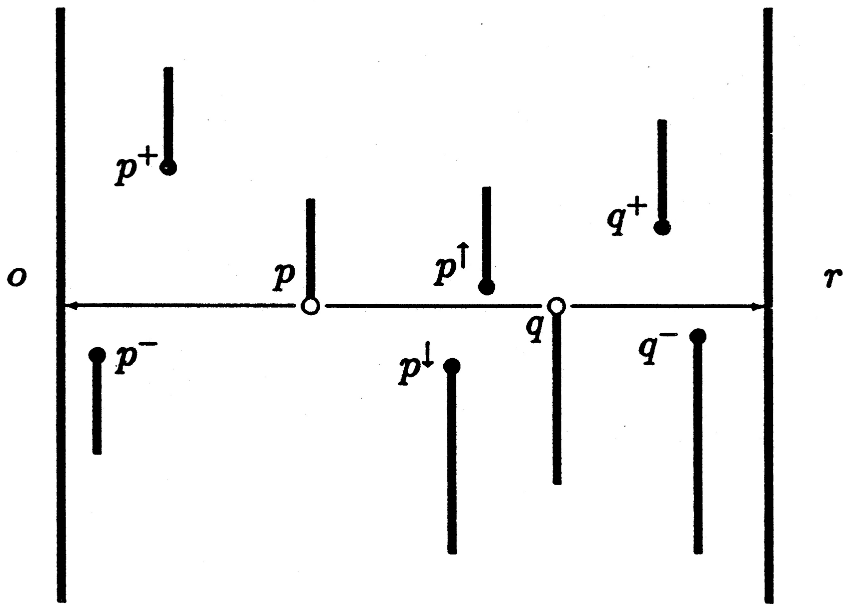

Figure 2