Minimum Polygon Covers of Parallel Line Segments

Henk Meijer and David Rappaport
Department of Computing and Information Science
Queen's University
Kingston, Ontario K7L 3N6

(Extended Abstract*)

Abstract: In this note we show that, given a set *S* of *n* parallel line segments, a perimeter minimizing polygon that intersects every segment of *S* can be found in $\Theta(n \log n)$ time.

Introduction.

The problem of intersecting a collection of objects with a common line has received considerable attention in the area of discrete and computational geometry. Such a line is known as a line transversal in the mathematics literature, or a line stabber in the computer science. One can generalize the notion of stabbing with a line to stabbing with a convex polygon. This problem can be attributed to [Tamir]. In [Goodrich and Snoeyink] an $O(n \log n)$ algorithm is given to determine whether a set of parallel lines can be stabbed by the boundary of a convex polygon.

We look at a related problem. Rather than restrict ourselves to stabbing objects with the boundary of a polygon we will allow the interior of the polygon to stab as well. In essence we want to find a polygon such that at least one point of every segment is covered. In this note we present an algorithm to compute the polygon of smallest perimeter that covers a set of parallel line segments with its interior and boundary.

Computing minimum polygon covers.

Let *S* be a set of *n* parallel line segments. Without loss of generality we can assume these line segments to be vertical. We define a *polygon cover* of S as a simple polygon that intersects every segment of *S* with its interior or with its boundary. We represent a polygon by its boundary. Therefore, we use $(p_0, p_1, ..., p_k)$, a list of vertices traversed clockwise on the boundary of P, to represent P. In order to avoid circularity of the list we assume that $p_0 = p_k$. Let any contiguous sublist of a polygon representation be denoted as a *polygonal chain*. Let *conv*(X) denote the *convex hull* of a set of points X, that is, the smallest convex region containing X, and let *CH*(X) denote a list of vertices that represent the boundary of conv(X). Our algorithms will be concerned with summing lengths of edges on the boundary of polygon covers. The sum of the lengths of the edges of a polygonal chain X is denoted by *len(X)*. Given a polygon P, len(P), should be understood as the sum of the boundary edges of P. A *minimum polygon cover* of *S* is a polygon cover of *S*, P, such that len(P) is minimized over all polygon covers of S.

We state all lemmas and theorems without proof. Proofs may be found in the complete paper.

Lemma 1: Every minimum polygon cover of a set of line segments is convex.

This research was supported by the Natural Sciences and Engineering Research Council of Canada under grants A–0282 and A–9204.

* A complete version of this result can be found in Queen's University Technical Report CISC 90–279.

Let *B* and *T* denote the set of all bottom and top endpoints respectively of the segments in S. Let b_L and b_R respectively denote the leftmost and rightmost points in B, breaking ties by choosing the point with the largest y–coordinate. Let *UpH(B)* denote the upper half hull of CH(B) represented by the sublist of CH(B) beginning at b_L and ending at b_R. Similarly let t_L and t_R denote the leftmost and rightmost points in T, breaking ties by choosing the point with smallest y–coordinate. Then, *LoH(T)* is the lower half hull of CH(T), represented by a sublist of CH(T) beginning and ending at t_L and t_R respectively.

We denote the subset of S that intersects the vertices of a polygonal chain X as S(X). Similarly we use s(x) to denote the line segment in S (if one exists) that intersects a point x.

Lemma 2: Every polygon cover of S'=S(UpH(B)) ∪ S(LoH(T)) is also a polygon cover of S.

Lemma 3: Every minimum polygon stabbing cover passes through the segments $s(t_L)$ $s(b_L)$ $s(t_R)$ and $s(b_R)$.

We define some operations on lists. Given a list A, *rev(A)* denotes the list in reverse order. If A and B are two lists then A + B denotes the concatenation of list B to list A. If a list C = A + B then C – B denotes the list A. Given a list L, we use {L} to denote a set consisting of the elements in L.

Lemma 4: If $s(t_L) \neq s(b_L)$ and $s(t_R) \neq s(t_L)$ then the polygon represented by concatenating the lists UpH(B) + rev(LoH(T)) is a minimum polygon cover of S.

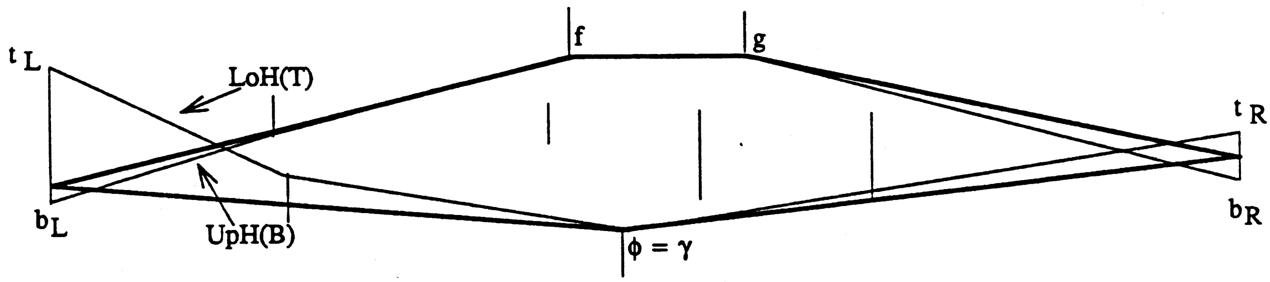

Figure 1.

If $s(t_L) = s(b_L)$ then we abbreviate the segment $[b_L, t_L]$ by s_L. Similarly, if $s(t_R) = s(b_R)$ then we abbreviate $[b_R, t_R]$ by s_R. If $[b_R, t_R] \in S$ then it is not necessary to cover both b_R and t_R. Rather, only a single point on the segment s_R needs to be covered. A similar situation occurs on the left with the segment s_L. See figure 1. An algorithm MINPOLYSTAB conveys this strategy in more detail.

ALGORITHM MINPOLYSTAB

Input: A set of vertical line segments S.

Output: P, a minimum polygon cover.

Step 1. Compute UpH(B) and LoH(T) as discussed above.

Step 2. Consider all the points in B with the largest y–coordinate. Let f and g be the leftmost and rightmost of these points. Similarly of all points in T with the smallest y–coordinate let ϕ and γ be the leftmost and rightmost.

RUp ← subchain of UpH(B) beginning at g and ending at b_R;

RLo ← subchain of LoH(T) beginning at γ and ending at t_R;

LUp ← subchain of UpH(B) beginning at f and ending at b_L;
LLo ← subchain of LoH(T) beginning at ϕ and ending at t_L;

Step 3. if $s(t_R) \neq s(b_R)$ then
RIGHTCOVER ← RUp + rev(RLo)
else
Find chains U and V both terminating at the same point r on s_R, that covers $\{RUp - b_R\} \cup \{RLo - t_R\} \cup \{s_R\}$ and minimizing len(U) + len(V);
RIGHTCOVER ← U + rev(V);

step 4. if $s(t_L) \neq s(b_L)$ then
LEFTCOVER ← LUp + rev(LLo)
else
Find chains U and V both terminating at the same point r on s_L, that covers $\{LUp - b_L\} \cup \{LLo - t_L\} \cup \{s_L\}$ and minimizing len(U) + len(V);
LEFTCOVER ← V + rev(U);

step 5. P ← LEFTCOVER + RIGHTCOVER. ∎

The correctness of algorithm MINPOLYSTAB follows as a consequence of the following lemma.

Lemma 5. There exists a minimum polygon cover that passes through every point in B with maximum y–coordinate and through every point in T with minimum y–coordinate.

Addressing the problem of computing the polygonal chains U and V as described above we must first consider the following subproblem.

Given two points p and q and a vertical line segment defined by its top and bottom endpoints [t,b] we determine the point r such that r is a point in [t,b], and the sum of the Euclidean distances d(p,r) + d(r,q) is minimized. We will make use of a function,

$\eta(p,q,[t,b])$

to return the value of such a point r given p, q, and [t,b]. This is a variant of Heron's problem, see [Courrant and Robbins] for a simple geometric solution.

We present an algorithm to compute the chains U and V as described in algorithm MINPOLYSTAB. We compute these chains on the right side. A symmetric approach is used to compute a solution for the left side. The points g and γ and the chains RUp, RLo, U and V are defined as in algorithm MINPOLYSTAB.

Algorithm RIGHT

Input: Polygonal chains RUp, RLo and the segment s_R.

Output: Polygonal chains U and V.

Step 1. Set p ← g ; q ← γ ; r ← $\eta(p,q,s_R)$.
u ← next(p, RUp); v ← next(q, RLo); U ← p; V ← q;
{next(x, L) is a function that returns the successor of x in the list L.}

Step 2. while u above [p, r] or v below [q, r] do
if u is above [p, r] then
U ← U + u;
p ← u;
r ← $\eta(p,q,s_R)$;
u ← next(u, RUp)

```
        else if v below [q, r] then
                V ← V + v;
                q ← v;
                r ← η(p,q,sR);
                v ← next(v, RLo);
endwhile.
```

∎

Lemma 6: At every iteration of the while loop in algorithm RIGHT the polygon formed by U + r + rev(V) is a minimum polygon cover of $\{U\} \cup \{V\} \cup (S_R)$.

We conclude with the main result of this paper.

Theorem: A minimum polygon cover for a set of n parallel segments can be constructed in O(n log n) time and this algorithm is optimal.

Discussion.

We have demonstrated an algorithm to compute a minimum polygon cover for a set of parallel line segments. Recently we have been able to extend our results to find the minimum polygon cover of a set of isothetic line segments in O(n log n) time [Lyons, Meijer and Rappaport]. We are also aware of a result due to [Souvaine] where the minimum polygon cover of a set of line segments that are the edges of a convex polygon can be found in O(n) time. However, the challenging problem of computing the minimum polygon cover of arbitrarily oriented line segments remains open.

References.

[Courrant and Robbins] R. Courrant and H. Robbins, What is Mathematics?, Oxford University Press (1941).

[Goodrich and Snoeyink] M. T. Goodrich and J. S. Snoeyink, Stabbing parallel segments with a convex polygon, Lecture Notes in Computer Science, Springer Verlag, pp. 231–242 (1989).

[Lyons, Meijer and Rappaport] K. Lyons, H. Meijer and D. Rappaport, Minimum polygon covers for isothetic line segments, manuscript in preparation (1990).

[Souvaine] D. Souvaine, personal communication (1990).

[Tamir] A. Tamir, Problem 4–2, Problems presented at the fourth NYU Computational Geometry Day (1987).