The Ice Rink Problem[1]

Bernard M.E. Moret,
Michael Collins, Jared Saia, and Ling Yu
Department of Computer Science
The University of New Mexico
Albuquerque, NM 87131, USA
email: `moret@cs.unm.edu`

ABSTRACT

Aerial search and rescue (or bombing) missions frequently have to locate a target within some prescribed area, running a search pattern that is constrained by the flight characteristics of the aircraft. The sweep is done in nearly straight lines at high speed, with the craft's sensors monitoring a narrow strip of ground underneath the craft; in between sweeps, the craft slows down for turns that are much wider than the sensor footprint. A similar task with a simpler geometry consists of cleaning an ice rink. We prove that the method used in ice rinks (the "Zamboni algorithm") is optimal and use it to develop heuristics for the more general task of sweeping an arbitrary simple polygon. We provide upper bounds on the performance of some of our heuristics and give the results of experiments showing that our heuristics produce solutions within a few percent of optimal.

1. Introduction

Search-and-rescue missions flown by fixed-wing aircraft and search-and-destroy missions flown by cruise missiles both require that a fast-moving aircraft search a predefined area (such as a simple polygon) to locate the target while conserving fuel and avoiding obstacles (terrain features or enemy threats). Typical sensors installed on these craft work well only during reasonably level flight and lose all detection power during sharp turns; thus the search pattern must consist of more or less straight sweeps (which can also be flown quite fast) alternating with much slower turns. A similar application is the airborne dispersal of pesticides or fertilizers from a fixed-wing plane. Further applications arise in various areas of manufacturing (machining, coating, etc.).

Minimizing the number of (sharp) turns is thus the main objective, since each turn represents an unproductive expenditure of time and fuel. Fast, fixed-wing craft typically have large turn radii; cruise missiles have even larger ones. In both cases the turn radius exceeds the width of the sensor footprint by a factor on the order of 2 to 5, so that the craft cannot easily fly two adjacent sweep lines—to do so requires a very long and wide turn in which the craft begins by actually moving away from the next sweep line. Additional constraints may include 3-dimensional considerations (craft are limited in climb rate, yet cannot fly too far above the ground if they are to be successful in locating the target, avoiding enemy detection, or fertilizing a field) and probabilistic considerations (given a probable initial location, certain areas have a high probability of containing the target).

We study the problem of routing such a craft in a complete sweep of a simple polygon under various constraints. We retain turn constraints and the typical ratios between turn radius and

[1]This work is supported by the Office of Naval Research under grants N00014-97-1-0244 and N00014-95-1-1010.

footprint, but do not as yet consider 3-dimensional aspects. Thus our problem can be modeled geometrically as that of covering a simple polygon with (nearly) straight strips of a fixed width, while minimizing the number of turns (and, incidentally, the total distance traveled). Most of us are familiar with a much simplified version of this problem: how does one sweep an ice rink with a machine that can sweep a fixed width, moves fast in straight line, but has to slow down considerably for turns, even then making large-radius turns? The Zamboni ice-rink machines are typically used in a fixed pattern all over the world: a wide oval that is shifted by (not quite) the width of a sweep at each pass.

In this paper we prove that the "Zamboni algorithm" is optimal for a class of polygons that we name "Zamboni polygons." We then use this result to develop heuristics to optimize sweep patterns for arbitrary simple polygons. Experimental results with a variety of polygonal shapes and a range of ratios between turn radius and sensor footprint indicate that the heuristics perform very well, typically returning solutions within a few percent of a simple lower bound. Theoretical upper bounds indicate that a class of heuristics based on convex decomposition returns solutions with no more than three times the number of turns required in an optimal solution for a class of polygons (related to star polygons) typical of applications.

2. Related Work

Most of the work on searching within a polygon has been done in the context of robotics, and thus mostly for on-line algorithms, either in terms of generating a shortest path from the current location to a target [6] or in terms of exploring an unknown polygonal region to locate an unknown target [12]. A few authors have investigated the search of known polygons for an unknown target but all have used special geometries (star polygons and lattices [7] or rectilinear polygons [9]) and none have included constraints on exploration patterns.

We are aware of three closely related problems: the pocket milling problem [11], the lawnmower problem [2], and the *d*-sweeper problem [16]. (A fourth problem, the minimal-link Hamiltonian tour in grids [13], is also closely related, but its grid setting is unrealistic in our applications.) All three require that the inside of a polygonal region be completely covered at minimal cost (typically with the shortest path). In the milling problem one also requires that the cutter never cut outside the boundary of the region (and thus never approach the boundary to within the cutting radius), whereas the sweeper is simply constrained to stay inside the polygon (but can walk along the wall if so desired) and, of course, one is allowed to cross a walkway when mowing the lawn. In [2], the lawnmower problem is shown to be NP-hard; in [3], a special case of the lawnmower problem where only certain points must be mowed (a type of TSP problem where customers need only be approached to within a certain radius) is shown to have constant-factor approximations; in [16] an asymptotic $\frac{4}{3}$ approximation is given for the *d*-sweeper problem (assuming that the sweep radius is small compared to the area to be swept). Arkin *et al.* [2] also showed that constant-factor approximations exist for the general lawnmower problem. Milling is discussed at length by Held [11], who gives a number of heuristics that match current practice and take into account additional goals of importance in machining. The grid setting for the minimum-link Hamiltonian problem is well solved for square grids in [13]. All of these problems are discussed in the recent report of Arkin *et al.* [1]; the survey of Mitchell [14] covers all of this work and more, particularly the various approaches based on approximation algorithms for the traveling salesperson problem.

Our problem comes closest to the lawnmower problem, but we may have additional constraints (not every area outside the polygon can be overflown: terrain or enemy forces may forbid certain areas) and our main criterion differs: not only do we want to minimize the total length of the sweep, we also want to minimize the number of (sharp) turns. Standard approximation techniques based on TSP approximations (e.g., by superimposing a grid and obtaining a tour that joins grid points) tend to produce very large numbers of sharp turns. Some recent work in minimizing the total turn angle along with other criteria in a bicriteria or multicriteria framework [4, 15] is of interest, but the total turn angle (over all turns) is not a good measure for us. Doddi *et al.* [10] recently demonstrated the power of randomized approaches in bicriteria optimization for some geometric problems, but their method again uses a grid and thus would lead to poor solutions.

3. The Problem

We formalize our problem as follows. Given a simple polygon, a footprint, a turn radius, and two precision requirements $\delta > 0$ and $\epsilon > 0$, find a route that covers the polygon and has length at most $1 + \delta$ times the shortest route and makes at most $1 + \epsilon$ as many turns as the route that makes the fewest turns. In order to gauge our results against a lower bound, we also merge the two criteria into a single time criterion by assigning different speeds to turns and to straight segments. In the simplest version, we ignore the straight segments (flown very fast) and only tally the number of turns (flown quite slowly), since they dominate the cost in most practical cases; even this simple version remains ambiguous—for instance, are we allowed to overfly the same area twice?—so we further refine our problem statement: "Given a simple polygon P and a footprint size w, place the fewest straight segments within the polygon such that (i) segments do not intersect and (ii) every point inside the polygon lies within distance at most w of some segment." The second condition ensures that the entire polygon is covered, while the first disallows covering any area more than once. (Allowing overlap coverage could reduce the number of turns by a factor of 2 in a simple polygon and by an arbitrarily large factor in a connected polygonal region—picture a connected region formed by regularly-spaced horizontal and vertical strips forming a grid pattern.)

In practice, we would want to add forbidden regions (along a boundary that corresponds to a cliff, for instance, or through forbidden airspace), different probability levels, and other similar constraints.

In most formulations our problem is NP-hard: the constraints placed in addition to those used in the lawnmower problem do not alter the tenor of the reduction used in [2].

4. Zamboni Shapes

Axis- or contour-parallel approaches have long been used in practice when covering an area with multiple sweeps is required—Held [11] mentions that these are the two standard approaches in pocket milling and we are all familiar with the approach taken by painters. The ice-rink pattern obeys the same philosophy. If we allow the machine (our aircraft) to move beyond the boundary of the polygon, then we can see that the shape of the boundary perpendicular to the long axis is (more or less) irrelevant, as long as that boundary is monotonic along the short perpendicular axis.

We define a *Zamboni polygon* to be a simple polygon composed of a rectangular middle piece flanked by two "endcaps," each consisting of a polygonal line monotonic with respect to the two end segments of the rectangle. In particular, any convex polygon is a Zamboni polygon, since we can regard it as composed of two endcaps with a degenerate middle piece. In practice, low-angle turns can be allowed and so we allow the middle piece to be distorted from a rectangle into a "tube" of constant width with small angle variations.

Whenever the ratio of the long side to the short side (the length of the polygonal line to the length of the translated segment) is sufficiently large and as long as the "endcaps" are convex or form a negligible fraction of the whole polygon, the ice-rink technique—what we call the *Zamboni algorithm*—is an optimal sweep for a Zamboni polygon. (The proof is easy, but lengthy and not particularly enlightening; we omit it here.) This suggests using Zamboni polygons in an approach to arbitrary simple polygons.

5. Our Approach

We studied two heuristic approaches, both based on the notion of Zamboni polygons.

- In one approach we directly attempt to decompose the polygon into (a variation of) Zamboni polygons. This task is at least as hard as decomposing a polygon into convex pieces when additional edges are allowed, so we resort to greedy algorithms; in any case, we are more interested in generating large pieces than in generating few pieces. We then use local iterative improvement, attempting to increase the size of the largest Zamboni piece identified. Finally, we remove it from the polygon and proceed recursively with the remaining pieces. Each Zamboni shape yields a number of sweep tracks; we solve a TSP subproblem to connect the tracks optimally, knowing the cost of each connection.

- In the other approach, we choose instead to use one characteristic of Zamboni polygons—the definition of their middle piece by a medial axis (the polygonal line used in the translation). We compute the skeleton of the polygon (with some noise-reduction methods to avoids detailed branches induced by small features of the perimeter), then decompose the skeleton into (generalized) polygonal lines (using turn angle constraints and a greedy approach to yield the longest possible line). Each polygonal line defines a generalized version of a Zamboni polygon (generalized because the distance from the perimeter to the medial axis need not remain constant) which can be swept by paralleling the polygonal line—the contour-following approach commonly used in milling. Then, as in the previous approach, we solve a TSP problem to connect the sweep lines.

Both approaches are fast (all parts run in quadratic time or less) and both take advantage of the properties of Zamboni polygons. Both approaches concentrate on the "core" of the search area and lend themselves well to an optimization based on maximizing early returns. Since decompositions can be guided by probability values as well as by geometry, both approaches naturally incorporate these parameters. Both approaches allow the incorporation of additional information or constraints, such as required sweep edges or lines (an important feature, since the nature of the terrain to be searched may dictate specific sweeps, such as a sweep along the edge of a forest or a coast). Both approaches also suffer from the creation of small artifacts—very small regions of the polygon that require a large number of short tracks and a large number of turns for very little area; this effect is more pronounced with the skeleton approach. In both cases, though, we can re-align some of the sweep to conform to the sweep pattern used in the adjacent (and much larger) area; we can also prioritize the TSP algorithm—basically by computing different solutions for different percentages of the polygon covered—under the reasonable assumption that the target is unlikely to be found in the last few percent of the area.

We implemented both methods using the LEDA library. Our skeleton computation considers only points at a minimum distance from the polygonal boundary in order to reduce noise effects and decrease the number of branches of the skeleton. Our decomposition method uses recursive ear pruning until a central convex piece is formed, then unfolds the recursion and merges pieces as it can to form a large polygon well-suited for parallel sweeps. In the process, new edges are introduced that often end up dictating the direction of sweep in adjacent pieces, with the advantage that fewer sweep directions are then required.

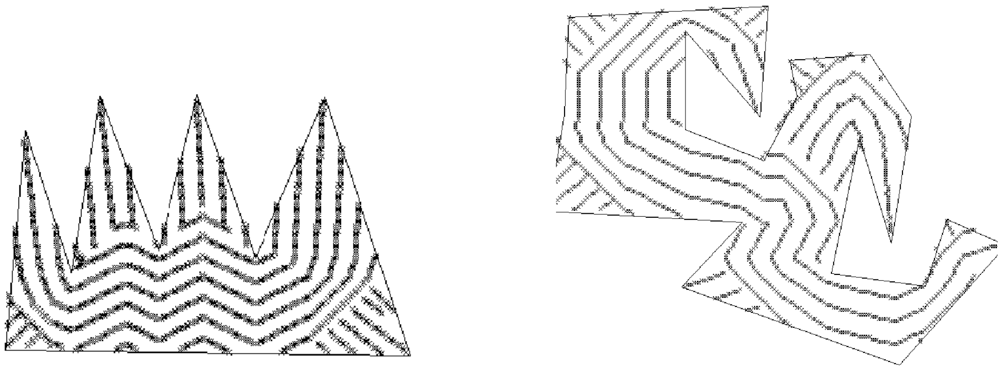

Figure 1: Sweep Tracks from Uniskeletal Decomposition

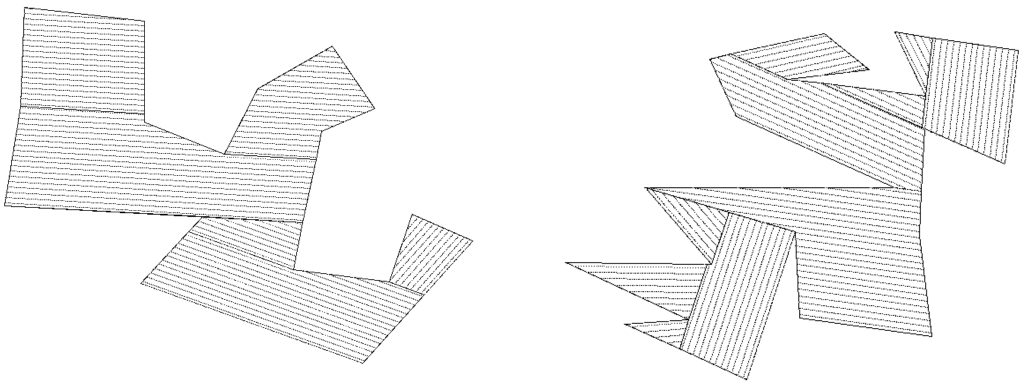

Figure 2: Sweep Tracks from Zamboni Decomposition

Typical results (showing the sweep lines before running the TSP algorithm) of the skeleton approach are shown in Figure 1 while typical results of the direct decomposition into Zamboni shapes are shown in Figure 2. Note that the left-hand polygon has 7 reflex vertices but gets decomposed into 6 polygons and uses only 3 directions of sweep.

6. Theoretical Results

We consider the simplest version of our problem, namely minimizing the number of straight-line segments (or, equivalently, turns) used in covering the area without overlap. If the area to be covered is convex, it is easy to see that the optimal solution sweeps the convex polygon in a direction determined by the narrowest strip (determined by a pair of parallel lines) that encloses the polygon. This direction, which can always be taken to be parallel to one of the edges of the polygon, is easily found and is used by our algorithm. Thus our algorithm is optimal for convex polygons. a single reflex chain.

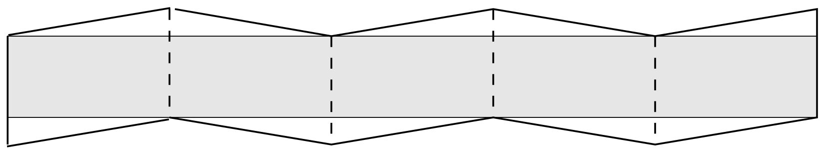

Figure 3: The Worst Case for Our Decomposition Algorithm

Theorem 6.1. *Given a simple polygon with k reflex vertices, our decomposition algorithm returns a solution that uses no more than $k+1$ times the number of turns required by any solution.*

The proof is trivial: with k vertices, a decomposition into convex pieces yields at most $k+1$ pieces, each of which is treated optimally. This worst case can actually be reached: Figure 3 illustrates a simple polygon where our algorithm will yield a decomposition into $k+1$ polygon, each swept in its major direction, whereas the optimal solution sweeps the shaded area horizontally; the pieces not included in the shaded area can be made arbitrarily small and thus made to contribute only a negligible number of segments.

We can further tighten the bound by observing that the worst case is only reached when our algorithm selects sweep axes that lead to a decomposition into $k+1$ pieces and when the optimal solution is able to use very long sweep lines that move across all $k+1$ pieces—as if the shapes were beads on a string, with our algorithm conducting a bead-by-bead sweep and the optimal solution sweeping along the string. We can conclude (again omitting the proof) that our algorithm remains within a ratio of k from the optimal in simple polygons in which, in any minimal convex decomposition, at most k convex pieces can be traversed along a single line without crossing a boundary of the polygon. The preponderant type of data in search applications is a polygon formed of a central area with branches radiating from it (roads, valleys, etc.), since the area to be searched is most often defined by a starting location and a number of likely directions of travel from the starting location. For these polygons, the value of k is 3 (if all branches are themselves convex) or little higher, so that our algorithm is guaranteed to return a solution that uses at most some small constant times the number of turns used in the optimal solution. (In practice, we expect our solutions to be much closer to the optimal than the upper bound suggests, since reaching the upper bound requires an unusual polygonal shape.)

If we now turn to the total length of the sweep, we can show even better results, because the total length of the segments is about the same in any non-overlapping solution (the only error introduced is due to rounding effects—i.e., to sweep segments used only to cover a tiny fraction of the sweep width). Thus, as the importance of the straight segments increases (because of slower aircraft or smaller turn radii or longer sweep lines), the approximation guarantees improve.

Finally, we studied the same problem on a rectilinear grid; in such a grid all turns are right-angle turns, so that we simply want to minimize the number of links in a covering tour of the polygon. This is a well-studied problem in the context of geometric graphs. Kranakis *et al.* [13] gave a $\frac{3}{2}$ approximation for cubic grids and derived a lower bound of $1.0232n^2$; we have improved the lower bound to $\frac{7}{6}n^2$ and generalized it to d-dimensional grids, again with an improved lower bound of $(1+\frac{1}{2d})n^{d-1}$ [8]. Our lower-bound technique uses a trade-off between the number of links entering (or leaving) a defined region within the grid and the maximum length of these links and can be refined through case analysis to yield tighter bounds. Moreover, this same technique can be used to study more complex grid shapes by decomposing them into rectangular pieces. While the problem on a rectilinear grid is of theoretical interest, however, it is not directly applicable to our original problem, since every turn on a rectilinear grid is a sharp turn.

7. Experimental Results

We generated polygons by hand rather than randomly, to take into account the fact that non-trivial search areas tend to have a lot of structure (they often follow terrain, such as valley floors, or infrastructure, such as road networks); most of our polygons had from 20 to 100 vertices. We used a set of 100 different polygons for our experiments, with somewhat unusual cases introduced deliberately into the mix. Then we ran experiments with varying sizes of footprints and turn radii; we used 6 different ratios between turn radius and footprint and observed the effect of relative scale by setting the footprint to be anywhere from 1/1000th of the diameter of the convex hull to 1/10th.

A trivial lower bound on the total length of tracks is the area of the polygon divided by the footprint and a trivial lower bound on the length of a turn is πR where R is the minimum turn radius. We also computed the optimal sweep for a rectangle of the same area as the original polygon and with its long dimension equal to the longest segment that can be included within the original polygon.

Results for the uniskeletal decomposition showed total track lengths within 13–28% of the lower bound on all "typical" polygons, but with far too many tracks due to the small triangular regions left at many vertices along the "spine" of the skeleton. (These triangular regions are clearly visible in Figure 1.) The number of tracks generated exceeded the number observed for the Zamboni decomposition by 20% to 50%. These small triangular regions are easily detected and can be incorporated within the tracks of the larger regions with simple heuristics, leading to a reduction in the number of tracks down to the values observed for the Zamboni decomposition. Results for the Zamboni decomposition are uniformly within 7% of the lower bound on all typical polygons, unless the ratio of turn radius to footprint is increased beyond 5, in which case the unrealistically small number of turns in the lower bound leads to some larger percentages—up to 30%. By designing special polygonal shapes, we were able to evince some of the worst-case behavior, but even hand-designed shapes that resemble the theoretical worst-case polygons only evinced 80% excess. Since our lower bounds are unattainable except for a rectangle, the experimental results are very encouraging; they compare quite favorably with the guaranteed approximations that one might derive based on TSP approaches on a grid for the lawnmower versions—50% at best if using Christofides' (slow) matching approach and better at higher prices if using some of the approaches recently described by Arora [5].

8. Conclusions

We have introduced a geometric problem with a number of important applications—a problem we termed the ice-rink problem by association with its simplest version. This problem is related to a number of other geometric problems of importance in manufacturing and is, naturally, NP-hard. We have described two successful heuristic approaches to the problem and given bounds on the worst-case behavior of one of these heuristics. Experimental work supports our contention that our heuristics return very good solutions that remain with a few percent of the optimum. Further work of experimental nature includes (i) incorporating probability distributions (for the location of the target) and (ii) adding cost functions that reflect terrain, fuel consumption, and other important real-world constraints.

References

[1] Arkin, E.M., S.P. Fekete, and J.S.B. Mitchell [1997], "Approximation algorithms for lawnmowing and milling," U. Köln Technical Report ZPR97-255.

[2] Arkin, E.M., S.P. Fekete, and J.S.B. Mitchell [1993], "The lawnmower problem," Proc. 5th Can. Conf. Comput. Geom., 461–466.

[3] Arkin, E.M., and R. Hassin [1994], "Approximation algorithms for the geometric covering salesman problem," *Discr. Appl. Math.* **55**, 197–218.

[4] Arkin, E.M., J.S.B. Mitchell, and C.D. Piatko [1991], "Bicriteria shortest path problems in the plane," Proc. 3rd Can. Conf. Comput. Geom., 153–156.

[5] Arora, S. [1996], "Polynomial-time approximation schemes for Euclidean TSP and other geometric problems," Proc. 37th IEEE Symp. Foundations Comput. Sci., 2–12.

[6] Bar-Eli, E., P. Berman, A. Fiat, and P. Yan [1992], "On-line navigation in a room," Proc. 3rd ACM-SIAM Symp. Disr. Algs., 237–249.

[7] Baeza-Yates, R., J. Culberson, and G. Rawlins [1993], "Searching in the plane," *Inf. and Comput.* **106**, 234–252.

[8] Collins, M.J., and B.M.E. Moret [1997], "Improved lower bounds for the link length of rectilinear spanning paths in grids," preprint available from the authors.

[9] Deng, X., T. Kameda, and C. Papadimitriou [1991], "How to learn in an unknown environment: the rectilinear case," Proc. 32nd IEEE Conf. Foundations of Comput. Sci., 298–303.

[10] Doddi, S., Marathe, M.V., Mirzaian, A., Moret, B.M.E., and Zhou, B., "Map labeling and its generalizations," Proc. 8th ACM-SIAM Symp. Discr. Algs., 148–157.

[11] Held, M. [1991]. *On the Computational Geoemtry of Pocket Machining.* Lecture Notes in Comput. Sci. **500**.

[12] Kleinberg, J. [1994], "On-line search in a simple polygon," Proc. 5th ACM-SIAM Symp. Discr. Algs., 8–15.

[13] Kranakis, E., D. Krizanc, and L. Meertens [1994], "Link length of rectilinear Hamiltonian tours on grids," *Ars Combinatoria* **38**, 177.

[14] Mitchell, J.S.B. [1997], "Geometric shortest paths and network optimization," preprint available from the author at `www.ams.sunsysb.edu/~jsbm/`.

[15] Mitchell, J.S.B., C.D. Piatko, and E.M. Arkin [1992], "Computing a shortest k-link path in a polygon," Proc. 33rd IEEE Symp. Foundations Comput. Sci., 573–582.

[16] Ntafos, S. [1992] "Watchman routes under limited visibility," *Comput. Geom.—Theory and Appl.* **1**, 149–170.

[17] Suzuki, I., and M. Yamashita [1992], "Searching for a mobile intruder in a polygonal region," *SIAM J. Comput.* **21**, 863–888.