Open Problems from CCCG'99*

Erik D. Demaine† Joseph O'Rourke‡

The following is a list of the problems presented on August 16, 1999 at the open-problem session during the 11th Canadian Conference on Computational Geometry.

Movable Square Blocks
Joseph O'Rourke, Smith College, orourke@cs.smith.edu

A robot and several movable unit-square block obstacles are confined to integral coordinates in a rectangle with integral length and width. A robot starts at the lower-left corner, and its goal is to occupy the upper-right corner. The robot may move horizontally or vertically in discrete unit steps, pushing any number of blocks in front of it. Is there a polynomial-time algorithm for deciding whether there exists a sequence of moves that enables the robot to reach the goal position?

This problem was first posed in [DO92]. That paper establishes two results: (1) the problem is polynomial if the path is restricted to be x- and y-monotone; and (2) the problem is NP-complete if some blocks may be glued to the plane. Subsequently, this second result was strengthened to PSPACE-hard by D. Bremner and T. Shermer [BOS94]. Closely related is the result that the block-pushing game *Sokoban* is PSPACE-hard [C98]; note that Sokoban also allows blocks to be glued to the plane. A recent result on another related problem is [DDO00].

[BOS94] David Bremner, Joseph O'Rourke, and Thomas Shermer, "Motion planning amidst movable square blocks is PSPACE complete," Draft, June 1994.

[C98] Joseph Culberson, "Sokoban is PSPACE-complete," in *Proceedings of the International Conference Fun with Algorithms*, Isola d'Elba, Italy, June 1998, pages 65–76.

[DDO00] Erik D. Demaine and Martin L. Demaine and Joseph O'Rourke, "PushPush is NP-hard in 2D," Technical Report 065, Smith College, Northampton, MA, January 2000. `http://www.arXiv.org/abs/cs.CG/0001019`

[DO92] Arundhati Dhagat and Joseph O'Rourke, "Motion planning amidst movable square blocks," in *Proceedings of the 4th Canadian Conference on Computational Geometry*, 1992, pages 188–191.

*This is a minor revision of the version that appears in the electronic proceedings of CCCG'99.

†Department of Computer Science, University of Waterloo, Waterloo, Ontario N2L 3G1, Canada. `eddemaine@uwaterloo.ca`.

‡Department of Computer Science, Smith College, Northampton, MA 01063, USA. `orourke@cs.smith.edu`. Supported by NSF Grant CCR-9731804.

Semi-rational Rectangular Tilings
Evangelos Kranakis, Carleton University, kranakis@scs.carleton.ca

The following two problems were solved positively by Eduardo Rivera-Campo and the solution was described during Evangelos's talk on August 18, 1999.

A rectangle is called *semi-rational* if one of its sides has a rational length. Given a simple orthogonal polygon (without holes) that can be tiled by semi-rational rectangles, must there be an edge of the polygon whose length is rational? Given a simple orthogonal polygon that can be tiled by $1 \times p$ and $p \times 1$ rectangles, must there be an edge of the polygon whose length is divisible by p?

Multi-label Map Labeling
Binhai Zhu, City University of Hong Kong, bhz@cs.cityu.edu.hk

Multi-label map labeling is a problem we encounter several times daily; for example, in the weather report on T.V., we need to put two labels on each city, its name and its temperature. Theoretically, almost nothing is known about this problem. In the particular problem posed here, we make the following assumptions:

- each city is a point,
- each city must be labeled by a pair of axis-parallel squares,
- each label must be placed so that its corresponding city is on the square's boundary (*sliding labels*),
- all labels are of the same size, and
- labels cannot overlap.

The goal is to find such a label placement that maximizes the label size. What is the complexity of this problem? In particular, is the problem NP-hard?

The only known result is a factor-4 approximation algorithm [ZP99]. Recently the factor was improved to 3 (Binhai Zhu, personal communication). Also shown in [ZP99] is a factor-2 approximation algorithm for labeling by pairs of disks (instead of axis-parallel squares). The complexity of this map-labeling problem is also open.

[**ZP99**] Binhai Zhu and C. K. Poon, "Efficient approximation algorithms for multi-label map labeling," in *Proceedings of the 10th Annual International Symposium on Algorithms and Computation*, Chennai, India, December 1999, pages 143–152.

Convexifying Polygons in 3-D
Godfried Toussaint, McGill University, godfried@cs.mcgill.ca

Godfried elaborated on three problems posed in his CCCG'99 paper [T99a] about applying operations to a simple polygon in 2 or 3 dimensions to *convexify* it, that is, make it convex.

Begin with a simple polygon in the plane. A *flip* takes a pocket of the convex hull (i.e., a connected region interior to the convex hull and exterior to the polygon) and rotates it 180° through 3-D about the incident convex-hull edge. It is known that a finite number of flips suffice to convexify a polygon, although the number cannot be bounded as a function of the number of vertices [G95, T99a].

1. A *flipturn* takes a pocket of the convex hull, detaches it from the rest of the polygon, rotates it 180° around the midpoint of the convex-hull edge, and re-attaches it to the polygon. How many flipturns are required to convexify a polygon? Joss and Shannon [G95] proved that at most $(n-1)!$ flipturns can be made before a polygon with n vertices becomes convex. They conjecture that $n^2/4$ flipturns always suffice.

   Recently, Ahn et al. [ABC+99] proved that at most $n(n-3)/2$ flipturns can be made before the polygon becomes convex. Biedl [B00] proved that a particular family of polygons has a (bad) sequence of at least $(n-2)^2/4$ flipturns before convexity is reached. Remaining open problems include closing the gap between $\sim n^2/4$ and $\sim n^2/2$ for a poor sequence of flipturns, and whether there are polygons whose shortest flipturn sequence has length $\omega(n)$.

2. A *deflation* is the reverse of a flip operation, that is, a rotation of a subchain of the polygon through 3-D that keeps the polygon simple and causes the subchain to become a pocket of the convex hull. Can a polygon be deflated more than a finite number of times?

The third problem is about polygons in 3-D:

3. There are five known classes of unknotted hexagons in 3-D that are *locked* in the sense that it cannot be reconfigured into a convex polygon while preserving the link lengths and without crossing the links [CJ98, T99b]. Prove that this exhausts all cases, i.e., that the configuration space of every unknotted hexagon has at most five connected components, or find another one.

[ABC+99] Hee-Kap Ahn, Prosenjit Bose, Jurek Czyzowicz, Nicholas Hanusse, Evangelos Kranakis, and Pat Morin, "Flipping your Lid," Manuscript, December 1999.

[B00] Therese Biedl, Technical Report CS-2000-04, Department of Computer Science, University of Waterloo, January 2000.

[CJ98] Jason Cantarella and Heather Johnston, "Nontrivial embeddings of polygonal intervals and unknots in 3-space," *Journal of Knot Theory and Its Ramifications*, volume 7, number 8, 1998, pages 1027–1039.

[G95] Branko Grünbaum, "How to convexify a polygon," *Geombinatorics*, volume 5, 1995, pages 24–30.

[T99a] Godfried Toussaint, "The Erdős-Nagy Theorem and its ramifications," in *Proceedings of the 11th Canadian Conference on Computational Geometry*, Vancouver, Canada, August 1999. `http://www.cs.ubc.ca/conferences/CCCG/elec_proc/fp19.ps.gz`

[T99b] Godfried Toussaint, "A new class of stuck unknots in Pol_6," Technical Report SOCS-99.1, School of Computer Science, McGill University, April 10, 1999. `http://cgm.cs.mcgill.ca/~godfried/publications/unknot.ps.gz`

Large Independent Sets in Delaunay Triangulations

Therese Biedl, University of Waterloo, biedl@uwaterloo.ca

An *independent set* in a graph is a set of vertices no two of which are adjacent to each other. The problem is to find a "big" independent set in a Delaunay triangulation "quickly." This is motivated by some work by Snoeyink and van Kreveld [SvK97].

To be more precise: It is known that, in any planar graph with n vertices, there exists an independent set I with $|I| \geq n/4$. This result follows from the 4-color theorem. However, while such a set can be found in $O(n^2)$ time, the algorithm is not very practical because of a large asymptotic constant. Despite many attempts, there is no linear-time algorithm to find $n/4$ independent vertices in a planar graph. Baker [B94] gave an approximation algorithm for independent sets that achieves $1-1/(k+1)$ times the optimal solution in $O(k8^k n)$ time. Thus we can get arbitrarily close to $n/4$ in linear time, but at the price of a high constant. The question is now whether the problem becomes any easier if we are looking at Delaunay triangulations rather than arbitrary planar graphs.

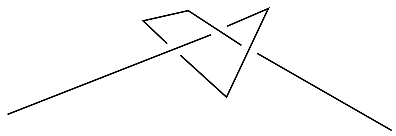
[B94] Brenda S. Baker, "Approximation algorithms for NP-complete problems on planar graphs," *Journal of the Association for Computing Machinery*, volume 41, number 1, 1994, pages 153–180.

[SvK97] Jack Snoeyink and Marc van Kreveld, "Linear-time reconstruction of Delaunay triangulations with applications," in *Proceedings of the 5th European Symposium on Algorithms*, LNCS 1284, Graz, Austria, September 1997, pages 459-471.

Cutting 3-D Linkages

Anna Lubiw, University of Waterloo, alubiw@uwaterloo.ca

The 5-link chain linkage shown below [CJ98, BDD+99] is known to be *locked* in the sense that it cannot be reconfigured into a straight line while preserving the link lengths and without crossing the links (provided the two end links are sufficiently long). However, if we cut any joint and disconnect the linkage into two pieces, the pieces can be straightened and rejoined at their "loose ends," thereby straightening the linkage.



This idea gives rise to the following problem. Let $f(n)$ denote the number of joints required to be cut in order to straighten the "worst" n-link chain in 3-D. Certainly, $f(n) < n$. Tying together several copies of the linkage above shows that $f(n) > n/4$. The problem is to close this gap by proving tighter upper or lower bounds on $f(n)$.

This problem is motivated by protein folding in molecular biology (Ming Li, personal communication). Proteins can be roughly modeled by chain linkages which are required to reconfigure into various positions. However, it seems that nature "cheats" by allowing joints of the chain to be cut temporarily during the motion.

[**BDD$^+$99**] T. Biedl, E. Demaine, M. Demaine, S. Lazard, A. Lubiw, J. O'Rourke, M. Overmars, S. Robbins, I. Streinu, G. Toussaint, and S. Whitesides, "Locked and Unlocked Polygonal Chains in 3D," Technical report 060, Smith College, October 1999. `http://www.arXiv.org/abs/cs.CG/9910009`

[**CJ98**] Jason Cantarella and Heather Johnston, "Nontrivial embeddings of polygonal intervals and unknots in 3-space," *Journal of Knot Theory and Its Ramifications*, 7(8):1027–1039, 1998.

HeapHull?

Jack Snoeyink, University of North Carolina, snoeyink@cs.ubc.ca

Is there a 2-dimensional convex-hull algorithm that operates on the same fundamental principle as Heapsort, and runs in $O(n \log n)$ time?

All of the notable sorting algorithms have their 2-D convex hull relatives, except for Heapsort. (Quicksort / quickhull, mergesort / divide and conquer, insertion sort / incremental construction, selection sort / Jarvis march; bubble sort isn't notable). Andrea Mantler has implemented one possibility based on a kinetic heap, but its time bound is $O(n \log^2 n)$. See `http://www.cs.ubc.ca/spider/mantler`.

Long Paths in Segment Endpoint Visibility Graphs

Prosenjit Bose, Carleton University, jit@scs.carleton.ca

Given a set of disjoint line segments in the plane, the *segment endpoint visibility graph* (see, e.g., [OR94]) has a vertex for each endpoint of a segment, and connects two vertices by an edge precisely if either

1. the corresponding endpoints are visible from each other in the sense that the open line segment connecting them is disjoint from the segments (a proper *visibility edge*), or
2. the endpoints are connected by a segment (a *segment edge*).

Mirzaian conjectured in 1992 that segment endpoint visibility graphs are Hamiltonian, and subsequently this has been established for a few classes of segments [M92, OR94]. More generally, how short can the longest cycle be in a segment endpoint visibility graph? How tight can we bound this length? One can prove using triangulation that there is always a cycle of length $\Omega(\sqrt{n})$ where n is the number of vertices (endpoints). Is there always a cycle of length $\Omega(n)$? Mirzaian's conjecture is that there is always a cycle of length *exactly* n.

An *alternating path* is a path in which every even-numbered edge is a segment and every odd-numbered edge is a visibility edge (or vice versa). How short can the longest alternating path be? **Conjecture**: There is always a path of length $\Omega(\log n)$.

[**M92**] Andranik Mirzaian, "Hamiltonian triangulations and circumscribing polygons of disjoint line segments," *Computational Geometry: Theory and Applications*, volume 2, number 1, 1992, pages 15–30.

[**OR94**] Joseph O'Rourke and Jennifer Rippel, "Two segment classes with Hamiltonian visibility graphs," *Computational Geometry: Theory and Applications*, volume 4, 1994, pages 209–218.