Open Problems from CCCG 2012

Joseph S. B. Mitchell*

On Wednesday afternoon, August 8, 2012, we held an open problem session at the *24th Canadian Conference on Computational Geometry*, in Charlottetown, Prince Edward Island, Canada. The following is a description of the problems presented and discussed, as scribed by Joe Mitchell, with follow-up comments and details written by the problem posers.

Zippered Volume
Anna Lubiw
University of Waterloo
alubiw@math.uwaterloo.ca

> Given a zipper of length 1, find a simply connected 2D shape, S, with perimeter 2 such that the 3D body obtained by "zipping up" the boundary of S has maximum volume. The problem is also interesting when S is restricted to be a polygon. See Figure 1. For more information, see [16].

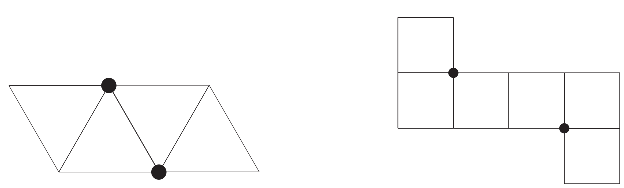

Figure 1: These unfoldings show that with a zipper of length 1 we can make a tetrahedron of side length $\frac{1}{3}$ and volume 44×10^{-4}, or a cube with side length $\frac{1}{7}$ and volume 29×10^{-4}. The start and end of the zipper are marked with dots, and the two sides of the zipper travel in opposite directions around the perimeter from the start to the end.

> Many people worked on this problem during the conference, including (but not limited to) Sarah Cannon, Jean-Lou De Carufel, Thomas Hackl, Stefan Huber, Denis Khromov, Matias Korman, Joe Mitchell, Vinayak Pathak, Diane Souvaine, Selim Tawfik, Ryuhei Uehara, Hamideh Vosoughpour.
>
> Denis Khromov suggested focusing on the final shape of the zipper in 3D, which leads to the problem of finding a curve C of length 1 that maximizes the volume of the convex hull of C. It turns out that this problem has a long history. For a closed curve in 2D, it is the isoperimetric problem (see the wikipedia page) and the solution is a circle. For an open curve in 2D, the solution is a semicircle. For an open curve in 3D this is problem A28 in Croft, Falconer, and Guy [5]. From the exposition there and in the paper of Tilli [23] it seems that the problem is solved for curves that do not cross any plane more than 3 times. In this case, the optimum solution is a circular helix, $x = sin(t)$, $y = cos(t)$, $z = t/\sqrt{2}$, as t goes from 0 to 2π, which gives volume 102×10^{-4}. See Figure 2.

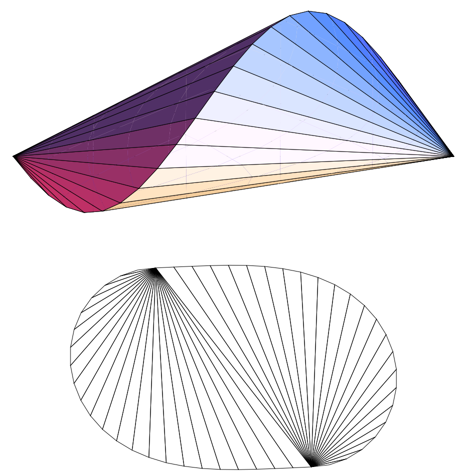

Figure 2: Among a large class of curves, the maximum volume of the convex hull, 102×10^{-4}, is achieved by a helix, with convex hull and unfolding as shown.

> Some people at CCCG found (suboptimal) solutions based on cones. Sarah Cannon, Diane Souvaine and I found one with a volume of 84×10^{-4}, where the curve consists of two semicircles lying in orthogonal planes. The 3D body consists of two half-cones where each half-cone has a semicircular base and an apex above one endpoint of the semicircle. See Figure 3.

Characterize Output of Poisson-Disk Process
Scott Mitchell
Sandia National Lab
samitch@sandia.gov

> (This might be considered a problem in spatial

*Stony Brook University, Stony Brook, NY 11794-3600, USA, Joseph.Mitchell@stonybrook.edu. Partially supported by grants from the National Science Foundation (CCF-1018388) and the Binational Science Foundation (BSF 2010074).

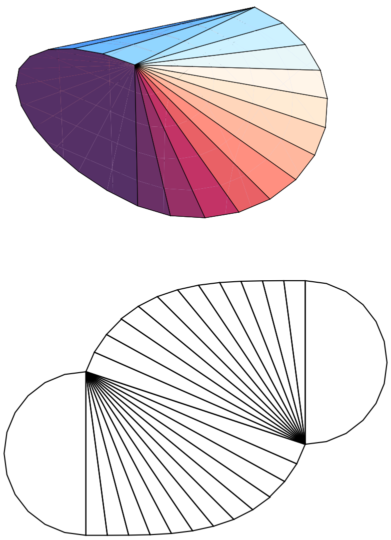

Figure 3: A volume of 84×10^{-4} is achieved by conjoined half cones.

statistics, but there are ties to Delaunay refinement and sphere packings.) Maximal Poisson-disk sampling (MPS) is a particular statistical process for generating a point cloud. The location for the next point is chosen uniformly by area at random. A point has an empty disk of radius r around it; if a new point falls into a prior point's disk, it is rejected and not added to the sample. The process continues until the sampling is maximal: the entire domain is covered by samples' disks and there is no room for another sample. Let the domain be a two-dimensional square with periodic (toroidal) boundary conditions, so there are no domain boundary issues to consider. A math definition appears in Ebeida et al. [9].

I am aware of no analytic description of what the correct output of MPS is supposed to be. I haven't even seen an experimental characterization! As such, currently for an algorithm to be correct, it must be step-by-step equivalent to the statistical process. For an example algorithm like this, see again [9]. A characterization of the output is important because it would enable the design of more efficient algorithms. A metaphor is that bubble-sort is a process, but the characterization of its output as "sorted order" allows the discovery of e.g. quicksort to generate "sorted order" more efficiently.

The computer graphics community typically measures the output of MPS by generating Fourier transform pictures of the output. See "Point Set Analysis" [17], for software and paper references for a standard way of generating these pictures.

My understanding of PSA follows. The vectors of distances between all pairs of points are calculated. The Fourier transform of the distance vectors are taken and displayed, and a picture with oscillating dark and light rings is expected. Integrating this transform over concentric circles produces a one-dimensional graph by increasing radius. (A nuance is how to bin distances to generate smooth pictures.) Figure 4 top shows the kinds of pictures the Graphics community expects to see for MPS.

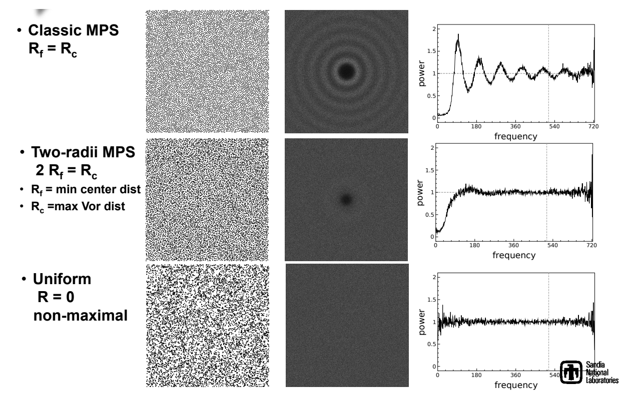

Figure 4: Point clouds visualized using PSA. Top is standard MPS, and middle two-radii MPS from CCCG 2012. The bottom is a uniform random point cloud without inhibition disks, using about the same number of points.

Subproblem A: Can you characterize the PSA pictures for MPS, especially Figure 4 top right? What is the mean location and height of the peaks? What is their standard deviation? Is the distribution around the mean normal? (Recall MPS is a random process.) Perhaps an experimental characterization is an easier place to start than an analytic characterization.

Subproblem B: Is some variant of MPS better than standard MPS for texture synthesis graphics applications? At CCCG 2012 I presented a paper "Variable Radii MPS." The two-radii MPS variant generates a spectrum with less oscillations; see Figure 4 middle. We suspect, but don't know for sure, if this is better for applications.

MPS produces a sphere packing, halve the disk radii r then the disks do not overlap. This is a well-spaced point set. Delaunay refinement also produces a well-spaced point set. Sometimes the PSA pictures of the output of Delaunay refinement look similar to MPS, sometimes not, depending on the target edge length, angle threshold, the use of off-centers, etc.; see Figure 5.

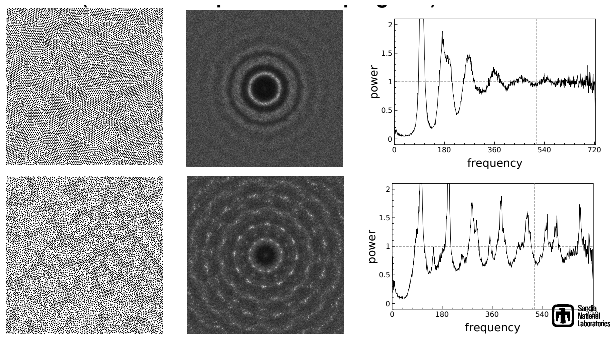

Figure 5: Delaunay refinement (Triangle) point clouds from particular choices of target edge lengths and angles, visualized using the PSA tool. Top left we see patches of hexagonal packings, and bottom left we see circular patterns of jumps in the point spacing. In the Fourier transform, middle column, in the top the rings are more pronounced than form MPS; in the bottom we see bright spots which indicate preferential directions, meaning nearby points are more dense in certain directions than others. In the radial average, right, on both the top and bottom we see accentuated spikes.

Subproblem C: characterize the PSA pictures (Fourier spectrum) of the output of Delaunay refinement and its variants.

In computational geometry we often measure point sets by the angles and edge length histograms in a Delaunay triangulation of the points. These histograms are different for MPS point clouds than for Delaunay refinement output; see Figure 6.

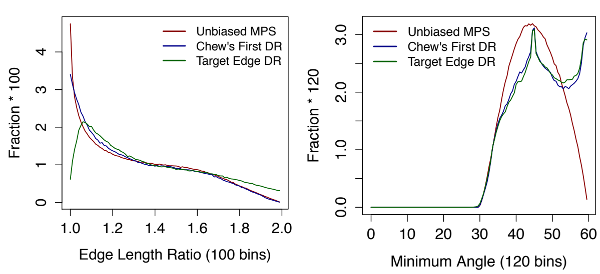

Figure 6: Computational Geometry measures of point clouds. Edge length and angle histograms of DR and MPS output. The minimum angle is the smallest angle of each triangle. The edge length ratio is the ratio of the length of Delaunay edges to the disc radius (MPS) or maximum Delaunay circumradius (DR). In both MPS and DR, the theories guarantee $r < |e| < 2r$.

Subproblem D: Characterize MPS output using computational geometry measures of Delaunay triangulation edge lengths and angle distributions.

Bonus subproblem E: do these problems for dimensions other than 2. Three to five dimensions have some graphics applications.

Bonus subproblem F: characterize the effect of the domain boundary, for non-periodic domains.

Partners: Alexander Rand, Mohamed Ebeida, John Owens, Anjul Patney, Andrew Davidson, Chandrajit Bajaj.

Hiding a Cycle
Paz Carmi
Ben Gurion University
carmip@cs.bgu.ac.il

Pat Morin had posed at CCCG'2007 the following question: Is it possible that a polygonal cycle C in 3D, with each edge axis-parallel, can project orthogonally onto each of the three coordinate planes in such a way that each projection does not have a cycle?

A solution to the problem appears on the cover of the book [24]. However, the three projections in this figure are (rectilinear) *trees*. Is it possible that all three projections of C are *paths*?

Divide and Conquer
Jérémy Barbay
Universidad de Chile
jbarbay@dcc.uchile.cl

Adaptive Divide and Conquer: For several problems, divide and conquer algorithms of complexity $O(n \log n)$ and $O(nk)$, for input size n and some additional parameter k, yield algorithms of complexity $O(n(1+\lg k))$, or even better, $O(n(1+\mathcal{H}(n_1,\ldots,n_k)))$:

- **Selecting** k **elements** of respective ranks $(r_1,\ldots,r_k)$ in an unsorted array of n elements can obviously be done by sorting the n elements in $O(n \lg n)$ comparisons. It can also be performed in $O(nk)$ comparisons by using k times the median of median quick select algorithm. Yet in 1981 Dobkin et al. [7] showed that those k ranks can be computed in $O(n(1+\mathcal{H}(r_1, r_2-r_1,\ldots,r_k-r_{k-1}))) \subset O(n(1+\lg k))$ comparisons and overall time, and in 2006 Kaligosi et al. [13] showed that those k ranks can be computed in $n(1+\mathcal{H}(r_1,r_2-r_1,\ldots,r_k-r_{k-1}))+o(n(1+\mathcal{H}(r_1,r_2-r_1,\ldots,r_k-r_{k-1})))+O(n)$ comparisons and $O(n(1+\mathcal{H}(r_1,r_2-r_1,\ldots,r_k-r_{k-1})))$ overall time. This result is *input order oblivious instance optimal*, in the sense that no algorithm can perform better in the worst (and average) input order, for any set of values.

- In 1972, Graham [11] showed a reduction of the computational complexity of the **convex hull** to sorting, yielding a complexity of $O(n \log n)$, which is optimal in the worst case over instances composed of n points. Yet in 1973, Jarvis [12] showed that this was not optimal on instances where the number h of points in the convex hull is smaller than $\log n$, via an algorithm running in time within $O(hn) \subset O(n^2)$. It is not until 1986 that Kirkpatrick and Seidel [15] solved the paradox through an algorithm running in time within $O(n(1 + \log h))$, which analysis Afshani et al. [1] later improved to $O(n(1 + \mathcal{H}(n_1, \dots, n_h)))$, where $\mathcal{H}(n_1, \dots, n_h) \equiv \sum_{i=1}^{h} \frac{n_i}{n} \log \frac{n}{n_i} \leq \log_2 h$ is the minimal entropy of a certificate of the instance.
- In 1983 Reif [19] described an algorithm computing the **minimal cut between two vertices** s and t of a planar graph over n vertices in time $O(n \log^2 n)$, using the fact that the minimal cut corresponds to a cycle of minimal total weight in the dual of the planar graph, which is intersected at most once by the minimal path between s^* and t^*. Another algorithm was known to perform in linear time when s and t share a face. In 2011, Kaplan et al. [14] generalized this result to yield an algorithm computing the minimal cut in $O(n(1 + \log p))$ operations, where p is the minimum number of edges crossed by a curve joining s to t, or equivalently the minumum number of edges from the face s^* to the face t^* in the dual of the planar graph.
- In 1952, Huffman showed that a **prefix free code of minimal redundancy** for n weighted symbols can be computed in $O(n \lg n)$ algebraic operations (In 1976, van Leeuwen showed that an equivalent code can be obtained in linear time when the weights are sorted.). In 2006 Belal et al. [3] claimed an algorithm computing a code of equivalent redundancy in $O(nk)$ algebraic operations, where k is the number of distinct code-lengths in a code of minimal redundancy.
- In an undirected planar graph of n vertices, the **maximum flow** between two vertices s and t can be computed in time $O(n \lg n)$ via $O(n)$ augmenting paths, which can be improved to $O(n(1 + \lg k))$ time when k faces separate s and t. This result is optimal in the worst case over instances for fixed values of n and k.

We ask the following questions:

1. For which other problems are there at once an algorithm working in time $O(n \lg n)$ and an algorithm working in $O(nk)$, for some parameter k, but no known algorithm running in $O(n(1 + \lg k))$?
2. For which other problems is there an algorithm running in $O(n(1 + \lg k))$, and a potential for an algorithm working in time $O(n(1 + \mathcal{H}(n_1, \dots, n_k)))$, where $(n_1, \dots, n_k)$ is a vector describing the difficulty of the instance in a finer way than k and $\mathcal{H}(n_1, \dots, n_k)$ is its entropy?
3. For the problems where there is an algorithm running in time $O(n(1 + \mathcal{H}(n_1, \dots, n_k)))$, can this be improved to $n(1+\mathcal{H}(r_1, r_2 - r_1, \dots, r_k - r_{k-1})) + o(n(1+\mathcal{H}(r_1, r_2 - r_1, \dots, r_k - r_{k-1}))) + O(n)$ comparisons and $O(n(1 + \mathcal{H}(r_1, r_2 - r_1, \dots, r_k - r_{k-1})))$ overall time, as Kaligosi et al. did for the multi select problem?
4. Can we identify a general principle at work for those problems (e.g. Input Order Oblivious Instance Optimal Complexity?), or are there counter examples, for example in the form of a problem for which there is an algorithm running in time $O(n \lg n)$, an algorithm taking advantage of particular cases running in $O(nk)$, but provably no algorithm running in $o(n \min\{\lg n, k\})$ in the worst case over all instances of fixed size n and fixed parameter value k?

Minimum Interference Networks
Pat Morin
Carleton University
morin@scs.carleton.ca

Given a set S of n points in $\mathbb{R}^d$, the goal is to construct a connected network on S in order to minimize the *(maximum receiver-centric) interference*, which is the maximum depth in the set of disks, centered on each $p_i \in S$, of radius equal to the length of the longest edge incident to p_i.

In two and higher dimensions, achieving a better than 5/4-approximation is NP-hard, as was shown by Buchin [4]. In 1D, the problem is not known to be NP-hard, yet the only result is a polynomial-time $O(n^{1/4})$-approximation [20]. Thus, in any dimension, the following question is open: Does there exist a polynomial time $o(n^{1/4})$-approximation algorithm for constructing a minimum interference network?

Another problem in this area involves bounding the minimum-interference network of a point set by the interference of its minimum spanning tree. This is captured by the following conjecture: If the minimum spanning tree of a point set, S, has interference k, then there exists a connected network on S with interference $O(\sqrt{k})$. If proven, this would resolve a question, from [6], on constructing minimum-interference networks for random point sets.

Generalized MST: 2-GMST
Bob Fraser
University of Waterloo
r3fraser@uwaterloo.ca

In the generalized (or "one-of-a-set" or "group") MST, we are given a collection of n finite sets of points, $S_1, \ldots, S_n$, and the goal is to compute a minimum-weight spanning tree that visits at least one point of each set S_i. Pop [18, Theorem 4.3] described a 2δ-approximate algorithm for generalized MST, where δ is the maximum cardinality of any imprecise vertex. The algorithm solves the linear programming relaxation of the integer programming formulation of the problem, and then chooses a spanning tree from the points in the solution. This result follows the approximation framework used by Slavík [21, 22] to establish approximation algorithms for the Generalized Travelling Salesman Problem and the Group Steiner Tree problem with approximation factors of $3\delta/2$ and 2δ, respectively.

We ask here about the 2-GMST in the Euclidean plane, in which each S_i is a pair of points; in fact, we are interested in the special case in which each pair is vertical (i.e., each set S_i consists of two points having the same x-coordinate). For the case that $|S_i| = 2$, for all i, a 4-approximation is immediate from the general 2δ-approximation algorithm of Pop mentioned above. Can geometry be exploited to do better?

Note that the generalized problem is APX-hard (Dror and Orlin[8]), and the restricted version described here (2-GMST) is NP-hard [10, §10.4] (the proof of hardness also rules out the possibility of an FPTAS).

Two Problems
Pankaj Agarwal
Duke University
pankaj@cs.duke.edu

(a). *Forcing a vertex minimum of a terrain.* Given a triangulated terrain with (piecewise-linear) heigh function $h(x, y)$, and given a vertex v of the terrain, our goal is to compute a new terrain, $h'(x, y)$, such that v is a unique minimum of h' and h' has no critical points other than v. The objective function is to select h' to minimize $||h - h'||$ (or $(||h - h'||)^2$).

(b). *Separating 3D polytopes (a classic problem).* Given two convex polytopes, P_1 and P_2 in $\Re^3$, each with a Dobkin-Kirkpatrick hierarchy, our goal is to compute a separating plane, or report that none exists, for P_1 and P_2. Known methods yield time $O(\log m \log n)$, where m and n are the complexities of P_1 and P_2. Is it possible to improve the time bound to $O(\log m + \log n)$?

Redrawing a Triangulation
Csaba Tóth
University of Calgary
cdtoth@ucalgary.ca

Given a triangulation T of n points in the plane. Let ℓ be a line intersecting T, crossing edges $(e_1, e_2, \ldots, e_k)$, of T at points $(p_1, p_2, \ldots, p_k)$, in order along ℓ. Now, consider another line, L, with points $(q_1, q_2, \ldots, q_k)$, in order along L. Is it always possible to draw a triangulation T', equivalent (in the sense of graph isomorphism) to T, with L crossing the edges $(e'_1, e'_2, \ldots, e'_k)$ (the mappings of the e_i's) at the points $(q_1, q_2, \ldots, q_k)$?

Wavelets and the Golden Ratio
Braxton Carrigan
Auburn University
bac0004@auburn.edu

Given a triangulation in which every triangle is isosceles. The goal is to find a subtriangulation by adding new vertices along edges (while keeping the cell complex property) in order to preserve the ratio of side lengths (in Golden ratio).

Hamiltonian Tetrahedralization of a 3-Polytope
Joe Mitchell
Stony Brook University
Joseph.Mitchell@stonybrook.edu

I conjecture that every three-dimensional convex polytope has a tetrahedralization (without adding Steiner points) whose dual graph has a Hamiltonian path. If true, then every finite point set in 3D has a tetrahedralization (without Steiner points) that is Hamiltonian.

For related work, see Arkin et al. [2].

References

[1] P. Afshani, J. Barbay, and T. Chan. Instance-optimal geometric algorithms. In *Proc. 50th IEEE Symposium on Foundations of Computer Science (FOCS)*, 2009.

[2] E. M. Arkin, M. Held, J. S. B. Mitchell, and S. Skiena. Hamiltonian triangulations for fast rendering. *The Visual Computer*, 12(9):429–444, 1996.

[3] A. A. Belal and A. Elmasry. Distribution-sensitive construction of minimum-redundancy prefix codes. In B. Durand and W. Thomas, editors, *23rd International Symposium on Theoretical Aspects of Computer Science (STACS 2011)*, volume 3884 of *Lecture Notes in Computer Science*, pages 92–103. Springer, 2006.

[4] K. Buchin. Minimizing the maximum interference is hard. *CoRR*, abs/0802.2134, 2008.

[5] H. T. Croft, K. J. Falconer, and R. K. Guy. *Unsolved Problems in Geometry*, Springer-Verlag, 1991.

[6] L. Devroye and P. Morin. A note on interference in random point sets. In *Proc. 24th Canadian Conference on Computational Geometry (CCCG)*, pages 201–206, 2012.

[7] D. P. Dobkin and J. I. Munro. Optimal time minimal space selection algorithms. *Journal of the ACM*, 28(3):454–461, 1981.

[8] M. Dror and J. B. Orlin. Combinatorial Optimization with Explicit Delineation of the Ground Set by a Collection of Subsets, *SIAM Journal on Discrete Mathematics*, 21(4):1019–1034, 2008.

[9] M. S. Ebeida, A. A. Davidson, A. Patney, P. M. Knupp, S. A. Mitchell, and J. D. Owens. Efficient maximal poisson-disk sampling. *ACM Transactions on Graphics*, 30(4):49, 2011.

[10] R. Fraser. *Algorithms for Geometric Covering and Piercing Problems*. PhD thesis, University of Waterloo, 2012.

[11] R. L. Graham. An efficient algorithm for determining the convex hull of a finite planar set. *Information Processing Letters*, 1(4):132–133, 1972.

[12] R. Jarvis. On the identification of the convex hull of a finite set of points in the plane. *Information Processing Letters*, 2(1):18–21, 1973.

[13] K. Kaligosi, K. Mehlhorn, J. I. Munro, and P. Sanders. Towards optimal multiple selection. In *Automata, Languages and Programming, 32nd International Colloquium, ICALP 2005, Lisbon, Portugal, July 11-15, 2005, Proceedings*, volume 3580 of *Lecture Notes in Computer Science*, pages 103–114. Springer, 2005.

[14] H. Kaplan and Y. Nussbaum. Minimum s-t cut in undirected planar graphs when the source and the sink are close. In T. Schwentick and C. Dürr, editors, *28th International Symposium on Theoretical Aspects of Computer Science (STACS 2011)*, volume 9 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 117–128, Dagstuhl, Germany, 2011. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

[15] D. G. Kirkpatrick and R. Seidel. The ultimate planar convex hull algorithm? *SIAM Journal on Computing*, 1986. 15(1):287–299.

[16] A. Lubiw, E. D. Demaine, M. L. Demaine, A. Shallit, and J. Shallit. Zipper unfoldings of polyhedral complexes. In *Proc. 22nd Annual Canadian Conference on Computational Geometry (CCCG)*, pages 219–222, 2010.

[17] Point Set Analysis, http://code.google.com/p/psa/.

[18] P. Pop. *The Generalized Minimum Spanning Tree Problem*. PhD thesis, University of Twente, 2002.

[19] J. Reif. Minimum s-t cut of a planar undirected network in $o(n \log n)$ time. In S. Even and O. Kariv, editors, *Automata, Languages and Programming*, volume 115 of *Lecture Notes in Computer Science*, pages 56–67. Springer Berlin / Heidelberg, 1981.

[20] P. von Rickenbach, S. Schmid, R. Wattenhofer, and A. Zollinger. A robust interference model for wireless ad-hoc networks. In *Proc. 19th International Parallel and Distributed Processing Symposium (IPDPS 2005)*. IEEE Computer Society, 2005.

[21] P. Slavík. The errand scheduling problem. Computer Science Technical Report 97-2, State University of New York at Buffalo, 1997.

[22] P. Slavík. *Approximation Algorithms For Set Cover And Related Problems*. PhD thesis, State University of New York at Buffalo, 1998.

[23] P. Tilli. *Transactions of the American Mathematical Society*, 2010. 362:4497–4509.

[24] P. Winkler. *Mathematical Mind Benders*. A K Peters/CRC Press, 2007.