Helmut Alt, Emo Welzl (editors)

Algorithmic Geometry

Dagstuhl-Seminar-Report; 4
8.10.1990 - 12.10.1990 (9041)

Das IBFI (Internationales Begegnungs- und Forschungszentrum für Informatik) ist eine gemeinnützige GmbH. Sie veranstaltet regelmäßig wissenschaftliche Seminare,welche nach Antrag der Tagungsleiter und Begutachtung durch das wissenschaftliche Direktorium mit persönlich eingeladenen Gästen durchgeführt werden.

Verantwortlich für das Programm:
Prof. Dr.-Ing. José Encarnaçao,
Prof. Dr. Winfried Görke,
Prof. Dr. Theo Härder,
Dr. Michael Laska,
Prof. Dr. Thomas Lengauer,
Prof. Ph. D. Walter Tichy,
Prof. Dr. Reinhard Wilhelm (wissenschaftlicher Direktor).

Gesellschafter: Universität des Saarlandes,
Universität Kaiserslautern,
Universität Karlsruhe,
Gesellschaft für Informatik e.V., Bonn

Träger: Die Bundesländer Saarland und Rheinland Pfalz.

Bezugsadresse: Geschäftsstelle Schloß Dagstuhl
Informatik, Bau 36
Universität des Saarlandes
W - 6600 Saarbrücken
Germany
Tel.: +49 -681 - 302 - 4396
Fax: +49 -681 - 302 - 4397
e-mail: dagstuhl@dag.uni-sb.de

INTERNATIONALES BEGEGNUNGS- UND FORSCHUNGSZENTRUM

FÜR INFORMATIK

SCHLOSS DAGSTUHL

Tagungsbericht

zum 1. Dagstuhl-Seminar über

Algorithmische Geometrie

8.Oktober bis 12.Oktober 1990

Report

of the 1st Dagstuhl-seminar on

Computational Geometry

October 8th - October 12th, 1990

The first Dagstuhl-seminar on Computational Geometry was organized by Helmut Alt and Emo Welzl (both FU Berlin). The 28 participants came from 10 countries, 7 of them came from North America and Israel.

20 lectures were given at the seminar, covering quite a number of topics in computational geometry. As was to be expected, a large percentage of the talks dealt with randomization in one way or the other, reflecting the current inclination of the community towards this field. Accordingly, a special discussion session on randomized algorithms was held, which in fact attracted most of the participants of the seminar.

Furthermore, two lectures dealt with the actual implementation of geometric algorithms, and two library/development systems were presented.

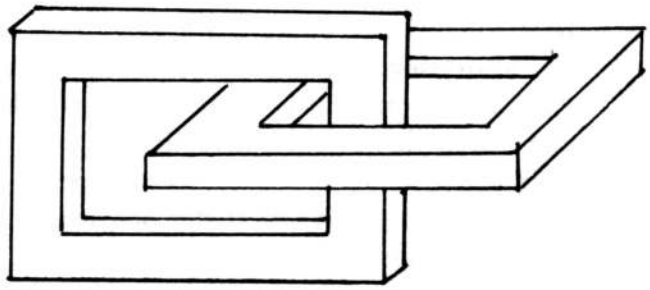
Other lectures dealt with problems concerning the similarity of two objects, e.g. matching points into regions, or measuring the (Hausdorff-) distance between polygons.

However, the most unforgettable event of the seminar was clearly the open problem session on monday evening, chaired by Ricky Pollack. It was there that Raimund Seidel declared the *zone theorem* an open problem, since the proof by Edelsbrunner, O'Rourke, Seidel (given e.g. in the book by Herbert Edelsbrunner) is incorrect. The excitement was considerable, and still rose when Jirka Matoušek pulled out of his bag a short manuscript entitled *A simple proof of weak zone theorem*! Micha Sharir also presented some ideas as to how to prove a weak zone theorem, and now there seems to be a new proof for the zone theorem by Herbert Edelsbrunner which proceeds along the same lines.

Participants: Helmut Alt, Freie Universität Berlin
Franz Aurenhammer, Freie Universität Berlin
Ken Clarkson, AT&T Bell Labs
Torben Hagerup, Universität des Saarlandes
John Hershberger, DEC Systems Research Center, Palo Alto
Klaus Hinrichs, Universität-GH Siegen
Giuseppe F. Italiano, Columbia University
Hermann Jung, Humboldt-Universität Berlin
Rolf Klein, Universität-GH Essen
Jiří Matoušek, Charles University, Prague
Kurt Mehlhorn, Universität des Saarlandes
Joe Mitchell, Cornell University
Heinrich Müller, Universität Freiburg
Stefan Näher, Universität des Saarlandes
Jürg Nievergelt, ETH Zürich
Hartmut Noltemeier, Universität Würzburg
Thomas Ottmann, Universität Freiburg
Mark Overmars, Utrecht University
János Pach, Hungarian Academy of Sciences and Courant Institute
Ricky Pollack, Courant Institute
Günter Rote, Technische Universität Graz
Jörg-R. Sack, Carleton University
Peter Schorn, ETH Zürich
Otfried Schwarzkopf, Freie Universität Berlin
Raimund Seidel, University of California, Berkeley
Micha Sharir, Tel Aviv University
Hubert Wagener, Technische Universität Berlin
Emo Welzl, Freie Universität Berlin

Output-Sensitive Hidden Surface Removal for Axis-Parallel Polyhedra

Mark Overmars
(joint work with Mark de Berg)
Dept. of Computer Science, Utrecht University



Let V be a set of axis-parallel polyhedra in 3-dimensional space. Given a point p we wish to compute which parts of the polyhedra are visible from p. The complexity of this *visibility map* can be as large as $\Omega(n^2)$, where n is the number of vertices of the polyhedra , and as small as $\mathcal{O}(1)$. The goal is to find an algorithm that computes the visibility map in an amount of time, depending on the complexity of the visibility map. Let k be this complexity. We give an algorithm that runs in time $\mathcal{O}((n+k)\log n)$. The method easily deals with polyhedra having holes, with cyclic overlap among the polyhedra, with perspective views, etc. The basic strategy behind the method is the following: In a first step those vertices of the polyhedra are computed that are visible. Next, starting at these visible vertices, we trace the connected components of the visibility map by using a kind of $2\frac{1}{2}$-dimensional ray shooting. The correctness of this approach follows from the fact that each connected component contains at least one visible vertex.

The method can be extended to sets of c-oriented polyhedra where the faces and edges are allowed to have c different orientations. When c is a constant, the algorithm runs in exactly the same time bound. A second extension concerns penetrating polyhedra, where the polyhedra are allowed to intersect in 3-space. Again, we can use the same strategy to solve the problem. The time bound though becomes a factor $\mathcal{O}(\log n)$ worse.

Problem Session on Randomized Algorithms

This problem session was chaired by Micha Sharir, who also presented his framework for the analysis of randomized algorithms. Furthermore, Ken Clarkson explained his simplified analysis of Incremental Randomized Construction. Its advantages are that no bounds on the size of k-sets are necessary, and that it enables us to compute the expected space and running time with a better constant factor. The key idea is to bound the expected number of conflict graph edges created in a fixed stage of the algorithm by backwards-analysis. The rest of the session was used to discuss some interesting open problems, as e.g.

- Finding tail-estimates for the time-complexity of randomized incremental construction
- Analyzing various heuristics to improve the random order in which objects are inserted, e.g. the *move-to-front*-heuristic in Emo Welzl's `miniDisk`-algorithm
- Dealing with situations where the output size is better than the worst-case, e.g. in Voronoi diagrams in three dimensions. Is there a good order for the objects, so that the output of the intermediate stages is not too big?
- Derandomizing incremental randomized algorithms. The Martingale approach by Torben Hagerup shows how to do this in polynomial time. Is there a better way?
- Finding a nice framework for a randomized analysis of dynamic algorithms, e.g. assuming that objects are inserted in random order, while objects to be deleted are drawn from the current set with uniform probability

How to walk a street

Rolf Klein
(partially joint work with Christian Icking)
University of Essen, Germany

A polygon with vertices s and g is called a *street* iff it is possible to move two points, p and q, on the two boundary chains connecting s with g in such a way that none of them ever backtracks and that the line segment $\overline{pq}$ is, at each time, fully contained in the polygon.

Given a street, we want to compute a path from s to g that is as short as possible, but without knowing the whole polygon in advance, only on the basis of local information comprised by the visibility polygons of the points visited so far.

It turns out that at each time there are at most two candidate vertices in the visibility polygon that could be visited by the actual shortest path. Furthermore, new information must come up as one proceeds to the line segment connecting the two candidates. We propose a strategy that locally minimizes the absolute detour, and show that the resulting path does not exceed in length K times the length of the shortest path from s to g, where K is a constant not depending on the street. Currently, we are able to show this result for $K = 2 + 2\pi$, but experiments lead to believe that it is true for $K < 2$. The case of a triangular street shows that K must be greater than $\sqrt{2}$.

Parallel Computation of Enclosing Triangles with Minimal Area

Hubert Wagener
Technische Universität Berlin

Given a convex polygon C, an enclosing triangle with minimal area can be found by an algorithm that generalizes the merging by binary search. A runtime of $\mathcal{O}(\frac{n}{p} + \log n)$ $\big(\mathcal{O}(\frac{n}{p} + \log\log n)\big)$ is obtained for p processors of a CREW-PRAM (CRCW-PRAM, resp.). Some other algorithms following the *merging-paradigm* (like rotating caliper-algorithms) can be parallelized in the same way.

An Application of Martingales to Computational Geometry

Torben Hagerup
Universität des Saarlandes

Using the theory of martingales, it is shown that a randomized parallel algorithm for constructing arrangements of n hyperplanes in d dimensions works in $\mathcal{O}(\log n)$ time with $\mathcal{O}(n^d/\log n)$ processors not only in the expected case, which was previously known, but in fact with high probability $(1 - 2^{-n^{\Omega(1)}})$. It is argued that martingales may be applicable to other situations in computational geometry where one wants to prove some random quantity to be sharply concentrated around its mean.

Planar Graphs and Order Types: Isotopy, Coordinate Complexity and Robustness

Ricky Pollack
Courant Institute, New York University

We survey the following results which show unusual differences between planar graphs and order types:

	Planar Graphs		**Order Types**
1g	Every planar graph with n vertices has a straight line drawing with the vertices on an $(n-1)\times(n-1)$ grid.	1o	Every order type of n points has a realization on a $2^{2^{cn}} \times 2^{2^{cn}}$ grid and a grid of this size may be required
2g	Two straight line drawings of the same maximal planar graph G can be continuously deformed one into the other so that we have a straight line drawing of G throughout the deformation. The realization space of the straight line drawing of a maximal planar graph is connected (In fact, it is contractible).	2o	The realization space of $\{p_1, \ldots, p_n\} \subset \mathbf{R}^d$, i.e. all points $(x_1^1, \ldots, x_d^1, \ldots, x_1^n, \ldots, x_d^n) \in \mathbf{R}^{nd}$ such that $\{(x_1^1, \ldots, x_d^1), \ldots, (x_1^n, \ldots, x_d^n)\}$ has the same order type as $\{p_1, \ldots, p_n\}$, can be homotopy equivalent to any semi-algebraic set.

1o shows that our approach to the problem of robustness is unreasonable.

A recent result of Malenkovic and Nachmann shows that a more reasonable approach is still unreasonable.

Geometric Applications of Parametric Search

Micha Sharir
Tel Aviv University and New York University

The talk describes Megiddo's parametric searching technique for finding a parameter r that optimizes a certain function $F(r)$. The technique runs an algorithm for computing $F(r)$ *generically* at the unknown desired value r^*, and resolves comparisons depending on r^* by off-line calls to the same algorithm. To attain efficient running time, the generic algorithm is run in parallel, thereby minimizing the number of off-line calls to the algorithm. The talk illustrates the technique on several geometric problems. The first problem is: Given n points in the plane and an integer $k \leq \binom{n}{2}$, find the k-th smallest distance between pairs of the points. Using Megiddo's technique, we obtain an $\mathcal{O}(n^{4/3} \text{polylog } n)$ randomized expected time algorithm. The second problem is: Given n points in the plane, find the smallest r such that the points can be covered by two disks of radius r. We present an $\mathcal{O}(n^2 \log^3 n)$ solution. We also mention the problems of covering n points by two strips of smallest width, and of finding the largest copy of a convex polygon that can be placed inside a given polygonal region.

A Workbench for Computation Geometry

Jörg-R. Sack
Carleton University, Ottawa

We describe the design and implementation of a workbench for computational geometry. We discuss issues arising from this implementation including comparisons of different algorithms (Tarjan, van Wyck triangulation versus other triangulation algorithms; finger trees, splay trees, randomized binary search trees, binary search trees etc. and their efficiency in different algorithms).

The workbench is not just a library of geometric operations and algorithms, but is designed as a geometrical computing environment, providing tools for: creating, editing and manipulating geometric objects, demonstrating and animating geometric algorithms; and most importantly for implementing new algorithms. In particular, automatic garbage collection, high-level debugging facilities and control mechanisms are provided.

Some geometric objects need to be created efficiently for testing timing of algorithms. We discuss algorithms for creation of Jordan sequences and simple polygons.

The XYZ program library for geometric computation

Jürg Nievergelt and Peter Shorn
ETH Zürich

The XYZ Geobench consists of several components: A class hierarchy for defining geometric object types and data structures; an arithmetic package for various types of integer and floating point arithmetic; geometric primitives, such as `WhichSide(Point, Line)`; and, finally, an expandable library of programs. In implementing algorithms for inclusion in the library we strive for perfection in regard to details: correctness and consistency in the presence of degenerate configurations and rounding errors. As an example we present the Closest Pair problem where our plane sweep algorithm carefully implemented in floating point arithmetic produces provably accurate results. Using the same primitives we give an accurate and practical algorithm for the All-Nearest-Neighbours problem in the L_∞-metric.

Polygons in Space

Hermann Jung
Humboldt-University, Berlin

Topological and geometrical properties of polygons in space are difficult to decide. We consider subclasses of the class of knots isotopic to the trivial knot. Among them are:

- polygons which admit triangulations of different kinds
- polygons which can be projected on the plane such that the projection is simple
- polygons that can be embedded on the surface of special types of polyhedra

We propose first polynomial time bounds for testing the membership for these subclasses.

Partition Trees

Jiří Matoušek
Charles University, Prague

We prove that given a n point set P in $\mathbf{E}^d$ (d a constant), there exists a partition $P = P_1 \dot{\cup} \ldots \dot{\cup} P_r$ and simplices $s_1, \ldots, s_r \subseteq \mathbf{E}^d$ with $P_i \subset s_i$, such that $\frac{n}{2r} \leq |P_i| \leq \frac{2n}{r}$ and no hyperplane intersects more than $\mathcal{O}(r^{1-1/d})$ of the s_i's; where r is a prescribed parameter. Using this one can show that a set P of n points in $\mathbf{E}^d$ equipped with a weight function $w : P \to S$, where S is a semigroup, can be preprocessed in $\mathcal{O}(n \log n)$ time, yielding a linear-space data structure which allows to evaluate $\sum_{p \in Q \cap P} w(p)$ for any query simplex Q in time $\mathcal{O}(n^{1-1/d}(\log n)^{O(1)})$. With $\mathcal{O}(n^{1+\delta})$ preprocessing time, the query time can be made $\mathcal{O}(n^{1-1/d}(\log \log n)^{O(1)})$, and further improvements are possible in special cases. Another application of the partition result is a fast computation of $1/r$-cuttings for collections of hyperplanes.

Turàn-type Theorems for Segments

Jànos Pach
Hungarian Academy of Sciences and Courant Institute, New York University

A *geometric graph* is a graph whose vertices are embedded in the plane in general position and whose edges are straight line segments. Given a class $\mathcal{H}$ of so called *forbidden* geometric subgraphs, let $t(\mathcal{H}, n)$ denote the maximum number of edges a geometric graph with n vertices can have without containing a subgraph isomorphic to an element of $\mathcal{H}$. Let $t_C(\mathcal{H}, n)$ be defined similarily, except that now the maximum is taken over all geometric graphs whose n vertices form a convex polygon.

Let $\mathcal{H} = \mathcal{C}_{k+1}$ be the family of all sets of $k+1$ pairwise crossing edges. By Euler's theorem, $t(\mathcal{C}_2, n) = 3n - 6$, i.e. every geometric graph with n vertices and more than $3n - 6$ edges contains two edges which cross each other. For $k \geq 2$, we only have

$$3kn - c_1 k \leq t(\mathcal{C}_{k+1}, n) \leq c_2 n^{2-(c_3/k^2)},$$

where c_1, c_2, c_3 are suitable positive constants. The upper bound follows from the Kővàri-Sòs-Turàn theorem and the following recent result of Aronov, Kleitman, Goddard and myself.

Theorem 1: Given $n/2$ red and $n/2$ blue points in the plane in general position, we can always choose $\sqrt{n/24}$ red-blue point pairs so that the segments determined by them are pairwise crossing. Furthermore, there is a polynomial time algorithm to find these point pairs.

In the case when the vertices of the geometric graphs are in convex position, V. Capolyeas and I have proved

Theorem 2:

$$t_c(\mathcal{C}_{k+1}, n) = \begin{cases} \binom{n}{2} & \text{if } n \leq 2k+1, \\ 2kn - \binom{2k+1}{2} & \text{if } n > 2k+1, \end{cases}$$

which is one of the very few exact results in this field.

Smallest Enclosing Disks (Balls and Ellipsoids)

Emo Welzl
Freie Universität Berlin

We present a simple randomized incremental algorithm for computing the smallest enclosing disk of n points in the plane; it also works for smallest enclosing balls or ellipsoids in $\mathbf{R}^d$, or for largest balls or ellipsoids contained in the intersection of n halfspaces in $\mathbf{R}^d$. The algorithm and its analysis is conceptually based on Raimund Seidel's recent Linear Programming algorithm. The expected running time is $\mathcal{O}(n)$ for constant dimension d, and the algorithm appears to be fast for small dimensions.

A Tail Estimate for the number of regions constructed by randomized incremental algorithms

Kurt Mehlhorn
Universität des Saarlandes

Let $m : \mathbf{N} \to \mathbf{R}_{\geq 0}$ be such that $m(n)/n$ is nondecreasing. Let T be a tree of depth n, such that every node of depth d, $0 \leq d < n$, has exactly $n-d$ children. Assume further that for each node v at depth d the edges starting in v have nonnegative labels which sum up to $m(n-d)$. For a leaf v of T, let $X(v)$ be the sum of the labels on the path from the root to v, and let X be the random variable defined by the uniform distribution on the $n!$ leaves of T. Then

$$\mathsf{Prob}(X \geq c \cdot m(n)) \;\leq\; \frac{1}{e}\left(\frac{c}{e}\right)^{-c}$$

for all $c \geq 1$. We then relate the random variable X to the number of regions constructed by randomized incremental algorithms.

Measuring the Resemblance of Polygonal Shapes

Helmut Alt
(joint work with Bernd Behrends and Johannes Blömer)
Freie Universität Berlin

Given two polygonal chains (or only sets of line segments) P and Q, we consider the problem of matching P and Q as good as possible. Formally, we look for an isometry I which minimizes the Hausdorff-distance $\delta_H(P, I(Q))$. We consider the following special cases of the general problem:

$\mathcal{I}0$: I is the identity (i.e. the problem is to determine the Hausdorff-distance $\delta_H(P, Q)$.

$\mathcal{I}1$: I is some translation along some fixed vector $\vec{t}$.

$\mathcal{I}2$: I is an arbitrary translation

$\mathcal{I}3$: the general problem

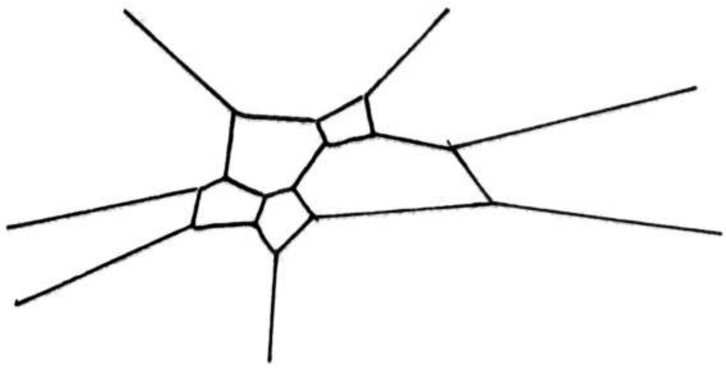
For $\mathcal{I}0, \ldots, \mathcal{I}3$ we obtain algorithms with runtimes $\mathcal{O}((p+q)\log(p+q))$, $\mathcal{O}((pq)\log(pq)\log^*(pq))$, $\mathcal{O}((pq)^3(p+q)\log(p+q))$, $\mathcal{O}((pq)^4(p+q)\log(p+q))$, respectively, where p and q are the number of vertices of P and Q.

For $\mathcal{I}2$ and $\mathcal{I}3$ we develop pseudo-optimal algorithms, i.e. ones producing solutions, which are not necessarily optimal, but a placement, where the Hausdorff-distance differs by at most a constant factor from the optimal one. The runtimes become much better: $\mathcal{O}((p+q)\log(p+q))$ and $\mathcal{O}((pq)\log(pq)\log^*(pq))$, respectively.

On-line Construction of Order-k Voronoi Diagrams via Convex Hulls

Franz Aurenhammer and Otfried Schwarzkopf
Freie Universität Berlin

We present an incremental algorithm for constructing the order-k Voronoi diagram in the plane. Given a set S of n points, this diagram subdivides the plane into regions, such that each point within a region has the same k nearest neighbours in S. The new algorithm relies on a duality between order-k Voronoi diagrams and convex hulls in 3-space. Inserting the regions defined by a newly inserted point essentially amounts to computing a particular convex hull. The algorithm is simple and on-line. It uses optimal $\mathcal{O}(kn)$ space and takes $\mathcal{O}(k^2 n \log n + kn \log^3 n)$ expected time if the points are inserted in random order. We also show that any randomized insertion algorithm for constructing the order-k Voronoi diagram must take $\Omega(k^2 n)$ expected time.



Offline Maintenance of Planar Configurations

John Hershberger
(joint work with Subhash Suri, Bellcore)
DEC Systems Research Center

We achieve an $\mathcal{O}(\log n)$ amortized time bound per operation for the offline version of the dynamic convex hull problem in the plane. Given a sequence of n insertions, deletions, and convex hull queries on a planar set of points, our algorithm processes the sequence and answers all the queries in $\mathcal{O}(n \log n)$ time and $\mathcal{O}(n)$ space. If we are allowed $\mathcal{O}(n \log n)$ space, our algorithm supports queries in history (we preprocess a sequence of n insertions and deletions; subsequent queries specify a time parameter t, with the requirement that the query be answered with respect to the convex hull present at that time t). The same result also holds for offline maintenance of several related structures, including maximal vectors, the intersection of half-planes, and the kernel of a polygon. Achieving an $\mathcal{O}(\log n)$ per-operation time bound for the online versions of these problems is a long-standing open problem in computational geometry, and our result should be viewed as a stepping stone toward that goal.

The Union of Fat Triangles

Micha Sharir
(joint work with J. Matoušek, J. Pach, S. Sifrony, and E. Welzl)
Tel Aviv University and New York University

Let $\delta > 0$ be a fixed constant. We say that a triangle Δ is *δ-fat* if every angle of Δ is $\geq \delta$. Let $\Delta_1, \ldots, \Delta_n$ be a given collection of n δ-fat triangles. We show:

1. The union $K = \bigcup \Delta_i$ has $\mathcal{O}(n)$ holes
2. The boundary ∂K consists of $\mathcal{O}(n \log\log n)$ edges.
3. If the Δ_i's are all wedges or triangles of roughly the same size (the ratio between any two edges of the triangles is bounded), then the complexity of ∂K is $\mathcal{O}(n)$
4. K can be computed in time $\mathcal{O}(n \log^2 \log\log n)$ by a deterministic algorithm, or in time $\mathcal{O}(n \log n \log\log n)$ by a randomized incremental algorithm

The constants of proportionality depend on δ. Günter Rote has shown that for wedges the constant is $\mathcal{O}((\frac{1}{\delta})^3 \log \frac{1}{\delta})$. These results have applications in several areas. One recent application to hidden surface removal is: Given n horizontal δ-fat triangles in 3-space, one can compute the way in which they are seen from $z = -\infty$ in time $\mathcal{O}(n \log^2 n \log\log n + k \log^2 n)$, where k is the size of the viewed scene (number of edges and vertices in the projected view).

Computing Trapezoidations

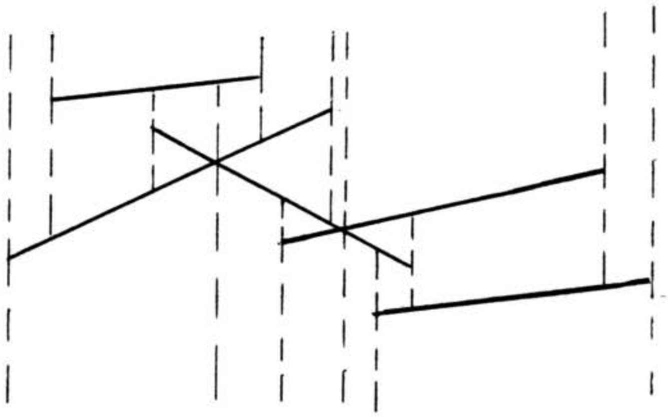
Raimund Seidel
Computer Science Division, University of California at Berkeley

We consider randomized algorithms for constructing the trapezoidal decomposition induced by a set S of n line segments in the plane.

At first, we consider Mulmuley's algorithm that deals with the case where segments in S are allowed to cross. Using ***backwards analysis*** we give a simple proof that the expected running time of that algorithm is $\mathcal{O}(n \log n + K)$, where K is the number of pairs of crossing segments in S.

Next we consider the non-crossing case. We present an incremental randomized algorithm with $\mathcal{O}(n \log n)$ expected running time that besides the trapezoidation $\mathcal{T}(S)$ also produces a data structure that allows point location queries in $\mathcal{T}(S)$ in expected time $\mathcal{O}(\log n)$.

Finally we show that when the segments in S form a simple polygonal chain our algorithm can be improved to achieve an expected running time of $\mathcal{O}(n \log^* n)$. This leads to a rather straightforward polygon triangulation algorithm with $\mathcal{O}(n \log^* n)$ expected running time that should be of practical interest.



Matching Points into Noise Regions: Combinatorial Bounds and Algorithms

Joseph S. B. Mitchell
(joint work with E. Arkin, K. Kedem, J. Sprinzak, and M. Werman)
SORIE, Cornell University

We consider the problem of finding a transformation $\tau : \mathbf{R}^2 \rightarrow \mathbf{R}^2$ of a point set $\mathcal{A} = \{a_1, \ldots, a_n\}$ that carries each point into some one of n "noise regions" $\mathcal{B} = \{B_1, \ldots, B_n\}$. We concentrate on the case of pairwise-disjoint regions $\mathcal{B}$, and we consider cases in which τ is pure translation, pure rotation, rotation and translation, similarity, etc.

We prove various bounds: 1) lower bounds on the number of possible perfect matches; 2) upper bounds on the number of perfect matches; 3) time bounds on computing all perfect matches. All bounds depend on $\mathcal{B}$ — which may be circles, squares or more general. For example, we show that under pure rotation and with $\mathcal{B}$ = circles, the upper and lower bound on the number of matches is n (tight), and they can be found in time $\mathcal{O}(n^2)$. For similarity transforms, all $\mathcal{O}(n^2)$ matches with unit circles are computed in time $\mathcal{O}(n^5 \log n)$. Etc., etc., etc. ...

The geometric three-clustering problem for the maximum diameter

Günter Rote
Technische Universität Graz, Institut für Mathematik

Given n points in the plane, we want to partition them into a given number k of subsets (*clusters*) such that the maximum of their diameters is as possible as small. For $k = 2$, an optimal $\mathcal{O}(n \log n)$-time algorithm is known, and for general k, Vasilis Capolyeas, Gerhard Wöginger, and I have shown that there is always an optimal k-clustering in which any two clusters can be separated by a line. This leads, in particular, to an $\mathcal{O}(n^5 \log n)$-time algorithm for the *three*-clustering problem. Since this time complexity is not overwhelming, I shall in my talk investigate alternative ways to tackle the problem: The question, whether there is a 3-clustering with maximum diameter less than d, is equivalent to the 3-colorability of the graph, whose vertices are the given points and which has an edge between two any vertices whose distance is at least d. This leads to the study of these "graphs of large distances". Of course, I am primarily interested in their properties as they relate to colorability, for example, to identify those graphs that are minimal under vertex removal with respect to the property that their chromatic number is four. I show that such a graph must at least contain a triangle (a property not shared by general graphs). I conjecture that these critical graphs are given by some class of structured examples. I also present a conjectured $\mathcal{O}(n^3)$-time algorithm to solve the 3-clustering problem. The results are preliminary.

Berichterstatter: Otfried Schwarzkopf

Bisher erschienene Titel:

W. Gentzsch, W.J. Paul:

Architecture and Performance, Dagstuhl-Seminar-Report 1 (9025), 18.6.1990 - 20.6.1990

K. Harbusch, W. Wahlster:

Tree Adjoining Grammars, 1st. International Worshop on TAGs: Formal Theory and Application, Dagstuhl-Seminar-Report 2 (9033), 15.8.1990 - 17.8.1990

Ch. Hankin, R. Wilhelm:

Functional Languages: Optimization for Parallelism, Dagstuhl-Seminar-Report 3 (9036), 3.9.1990 - 7.9.1990

H. Alt, E. Welzl:

Algorithmic Geometry, Dagstuhl-Seminar-Report 4 (9041), 8.10.1990 - 12.10.1990