Efficient Approximation and Optimization Algorithms for Computational Metrology

Christian A. Duncan*
Center for Geometric Computing
The Johns Hopkins University
duncan@jhu.edu

Michael T. Goodrich†
Center for Geometric Computing
The Johns Hopkins University
goodrich@jhu.edu

Edgar A. Ramos‡
DIMACS
Rutgers University
ramose@dimacs.rutgers.edu

Abstract

We give efficient algorithms for solving several geometric problems in computational metrology, focusing on the fundamental issues of "flatness" and "roundness." Specifically, we give approximate and exact algorithms for 2- and 3-dimensional roundness primitives, deriving results that improve previous approaches in several respects, including problem definition, running time, underlying computational model, and dimensionality of the input. We also study methods for determining the *width* of a d-dimensional point set, which corresponds to the metrology notion of "flatness," giving an approximation method that can serve as a fast exact-computation filter for this metrology primitive. Finally, we report on experimental results derived from implementation and testing, particularly in 3-space, of our approximation algorithms, including several heuristics designed to significantly speed-up the computations in practice.

1 Introduction

Dimensional Tolerancing and Metrology (DT&M) is concerned with the specification and measurement of error tolerances in geometric shapes (primarily for rigid manufactured parts). In the measurement (metrological) part of this task one is given a mathematical tolerance description and a set of points sampled from the surface of the geometric shape and asked if the set of points satisfies the given tolerance. Historically, this activity was a labor-intensive activity, but relatively recent developments in technology have led to the use of coordinate measurement machines, laser range-finders, and scanning electron microscopes (e.g., see [14]) to sample points on manufactured surfaces to provide large volumes of data for testing the conformance of these surfaces to given specifications. Unfortunately, the algorithms and software needed to quickly and accurately determine if these sample points conform to the given specifications has not advanced as far as one might have hoped. Thus, we are interested in the design of efficient, easy-to-implement algorithms for performing such *metrology primitives*, with particular attention focused on the fundamental issues of "flatness" and "roundness" [38].

*This research partially supported by ARO under grant DAAH04-96-1-0013.

†This research partially supported by NSF under Grants CCR-9300079 and CCR-9625289, and ARO under grant DAAH04-96-1-0013.

‡Supported by a DIMACS Postdoctoral Fellowship. DIMACS is a cooperative project of Rutgers University, Princeton University, AT&T Research, Bell Labs and Bellcore. DIMACS is an NSF Science and Technology Center, funded under contract STC-91-19999; and also receives support from the New Jersey Commission on Science and Technology.

1.1 Previous Related Work.

Since testing to see if a set of points conforms to a notion of flatness or roundness is essentially a geometric computation, it should come as no surprise that some of the issues involved in the design of algorithms for testing these concepts have been previously studied in the computational geometry literature. Indeed, the first paper in the first ACM Symposium on Computational Geometry is a paper by Houle and Toussaint [19] (which subsequently appeared in journal form [20]) on efficient algorithms for solving the flatness problem for points in $\mathbb{R}^2$ and $\mathbb{R}^3$. Interestingly, the specific problem they address, which they called the *width* problem, corresponds exactly to the mathematical notion of "flatness" that is used in the computational metrology community (e.g., see [38]). In this problem one is given a set S of n points in $\mathbb{R}^d$ and asked to find a closest pair of parallel hyperplanes in $\mathbb{R}^d$ that contain all the points of S on or between them. Houle and Toussaint [19, 20] give exact methods that run in $O(n \log n)$ time for points in $\mathbb{R}^2$ and in $O(n^2)$ time for points in $\mathbb{R}^3$. Their methods are based upon well-established computational geometry computations, including convex hull construction and line-segment intersection (e.g., see [17, 28]), and they can even be implemented in the rational-RAM computation model, where all arithmetic is performed exactly on rational numbers.

Several computational geometry researchers [1, 3, 4, 5, 9, 21, 30, 35] have studied the width problem further. Since the 2-dimensional algorithm of Houle and Toussaint is already quite efficient, most of the subsequent 2-dimensional work has been on dynamic approximation methods [21, 30], whereas further research on static algorithms has focused on the 3-dimensional problem. For example, the best previous deterministic algorithm for determining the width of a set of n points in $\mathbb{R}^3$ is due to Chazelle *et al.* [9] and runs in $O(n^{8/5+\epsilon})$ time, for any fixed constant $\epsilon > 0$. The best randomized algorithm for points in $\mathbb{R}^3$ is due to Agarwal and Sharir [5] and runs in expected time that is $O(n^{3/2+\epsilon})$, for any fixed constant $\epsilon > 0$ (see also [1]). All of these sub-quadratic methods are based on fairly sophisticated techniques, however, such as parametric searching [6, 9, 12, 25], and involve the computation of lower envelopes of non-linear algebraic surfaces in $\mathbb{R}^4$; hence, they cannot be implemented exactly in the rational-RAM model.

There has also been some interest in the computational geometry community on various 2-dimensional problems related to "roundness" [5, 6, 16, 22, 33, 35, 37]. All of these papers involve determining an annulus A that contains the input set S of n points in the plane. (Recall that an annulus A of radius ρ_A and width $0 \leq \omega_A \leq 2\rho_A$ is the region between two concentric circles of radius $\rho_A + \omega_A/2$ and $\rho_A - \omega_A/2$.) Ebara *et al.* [16] (see also [22, 35]) study the minimum-width annulus problem, where one wishes to find an annulus A that minimizes the quantity ω_A (over all annuli in the plane). They show that the center of a minimum-width annulus containing S must be at a vertex of the farthest-neighbor or nearest-neighbor Voronoi diagrams (allowing for vertices at infinity [35]) or at an intersection point between these two diagrams. This, of course, leads to an $O(n^2)$-time algorithm, which can be implemented in the rational-RAM model. Agarwal and Sharir [5] give a randomized algorithm for solving this problem in $O(n^{3/2+\epsilon})$ expected time, for any fixed constant $\epsilon > 0$, but their method cannot be implemented in the rational-RAM model, as their method essentially amounts to a reduction to a 3-dimensional width computation. Alternatively, Agarwal, Sharir, and Toledo [6] observe that one can solve the 2-dimensional minimum-area annulus problem in linear time, via a reduction to fixed-dimensional linear programming[1]. Shermer and Yap [33] consider another notion of roundness, however, which they call *relative roundness*, where one wishes to find the annulus A containing S that minimizes the quantity $\omega_A/(\rho_A - \omega_A/2)$, restricting its center to be in the convex hull of S. They give an $O(n^2)$-time algorithm for computing such an annulus A (which may not be unique, even for points in general position). They also identify an interesting research direction for DT&M algorithms that allows one to make certain "reasonable" assumptions about the input set. In the case of relative roundness, for example, they show that one can derive a linear time algorithm for point sets that are angularly uniformly-distributed about a "near center," "almost" round, and in convex position.

[1] In fact, in linear time, one may find the d-dimensional annulus of minimum difference in squared radius ($(\rho_A + \omega_A/2)^2 - (\rho_A - \omega_A/2)^2$) or, equivalently, minimize the product of the radius and width, $\rho_A\omega_A$, assuming that d is fixed.

1.2 Our Results. Unfortunately, while these previous notions of roundness may capture certain intuitive notions of what it means for a set of points to be round or almost-round, none of them solve the roundness problem motivated from DT&M. Recall that in the DT&M framework one must determine if a given set of points satisfies a given specification. It is easy (e.g., see Figure 1) to come up with sample sets S that satisfy a given specification, but for which a solution to one of these previous roundness definitions does not determine this fact.

We believe the more natural notion of roundness motivated from DT&M is something we call *referenced roundness*, where one wishes to find an annulus A with a given reference radius ρ that contains S and has

- width $\omega_A \leq \epsilon$, for a given $\epsilon > 0$; or
- minimum width, ω_A, taken over all annuli A with $\rho_A = \rho$.

We refer to these two problems, respectively, as the *decision* and *optimization* versions of the referenced roundness problem. In either case, our referenced roundness definition conforms to the *tolerance zone* semantics described by Requicha [29], Srinivasan [36] and Yap [40] for computational metrology.

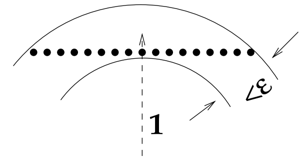

Figure 1: An example of a set with a roundness of 0 (a perfectly flat line) but a referenced roundness of ϵ.

In this paper we give a number of efficient, easy-to-implement algorithms for solving computational metrology problems dealing with the flatness and roundness primitives. For example, we give simple deterministic and randomized methods for solving the referenced roundness problem for planar point sets that run in $O(n \log n)$ time (with high probability for the randomized method). We also study the 3-dimensional (sphericality) version of this problem, as well, giving what appears to be the first non-trivial solution for a 3-dimensional roundness problem (we also give a simple approximation algorithm for this problem). Interestingly, our (exact) solution for 2-dimensional referenced roundness can also be used to solve the approximation version of the minimum-width annulus problem, where the output width is guaranteed to be within a factor of $1+\epsilon$ of the true width, in time $O(n(\log n + 1/\epsilon))$, with high probability, for point sets that satisfy a natural uniformity assumption.

An interesting aspect of the referenced roundness problem is that it becomes equivalent to the flatness (a.k.a., width) problem when $r = \infty$. We show in this case, however, that we can significantly improve the running time for determining the width of a set of points in $\mathbb{R}^d$, for any fixed $d \geq 1$, by considering an approximation version of this problem. In particular, we show how to compute a width that is guaranteed to be within a $1+\epsilon$ factor of the true width in time $O(\alpha^{d-1}n)$, where $\alpha = \pi/(2 * \arctan\sqrt{(2\epsilon+\epsilon^2)/(d-1)})$. We achieve this result by several reductions to fixed-dimensional linear programming, generalizing a 2-dimensional approach of Janardan [21].

It may not be immediately apparent, but tolerancing metrology actually lends itself quite naturally to approximation algorithms, in a fashion analogous to the adaptive-precision approach of exact arithmetic computations [18, 34, 39]. For example, if a sample of a machine part with a tolerance of $\pm$1mm were tested using an approximation algorithm with an error of $\pm$0.1mm, a reported tolerance t with $t <$ .9mm would, even at maximum possible error, still guarantee that the sample falls within the specified tolerance, and a reported tolerance with $t >$ 1.1mm would, at maximum error, guarantee that the sample fails tolerance. On the other hand, if a sample has a tolerance that is "too close to call," which should be a fairly rare event if the manufacturing process is correctly calibrated, then we can resort to a (slow) exact algorithm (e.g., [20]).

We have implemented our approximation approach to the referenced roundness and width problems and provide the results of benchmarking tests, which give empirical evidence to the efficiency and ease-of-implementation of our methods. In addition, we describe several heuristics for improving the running time of our methods in practice, and we give empirical evidence that these heuristics do indeed improve running times.

In the sections that follow we outline the main ideas of our results.

2 Exact and Approximate Referenced Roundness

In this section we give efficient algorithms for the *referenced roundness* problem in the plane. The first algorithm is deterministic and uses a simple version of parametric searching [6, 9, 12, 25], running in $O(n \log n)$ time; the second one, which is somewhat simpler, is randomized and achieves a running time of $O(n \log n)$ with high probability. The deterministic algorithm generalizes to an algorithm for solving the referenced roundness problem in 3-dimensions in near-quadratic time. Using the referenced roundness algorithm, we obtain an approximation algorithm for the minimum width annulus problem, which, under certain reasonable uniformity assumptions on the input, determines the minimum width to within a factor $1+\epsilon$ and runs in time $O(1/\epsilon)$ times the time for the referenced roundness algorithm. We also describe a simple approximation algorithm for the referenced roundness problem in a fixed dimension d with absolute error ϵ and running time $O(n/\epsilon^d)$, which can be improved to $O(n \log n + (1/\epsilon^2)\log n)$ in the plane.

2.1 Preliminaries.

Let S be a set of n points in $\mathbb{R}^d$; we will be interested in $d = 2, 3$ although the definitions and some of the results extend to higher dimensions.

Balls, annulus, intersection and union of balls. For a point p in the plane and a real number $r \geq 0$, let $b_r(p)$ denote the ball of radius r centered at p. Let $A_{r,w}(p)$ be the annulus $b_{r+w/2}(p) - \mathrm{int}(b_{r-w/2}(p))$. We say that $A_{r,w}$ has *radius* r and *width* w. Let $\mathcal{I}_r = \mathcal{I}_r(S) = \bigcap_{p\in S} b_r(p)$ denote the intersection of the r-radius balls centered at points in S and let $\mathcal{U}_r = \mathcal{U}_r(S) = \bigcup_{p\in S} b_r(p)$ denote the union of this set of balls. Also, let $X_{r,w} = X_{r,w}(S) = \bigcap_{p\in S} A_{r,w}(p)$. Note that $X_{r,w} = \mathcal{I}_{r+w/2} - \mathrm{int}(\mathcal{U}_{r-w/2})$. For fixed r, we call $X_w = X_{r,w}$ the *w-feasible region*, because $q \in X_{r,w}$ if and only if $S \subseteq A_{r,w}(q)$.

Nearest and furthest point Voronoi diagrams. The nearest point Voronoi diagram $\mathcal{V}_n = \mathcal{V}_n(S)$ of S is the complex of closed convex cells consisting of the d-dimensional cells $V_p^n(S) = \{q \in \mathbb{R}^d : \|q-p\| \leq \|q-p'\|, p' \in S\}$ and their intersections $V_{S'}^n(S) = \bigcap_{p\in S'} V_p^n$, for $S' \subseteq S$. The 0- and 1-cells are called *vertices* and *edges*. The furthest point Voronoi diagram $\mathcal{V}_f = \mathcal{V}_f(S)$ is defined similarly with the d-cells $V_p^f(S) = \{q \in \mathbb{R}^d : \|q-p\| \geq \|q-p'\|, p' \in S\}$ (which are non empty iff p is on the boundary of the convex hull of S). Both Voronoi diagrams have size $O(n^{\lceil d/2\rceil})$ [24] ($O(n)$ for $d = 2$ and $O(n^2)$ for $d = 3$) and can be computed in time $O(n \log n + n^{\lceil d/2\rceil})$ [8] ($O(n \log n)$ for $d = 2$ and $O(n^2)$ for $d = 3$). The 1-skeleton of $\mathcal{V}_n$ (resp. $\mathcal{V}_f$) is the subcomplex consisting of the 0- and 1-cells. For further information see, e.g., [17, 28]. The *carrier* $c_n(q)$ of a point q in $\mathcal{V}_n$ is the cell in $\mathcal{V}_n$ of smallest dimensionality that contains q (we will use the concept of carrier for objects other than points, for example, the carrier of an edge). $c_n(q)$ is V_T^n where T is the set of *sites* in S

nearest to q. Let $n(q)$ be the corresponding distance to a nearest site. Then q is in the boundary of $\mathcal{U}_{n(q)}$. Similarly, $c_f(q)$ is the carrier of q in $\mathcal{V}_f$, $f(q)$ is the distance to a furthest site, and q is in the boundary of $\mathcal{I}_{f(q)}$.

Constructing the ω-feasible region. Given a point set S, a radius ρ_0 and a width ω, we compute the ω-feasible region $X_\omega = X_{\rho_0,\omega}(S)$.

LEMMA 2.1. *Given $\mathcal{V}_n$ and $\mathcal{V}_f$, X_ω can be constructed in time $O(n)$ for $d = 2$ and in time $O(n^2)$ for $d = 3$.*

Proof. Let us consider first the case $d = 2$. Let $R = \rho_0 + \omega/2$ and $r = \rho_0 - \omega/2$. Let $P = \mathrm{bd}(\mathcal{I}_R)$ and $Q = \mathrm{bd}(\mathcal{U}_r)$. Assuming $\mathcal{V}_n$ and $\mathcal{V}_f$ are known, P and Q can be computed in time $O(n)$. More precisely, in time $O(n)$, we can determine P within each cell of $\mathcal{V}_f$, and Q within each cell of $\mathcal{V}_n$. Next, taking advantage of the fact that $\mathcal{I}_R$ is convex, we compute the intersection of P with the cells of $\mathcal{V}_n$ in time $O(n)$ by walking along P and the portions outside $\mathcal{I}_R$ of the cells of $\mathcal{V}_n$ that intersect P. See Figure 2.

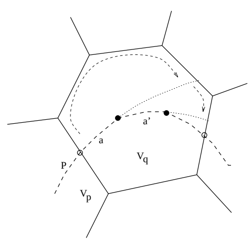

Figure 2: Intersecting P and $\mathcal{V}_n$.

Next, the intersection of P and Q is obtained by walking along P on $\mathcal{V}_n$. As each cell c is visited, the intersection with Q inside c is computed; for this, the following information is needed: the relative order of the intersections of P and Q with the boundary of c and the relative position of the vertices and edges of P in c respect to the ball in c. Finally, X_ω is computed: If the intersection of P and Q is empty then the result depends on whether P is contained in $\mathcal{U}_r$ and on whether some components of Q are contained in $\mathcal{I}_R$. If their intersection is not empty, then the boundary of X_ω consists of a portion of P and the portions of Q inside P. The overall time used is $O(n)$.

Each of the steps can be extended to the case $d = 3$. In this case the size of $\mathcal{V}_n$ and $\mathcal{V}_f$ is $O(n^2)$. As before, we first compute the intersection of P with $\mathcal{V}_f$ and of Q with $\mathcal{V}_n$, but now in time $O(n^2)$ (even though the size of P is just $O(n)$). Note that the size of the intersection of P with $\mathcal{V}_n$ is $O(n^2)$. The same bound is true then for the intersection of P and Q, and so for the boundary of X_ω. Thus, taking advantage of the fact that $\mathcal{I}_R$ is convex, we compute the intersection of P with the cells of $\mathcal{V}_n$ and the intersection of P and Q in time $O(n^2)$. Finally, X_ω is obtained using overall time $O(n^2)$. The details are omitted here. ■

Thus, the decision version of the referenced roundness problem can be solved in time $O(n)$ for $d = 2$ and in time $O(n^2)$ for $d = 3$, given $\mathcal{V}_n$ and $\mathcal{V}_f$.

2.2 Algorithms for optimal referenced roundness in the plane. Let us now consider the optimization version of referenced roundness, which is to determine, given a point set S and a radius $\rho_0 \geq 0$, the minimum width $\omega_0 \geq 0$ such that for some point x, $S \subseteq A_{\rho_0,\omega_0}(x)$. Note that this is equivalent to determining the minimum ω_0 so that $X_{\rho_0,\omega_0} \neq \emptyset$.

As in the previous subsection, since the radius ρ_0 remains fixed, we drop ρ_0 from the notation. Let us assume that for some x, $S \subseteq A_{2\rho_0}(x)$, and hence $X_{\rho_0} \neq \emptyset$ (this can be determined quickly given $\mathcal{V}_n$ and $\mathcal{V}_f$). Consider X_ω as ω decreases from $\omega = 2\rho_0$ to $\omega = 0$. X_ω decreases monotonically, that is, if $x \notin X_{\omega'}$ then $x \notin X_\omega$ for $\omega < \omega'$. This is because $\mathcal{I}_{\rho_0+\omega/2}$ decreases monotonically, while $\mathcal{U}_{\rho_0-\omega/2}$ increases monotonically. This elementary observation is essential for being able to obtain a fast algorithm because it allows the use of a binary search to determine ω_0. Specifically, it implies that for $\omega_0 \leq \omega \leq 2\rho_0$, $X_\omega \neq \emptyset$ while for $\omega < \omega_0$, $X_\omega = \emptyset$.

Deterministic algorithm. Using the technique of parametric searching and a parallel version of the algorithm of the previous section for constructing an ω-feasible region, we can obtain an algorithm for the optimization version of the referenced roundness problem that runs in time $O(n \log n)$.

THEOREM 2.1. *The optimal referenced roundness problem in the plane can be solved in time $O(n \log n)$ deterministically.*

Proof. In the construction of the feasible region, all comparisons involving ω can be parallelized into $O(1)$ batches of $O(n)$ comparisons or $O(n)$ independent binary searches. Other parts of the algorithm not involving ω are performed sequentially. We ellaborate on this claim: To compute $P = \mathrm{bd}(\mathcal{I}_R)$, first determine in parallel for each edge of $\mathcal{V}_f$ whether it intersects P (this requires $O(1)$ comparisons involving ω), and then construct P sequentially; to obtain the intersection of P and $\mathcal{V}_n$, for each vertex of $\mathcal{V}_n$ in parallel, using a binary search (point location) determine whether it lies inside or outside $\mathcal{I}_R$, and for each edge of $\mathcal{V}_n$ in parallel, using a binary search determine whether it intersects P; finally, to compute the intersection of P and Q, for each edge of $\mathcal{V}_n$ in parallel, determine the relative position of its intersections with P and Q, for each vertex of P in parallel determine whether it lies inside or outside the ball in its carrier in $\mathcal{V}_n$ containing it, and for each segment of edge of P in parallel determine whether it intersects the boundary of the ball in its carrier in $\mathcal{V}_n$. The algorithm uses $O(n \log n)$ work. Then, using parametric searching, with this algorithm and the sequential $O(n)$ time oracle from the previous subsection, plus Cole's speeding up trick [12], we derive an optimization algorithm using overall time $O(n \log n)$. ■

Randomized algorithm. Even though the previous method uses a simple version of parametric search, we are aiming for an algorithm that combines the maximum achievable efficiency and simplicity. So we describe an even simpler randomized algorithm for referenced roundness in the plane.

For any edge e in $\mathcal{V}_n$ (resp. $\mathcal{V}_f$) its *center* is the intersection of e with the line segment joining the sites determining e (if nonempty). If we introduce these centers as additional vertices, hence splitting some edges into two new edges, then we have the following observation (actually, in $\mathcal{V}_f$ only one center may be needed, the center of the minimum enclosing disk): $n(q)$ is monotone along edges in $\mathcal{V}_n$; $f(q)$ is monotone along edges in $\mathcal{V}_f$. Let $\omega_n(x) = 2(\rho_0 - n(x)))$ and $\omega_f(x) = 2(f(x) - \rho_0)$. ω_n and ω_f are monotone on edges of $\mathcal{V}_n$ and $\mathcal{V}_f$ respectively. Let $\omega(x) = \min\{\omega_n(x), \omega_f(x)\}$. Note that x disappears from X_ω at $\omega = \omega(x)$.

LEMMA 2.2. *ω_0 is $\omega(x)$ for some point x on the 1-skeleton of $\mathcal{V}_n$ or $\mathcal{V}_f$.*

Figure 3 shows $X_{\rho_0,\omega}$ just before vanishing along edges in $\mathcal{V}_n$ and $\mathcal{V}_f$. Large circles are shown dashed and small circles are shown continuous.

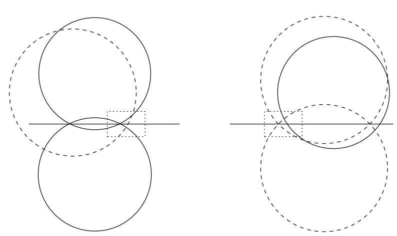

Figure 3: $X_{\rho_0,\omega}$ vanishing along edges in nearest and furthest Voronoi diagrams.

For edges e_n and e_f in $\mathcal{V}_n$ and $\mathcal{V}_f$ respectively, let $v(e_n, e_f)$ be their intersection if it exists. Let $\mathcal{I}$ be the set of those intersection points $v(e_n, e_f)$, and let $\mathcal{I}'$ consist of $\mathcal{I}$ together with the vertices of $\mathcal{V}_n$ and $\mathcal{V}_f$ and the centers of their edges. An *elementary segment* is a segment e' of an edge e_n or e_f in $\mathcal{V}_n$ or $\mathcal{V}_f$, whose endpoints are in $\mathcal{I}'$ and contains no point in $\mathcal{I}'$ in its interior. For an elementary segment e', the minimum ω so that $e' \cap X_\omega \neq \emptyset$, denoted $\omega(e')$, can be computed in constant time (assuming the carrier in the other Voronoi diagram is known). By the previous lemma, to obtain ω_0, it is sufficient to compute $\omega(x)$ for $x \in \mathcal{I}'$, and $\omega(e')$ for all elementary edge pieces. Since in the worst case there may be $\Theta(n^2)$ of them, we need a way to reduce the number of them that need to be considered. The way we proceed is inspired by randomized algorithms for the slope selection problem [13, 23, 32].

First, we establish a linear ordering $<_l$ on the points in $\mathcal{I}'$ as follows: $v <_l v'$ iff $\omega(v) < \omega(v')$ or (to break matches) $\omega(v) = \omega(v')$ and $v <_a v'$ where $<_a$ is some other arbitrary linear ordering. The key point of the approach is the following: For each $k > 0$, there are constants $C, C' > 0$ such that if we choose at random $K = Cn$ points $v_1 <_l \cdots <_l v_K$ from $\mathcal{I}$ then with probability at least $1 - 1/n^k$, each interval v_i, v_{i+1} contains at most $C'n \log n$ points in $\mathcal{I}$. In particular, there are $O(n \log n)$ points v in $\mathcal{I}$ with $v <_l v_1$; that is, with $\omega(v) < \omega(v_1)$. Let $\omega^* = \omega(v_1)$. It remains to find the portions of the 1-skeletons of $\mathcal{V}_n$ and $\mathcal{V}_f$ contained in X_{ω^*} and from there the points in $\mathcal{I}'$ contained in X_{ω^*}, and deal with all the corresponding elementary segments directly. Using simple techniques, the steps of the algorithm described can be implemented to run in $O(n \log n)$ time, with high probability, which establishes the following theorem:

THEOREM 2.2. *The optimal referenced roundness problem can be solved in time $O(n \log n)$, with high probability.*

2.3 Roundness in 3-d space. The roundness problem in 3-dimensional space is also of practical importance, but has not received much attention. In this case the ideal shape is a sphere or a spherical patch. Our referenced roundness concept and the approach to computing it extend to the 3-dimensional case. As already pointed out, the feasible region has complexity $O(n^2)$ and can be computed from the Voronoi diagrams in time $O(n^2)$; then using parametric searching we derive an efficient algorithm for referenced sphericality.

THEOREM 2.3. *The referenced roundness problem in 3-dimensional space can be solved in time $O(n^2 polylog(n))$.*

2.4 Approximation algorithm for referenced roundness. We develop a simple approximation algorithm for the referenced roundness problem in any fixed dimension $d \geq 2$. We show that in $O(\alpha^d n)$ time we can compute ω' such that $\omega' - \omega \leq \epsilon$ where $\alpha = \sqrt{d}/\epsilon$ and ω' and ω are the approximate and true widths, respectively. The algorithm is quite straightforward, it works by dividing a box bounding the points into a uniform grid of size α^d, but the dependency on ϵ is extremely high. To alleviate this burden, we show that various heuristics can significantly reduce the α value in practice, including the use of approximate nearest- and farthest-neighbor searching [2, 7]. In the full version we establish the following.

THEOREM 2.4. *The referenced roundness of a point set S in $\mathbb{R}^d$, for fixed $d \geq 2$, can be approximated in $O(\alpha^d n)$ time such that $\omega' - \omega \leq \sqrt{d}/\alpha$, where ω' and ω are the approximate and true widths respectively.*

In our section on implementation we explore some heuristics that should reduce the α^d factor significantly in practice.

2.5 Approximation algorithms for min-width annulus. Recall the minimum-width annulus problem, where one is given a point set S in the plane, and asked to determine the minimum width $\omega_m \geq 0$ such that for some radius ρ_m and some point x_m, $S \subseteq A_{\rho_m,\omega_m}(x_m)$. As mentioned earlier, the min-width annulus algorithms with running times that are $o(n^2)$ [5, 6] do not seem to be of practical value in the near future. Thus, we consider approximation algorithms, in the spirit of the approach of Shermer and Yap [33], under reasonable assumptions about the input data, which are realistic in the sense that they can be expected from data in tolerancing metrology or enforced in the measurement process. Also, an absolute error does not seem acceptable since the width is likely to be very small. We introduce a reasonable restriction on the input data that allows to obtain a relative error: We say that S is *θ-uniform* if a minimum width annulus $A_{\rho_m,\omega_m}(x_m)$ is so that any sector of angle θ centered at x_m intersects S.

Our approach is quite simple: First determine an approximation ρ_e of ρ_m such that $|\rho_e - \rho_m| \leq C\omega_m/2$, and an approximation ω_e of ω_m such that $\omega_m \leq \omega_e \leq D\omega_m$, for some constants C, D. Then repeat the referenced roundness algorithm $R = E(1/\epsilon)$ times with values of ρ uniformly distributed between $\rho_e - C\omega_e/2$ and $\rho_e + C\omega_e/2$, with $E = 2CD$. Note that if $|\rho_e - \rho_m| \leq \Delta$ then $|\omega(\rho_e)/2 - \omega_m/2| \leq \Delta$. So, it only remains to describe how to obtain ρ_e and ω_e.

Let $z(S)$ be the center of the minimum enclosing disk of S. Then let ρ_e and ω_e be the radius and the width of the minimum width annulus centered at z. If $\omega_m/2\rho_m > 1/5$ then the conditions required for ρ_e and ω_e are trivially satisfied: $|\rho_e - \rho_m| \leq \rho_m < 5\omega_m/2$, and $\omega_e/2 \leq \rho_m < 5\omega_m/2$. So assume $\omega_m/2\rho_m \leq 1/5$. Figure 4 shows a min-width annulus and the minimum enclosing disk (in the extreme case that its radius is equal to the exterior radius of the min-width annulus).

If the centers are displaced by $k\omega_m/2$ with $k \geq 2$

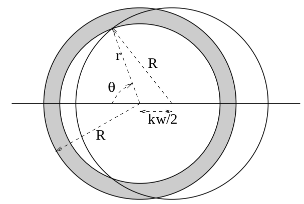

Figure 4: Approximation by minimum enclosing disk.

then the angle θ shown in the figure satisfies

$$\cos\theta = \frac{2}{k} - \frac{\omega_m/2}{\rho_m - \omega_m/2}\left(\frac{k}{2} - \frac{2}{k}\right).$$

Take $k \geq 4$. Then $\cos\theta \leq 1/2$ and hence $\theta \geq \pi/6$. Assuming that S is $(\pi/3)$-uniform, then $k \leq 4$ and consequently $|\rho_e - \rho_m| \leq 5\omega_m/2$ and $\omega_e/2 \leq 5\omega_m/2$. The concept of uniformity can be extended to $d = 3$ (using cones), and a similar analysis applies. Thus, we have:

THEOREM 2.5. *There is an approximation algorithm for the min-width annulus problem on θ-uniform sets of points, with appropriate θ, that outputs a width within a factor $1+\epsilon$ of optimal in time $O((1/\epsilon)n\log n)$ for $d = 2$ and in time $O((1/\epsilon)n^2 polylog(n))$ for $d = 3$.*

In the plane, using the randomized algorithm we obtain a somewhat simpler randomized algorithm with the same running time with high probability. The same approach estimates the center of a min-width annulus within distance $O(\omega_m)$. Therefore, using the grid approach described above, we obtain the following alternative algorithms:

THEOREM 2.6. *There is an approximation algorithm for the min-width annulus problem on θ-uniform sets of points, with appropriate θ, that outputs a width within a factor $1 + \epsilon$ of optimal in time $O(n/\epsilon^d)$. For $d = 2$, this can be improved to $O(n\log n + (1/\epsilon^2)\log n)$.*

3 Approximating the Width

Recall that in the *width* problem one is given a set S of n points in $\mathbb{R}^d$ and asked to find a closest pair of hyperplanes that contain all the points of S on or between them. Janardan [21] describes an algorithm that efficiently maintains the approximate width of a dynamic 2-dimensional point-set in $O(\alpha\log^2 n)$ time. In this section we extend his approach to d-dimensions and show how a carefully formulated subproblem is equivalent to a linear program with $O(n)$ constraints and $d + 1$ variables, which, for $d \geq 2$ being a fixed constant, is a problem that can be solved in $O(n)$ time [10, 11, 15, 26, 27, 31].

3.1 Skewed distances between hyperplanes. Let $\mathcal{A}$ and $\mathcal{B}$ be two parallel hyperplanes in d-space defined by the respective equations $(a_0, a_1, \ldots, a_{d-2}, -1)\mathbf{x} + a_{d-1} = 0$ and $(b_0, b_1, \ldots, b_{d-2}, -1)\mathbf{x} + b_{d-1} = 0$. Since the two planes are parallel, we know that their normals are parallel, which implies that $a_0 = b_0, a_1 = b_1, \ldots, a_{d-2} = b_{d-2}$.

Recall that the distance, δ, between these two hyperplanes is $|a_{d-1} - b_{d-1}|/\sqrt{N \cdot N}$, where the normal N to the hyperplanes is $N = (a_0, a_1, \ldots, a_{d-2}, -1)$. We define the *skewed* distance, $\mathcal{F}(\delta)$, between two hyperplanes as $\mathcal{F}(\delta) = |a_{d-1} - b_{d-1}| = \delta\sqrt{a_0^2 + a_1^2 + \cdots + a_{d-2}^2 + 1}$, which can be significantly larger than δ for large values of $a_0, a_1, \ldots,$ or a_{d-2}. However, if the normal to the hyperplanes is small, we can see that $\mathcal{F}(\delta) \approx \delta$. We will exploit this property shortly.

3.2 The *Skewed* Width Problem. Before we approximate the minimum width problem, let us consider the simpler problem of finding the pair of hyperplanes containing S with minimum skewed distance, i.e. the skewed width problem.

In order to solve this simpler width problem, we must first find at least two parallel hyperplanes containing all of the points between them. It is known that a hyperplane $\mathcal{A}$ (resp. $\mathcal{B}$) is above (below) all of the points in S if and only if $\forall p \in S, p$ lies in the half-space defined as $\mathcal{A} \geq 0$ (resp. $\mathcal{B} \leq 0$).

Thus, our search space is the set P of all pairs of parallel planes $(\mathcal{A}, \mathcal{B})$ such that $\mathcal{A}$ (resp. $\mathcal{B}$) is above (below) all points in S. Notice also that the boundary of P corresponds to the convex hull of S.

We now proceed to find the pair $(\mathcal{A}, \mathcal{B}) \in P$ with the minimum skewed distance. From the above, we can see that this optimization problem is simply a linear program of the form:

$$
\begin{array}{l}
\text{Minimize} \\
\quad a - b \\
\text{such that} \\
\quad \forall p \in S, \text{with } p = (p_0, p_1, \ldots, p_{d-1}) \\
\quad (a_0, a_1, \ldots, a_{d-2}, -1) \cdot (p_0, \ldots, p_{d-1}) + a \geq 0 \\
\quad (a_0, a_1, \ldots, a_{d-2}, -1) \cdot (p_0, \ldots, p_{d-1}) + b \leq 0
\end{array}
$$

This is a linear program with $2n$ constraints and $d + 1$ variables, which can be solved in $O(n)$ time for fixed d. However, as noted by Janardan [21], the skewed width solution does not necessarily yield an accurate approximation to the width, because $\mathcal{F}(\delta)$ can be significantly larger than δ, as noted in Section 3.1.

3.3 Approximating the Width. In the plane to yield a tightly bounded approximation, we solve the skewed width problem in α different coordinate systems $C_i = (X_i, Y_i)$, for some $\alpha > 1$, where X_i and Y_i are the respective x- and y-axes. X_0 is horizontal (original x-axis) and $X_i (i = 1, \ldots, \alpha - 1)$ makes an angle of $\frac{\pi}{\alpha}$ with X_{i-1}. In one of the C_i, the optimal solution has parallel lines with slope m such that $|m| \leq \tan\frac{\pi}{2\alpha}$. Intuitively, one can see that the optimal lines must lie at an angle which is in the range $(\frac{-\pi}{2\alpha}, \frac{\pi}{2\alpha})$ for some coordinate system.

This idea can easily be extended to the d-dimensional case by having α^{d-1} coordinate systems $C_{i_0 \ldots i_{d-2}} = (X_0, \ldots, X_{d-1})_{i_0 \ldots i_{d-2}}$ for $0 \leq i_o \ldots i_{d-2} \leq \alpha$ where the plane formed by the axes X_j and $X_{d-1} (j = 0, \ldots, d-2)$ is rotated within the plane by an angle $\frac{\pi i_j}{\alpha}$. Now the optimal pair of hyperplanes with normal $N = (m_0, \ldots, m_{d-2}, -1)$ must have all $|m_j| \leq \tan\frac{\pi}{2\alpha}$ for some coordinate system $C_{i_0 \ldots i_{d-2}}$, denoted by C_ω.

Recall, from Section 3.1, that $\mathcal{F}(\delta) = \delta\sqrt{N \cdot N}$. Let $\mathcal{F}(\delta_{i_0 \ldots i_{d-2}})$ for $(0 \leq i_0 \ldots i_{d-2} \leq \alpha)$ be the minimum skewed width in the coordinate system $C_{i_0 \ldots i_{d-2}}$. Specifically, let $\mathcal{F}(\delta_\omega)$ be the minimum skewed width found in C_ω. Let W' be the minimum over all $\delta_{i_0 \ldots i_{d-2}}$, and let W be the true optimal width over the set S. It follows that

$$
\begin{aligned}
W' &\leq \delta_\omega \leq \mathcal{F}(\delta_\omega) \leq \mathcal{F}(W) = W\sqrt{N \cdot N} \\
&\leq W\sqrt{1 + (d-1)\tan^2\frac{\pi}{2\alpha}}
\end{aligned}
$$

Letting $\frac{\pi}{2\alpha} = \arctan\sqrt{\frac{2\epsilon + \epsilon^2}{d-1}}$ yields $W' \leq W(1 + \epsilon)$.

Since the linear program in each coordinate system can be constructed in $O(n)$ time and solved in $O(n)$ time for fixed d, the total running time of our algorithm is $O(\alpha^{d-1}n)$ time.

THEOREM 3.1. *Given a set S of n points in $\mathbb{R}^d$, for a fixed $d \geq 2$, one can compute in $O(n)$ time an approximation W' to the width of S such that $W' \leq (1 + \epsilon)W$, where W is the width of S, for any fixed constant $\epsilon > 0$.*

4 Implementation and Experimentation

Both the referenced roundness and minimum width approximation algorithms, although running in linear time, depend heavily on the number of grids or coordinate systems, respectively, hereafter referred to as *partitions*, tested. A natural question then is, "Can we effectively reduce the number of partitions tested?"

4.1 Implemented heuristics for the approximate width problem. In practice, for the width problem, the answer is probably "yes," since most width computations will be performed on point sets that are known to be reasonably flat. Theoretically, of course, one can construct almost degenerate sets of points nearly uniformly distributed on a unit hypersphere, for which the answer to the above question would be "no."

Therefore, we implemented and tested our approximate width algorithm on example point sets (particularly in $\mathbb{R}^3$) that were known to be "fairly flat." With this application in mind, we analysed some heuristics for reducing the number of tested coordinate systems needed to produce an accurate approximate solution to the width problem. Even though the following heuristics exploit this flatness property, they remain robust enough to handle sets of other shapes.

To construct our fairly flat surfaces of varying orientation, size, and width, we selected random points on a surface of random size and orientation in space and perturbed them slightly away from the surface at random amounts. The graph in Figure 5 shows two values: the maximum error ratio between the approximate and true widths (a straightforward formula) and the average ratio between the approximate and true widths as reported in our testings for various values of α. As can be seen from the graph, regardless of the theoretical error rate, for fairly flat surfaces, the algorithm tends to perform extremely well in approximating the width.

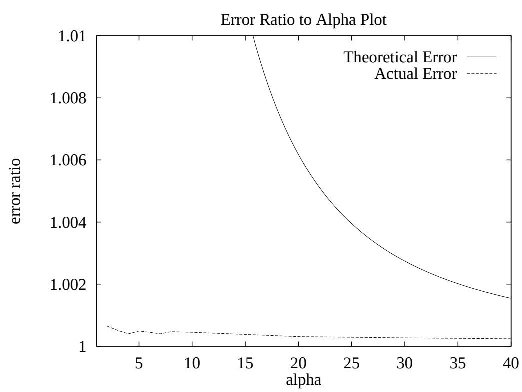

Figure 5: Comparing the theoretical error bound to actual error bounds.

The bottleneck in this approximation algorithm is the heavy dependence of the running time on α^{d-1}. If we can, in any way, prevent several possibly needless partitions from running the linear program, we drastically reduce the running time, allowing even better approximations in equal amounts of time. We begin with a simple observation.

LEMMA 4.1. *Let δ_i and N_i be, respectively, the minimal distance between and normal to the two supporting hyperplanes found by our linear program. For any two parallel hyperplanes with normal N_j and distance δ_j, if $N_j \cdot N_j \leq N_i \cdot N_i$, then $\delta_i \leq \delta_j$.*

Proof. Since $\delta = \mathcal{F}(\delta)/\sqrt{N \cdot N}$, $\mathcal{F}(\delta_i) \leq \mathcal{F}(\delta_j)$, and $N_i \cdot N_i \geq N_j \cdot N_j \geq 1$, it follows that $\delta_i \leq \delta_j$. ∎

Before we can remove all possible planes fulfilling this property, we need one more observation. Let N_{i_P} be the normal for any pair of parallel hyperplanes, $\mathcal{A}_i$ and $\mathcal{B}_i$, in the partition P, and let $\mathcal{P}$ be the set of all partitions tested. We define the *scope* of a partition P to be the set $\mathcal{S}_P$ of all pairs $(\mathcal{A}_i, \mathcal{B}_i)$ having normal N_i such that $N_i \cdot N_i \leq N_{i_Q} \cdot N_{i_Q}, \forall Q \in \mathcal{P}$. In other words, the scope contains all hyperplanes whose normal is minimized in that particular partition.

Because the normal is minimized, the distance $\mathcal{F}(\delta_i)$ between the hyperplanes $\mathcal{A}_i$ and $\mathcal{B}_i$ is minimized in the partition P where $(\mathcal{A}_i, \mathcal{B}_i) \in S_P$. If we reject an entire set $\mathcal{S}_P$ in another partition Q, we can ignore partition P because all remaining planes must be elements of the scopes of other partitions. Consequently, we may eliminate, after only a few linear program calls, several unlikely candidate partitions.

In order to maximize the elimination process, we now only need to determine the evaluation order of the partitions. We implemented three slightly different heuristics. The first of our heuristics simply does each partition in lexicographic order. The second and third implementations "bounce" around the partition set in, respectively, a random order and a predefined manner, trying the middle partition first, subdividing, and recursing on each half. The graph in Figure 6 shows the number of partitions tested for a random sample of fairly flat hyperplanes in the standard mode (α^{d-1}) and in each of these three heuristics.

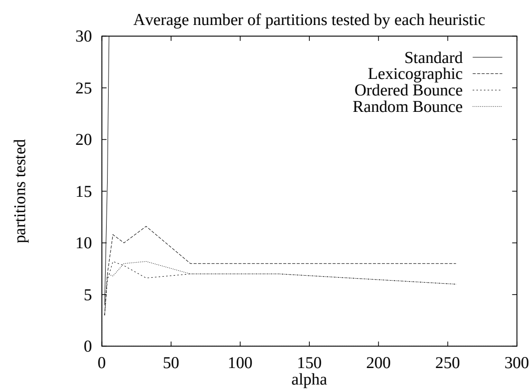

Figure 6: Comparing the partition-reduction heuristics to the standard mode. Each point on the graph is an average of four random orientations of an almost-flat set of 50,000 points in $\mathbb{R}^3$.

As one can see, these heuristics provide a significant decrease in the number of partitions tested, yielding a much improved running time over the original method for higher values of α. Thus, utilization of the above heuristics can significantly improve performance.

4.2 Implemented heuristics for the referenced roundness problem. Previously, we showed that the referenced roundness could be approximated in linear time with a heavy dependency on ϵ. To cut the dependence upon ϵ for several practical scenarios, we use the following observation.

LEMMA 4.2. *Let P be any subdivision of a bounding box for a set S. Let g be a grid of P with a referenced roundness ρ_g at its center and an error bound of δ_g equivalent to the distance from its center to farthest corner, and let ρ' be the referenced roundness for any point in P. If $\rho_g - \delta_g \geq \rho'$, then $\rho_h \geq \rho'$ for any point $h \in g$.*

Proof. By the definitions of referenced roundness and nearest and farthest neighbors, $\rho_h \in [\rho_g + \delta_g, \rho_g - \delta_g]$. ■

If we recursively subdivide the grids starting from a single grid, we may eliminate any grid with $\rho_g - \delta_g \geq \rho'$ where ρ' is the current minimum observed referenced roundness. In practice, this should significantly reduce the number of grid points tested. Again, the main difficulty becomes determining the order in which the grids are searched. We implemented two search methods. The first is a depth-first search that continuously evaluates and subdivides one particular grid completely until the maximum number of divisions is attained before proceeding to the next grid. The second is a breadth-first search that evaluates all grids at a particular division level and subdivides only valid grids before reevaluating.

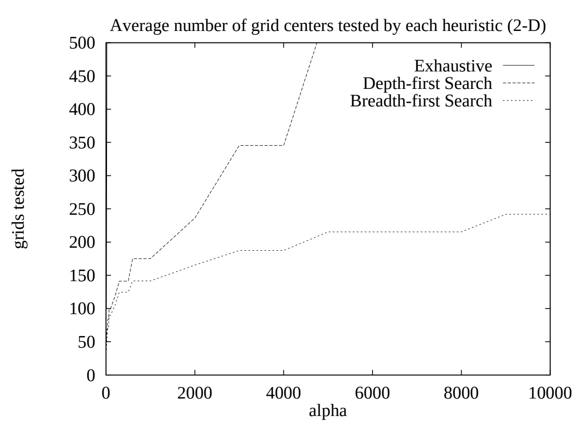

Figure 7: Comparing the grid-reduction heuristics to the standard mode. Each point on the graph is an average over 40 runs consisting of random orientation and thickness of an annulus of fixed radius with points ranging from 100 to 10,000 points in $\mathbb{R}^2$. Notice the immediate departure of the exhaustive method.

Figures 7 and 8 compare these two heuristics to the straightforward exhaustive search. Note that there is a vast reduction in the number of partitions tested in the breadth-first search method in comparison to the original method. It is interesting to note that the depth first search tended to improve as the number of points increased, possibly due to a reduction in potential centers.

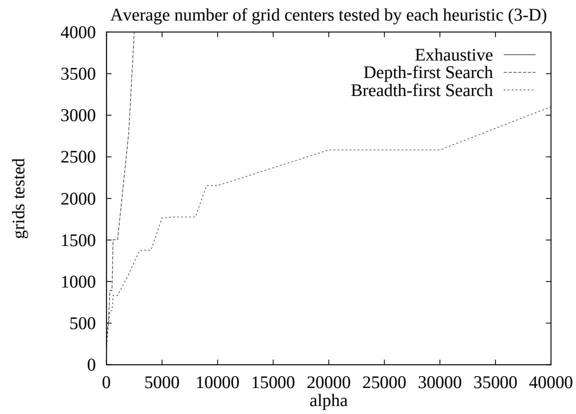

Figure 8: Similar to the previous figure with the points in $\mathbb{R}^3$. Again notice the immediate departure of the exhaustive and depth-first pruning method.

5 Open Problems

There are still many questions to be answered and many more yet to be asked.

- Can the bound on the referenced roundness problem in 3-space be improved?
- Are there any better non-trivial solutions to approximating the referenced roundness problem in higher dimensions, possibly by assuming the presence of "nearly round" objects?
- Can we tighten the error bound on the approximate width problem?
- Can we prove that any of the heuristics, under reasonable constraints, improve on the theoretical performances?
- What other solutions are there to such tolerancing problems as cylindricity, perpendicularity, and position?

References

[1] P. K. AGARWAL, B. ARONOV, AND M. SHARIR, *Computing envelopes in four dimensions with applications*, in Proc. 10th Annu. ACM Sympos. Comput. Geom., 1994, pp. 348–358.

[2] P. K. AGARWAL, J. J. MATOUŠEK, AND S. SURI, *Farthest neighbors, maximum spanning trees and related problems in higher dimensions*, Comput. Geom. Theory Appl., 1 (1992), pp. 189–201.

[3] P. K. Agarwal and M. Sharir, *Off-line dynamic maintenance of the width of a planar point set*, Comput. Geom. Theory Appl., 1 (1991), pp. 65–78.

[4] ———, *Planar geometric location problems and maintaining the width of a planar set*, Algorithmica, 11 (1994), pp. 185–195.

[5] ———, *Efficient randomized algorithms for some geometric optimization problems*, in Proc. 11th Annu. ACM Sympos. Comput. Geom., 1995, pp. 326–335.

[6] P. K. Agarwal, M. Sharir, and S. Toledo, *Applications of parametric searching in geometric optimization*, J. Algorithms, 17 (1994), pp. 292–318.

[7] S. Arya, D. M. Mount, N. S. Netanyahu, R. Silverman, and A. Wu, *An optimal algorithm for approximate nearest neighbor searching*, in Proc. 5th ACM-SIAM Sympos. Discrete Algorithms, 1994, pp. 573–582.

[8] B. Chazelle, *An optimal convex hull algorithm in any fixed dimension*, Discrete Comput. Geom., 10 (1993), pp. 377–409.

[9] B. Chazelle, H. Edelsbrunner, L. Guibas, and M. Sharir, *Diameter, width, closest line pair and parametric searching*, Discrete Comput. Geom., 10 (1993), pp. 183–196.

[10] K. L. Clarkson, *Linear programming in $O(n3^{d^2})$ time*, Inform. Process. Lett., 22 (1986), pp. 21–24.

[11] ———, *A Las Vegas algorithm for linear programming when the dimension is small*, in Proc. 29th Annu. IEEE Sympos. Found. Comput. Sci., 1988, pp. 452–456.

[12] R. Cole, *Slowing down sorting networks to obtain faster sorting algorithms*, J. ACM, 34 (1987), pp. 200–208.

[13] M. B. Dillencourt, D. M. Mount, and N. S. Netanyahu, *A randomized algorithm for slope selection*, Internat. J. Comput. Geom. Appl., 2 (1992), pp. 1–27.

[14] W. P. Dong, E. Mainsah, P. F. Sullivan, and K. F. Stout, *Instruments and measurement techniques of 3-dimensional surface topography*, in Three-Dimensional Surface Topography: Measurement, Interpretation and Applications, K. F. Stout, ed., Penton Press, Bristol, Penn., 1994.

[15] M. E. Dyer, *On a multidimensional search technique and its application to the Euclidean one-center problem*, SIAM J. Comput., 15 (1986), pp. 725–738.

[16] H. Ebara, N. Fukuyama, H. Nakano, and Y. Nakanishi, *Roundness algorithms using the Voronoi diagrams*, in Abstracts 1st Canad. Conf. Comput. Geom., 1989, p. 41.

[17] H. Edelsbrunner, *Algorithms in Combinatorial Geometry*, vol. 10 of EATCS Monographs on Theoretical Computer Science, Springer-Verlag, Heidelberg, West Germany, 1987.

[18] S. Fortune and C. J. Van Wyk, *Efficient exact arithmetic for computational geometry*, in Proc. 9th Annu. ACM Sympos. Comput. Geom., 1993, pp. 163–172.

[19] M. E. Houle and G. T. Toussaint, *Computing the width of a set*, in Proc. 1st Annu. ACM Sympos. Comput. Geom., 1985, pp. 1–7.

[20] ———, *Computing the width of a set*, IEEE Trans. Pattern Anal. Mach. Intell., PAMI-10 (1988), pp. 761–765.

[21] R. Janardan, *On maintaining the width and diameter of a planar point-set online*, Internat. J. Comput. Geom. Appl., 3 (1993), pp. 331–344.

[22] V. B. Le and D. T. Lee, *Out-of-roundness problem revisited*, IEEE Trans. Pattern Anal. Mach. Intell., PAMI-13 (1991), pp. 217–223.

[23] J. Matoušek, *Randomized optimal algorithm for slope selection*, Inform. Process. Lett., 39 (1991), pp. 183–187.

[24] P. McMullen, *The maximal number of faces of a convex polytope*, Mathematica, 17 (1970), pp. 179–184.

[25] N. Megiddo, *Applying parallel computation algorithms in the design of serial algorithms*, J. ACM, 30 (1983), pp. 852–865.

[26] ———, *Linear-time algorithms for linear programming in R^3 and related problems*, SIAM J. Comput., 12 (1983), pp. 759–776.

[27] ———, *Linear programming in linear time when the dimension is fixed*, J. ACM, 31 (1984), pp. 114–127.

[28] F. P. Preparata and M. I. Shamos, *Computational Geometry: An Introduction*, Springer-Verlag, New York, NY, 1985.

[29] A. A. G. Requicha, *Mathematical meaning and computational representation of tolerance specifications*, in Proc. 1993 Int. Forum on Dimensional Tolerancing and Metrology, 1993, pp. 61–68.

[30] G. Rote, C. Schwarz, and J. Snoeyink, *Maintaining the approximate width of a set of points in the plane*, in Proc. 5th Canad. Conf. Comput. Geom., Waterloo, Canada, 1993, pp. 258–263.

[31] R. Seidel, *Small-dimensional linear programming and convex hulls made easy*, Discrete Comput. Geom., 6 (1991), pp. 423–434.

[32] L. Shafer and W. Steiger, *Randomizing optimal geometric algorithms*, Technical Report 94-22, DIMACS, 1995.

[33] T. C. Shermer and C. K. Yap, *Probing for near-centers and estimating relative roundness*, in Proc. ASME Workshop on Tolerancing and Metrology, 1995.

[34] J. R. Shewchuk, *Robust adaptive floating-point geometric predicates*, in Proc. 12th Annu. ACM Sympos. Comput. Geom., 1996, pp. 141–150.

[35] M. Smid and R. Janardan, *On the width and roundness of a set of points in the plane*, in Proc. 7th Canad. Conf. Comput. Geom., 1995, pp. 193–198.

[36] V. Srinivasan, *Role of sweeps in tolerance semantics*, in Proc. 1993 Int. Forum on Dimensional Tolerancing and Metrology, 1993, pp. 69–78.

[37] K. Swanson, D. T. Lee, and V. L. Wu, *An optimal algorithm for roundness determination on convex polygons*, Computational Geometry: Theory and Applications, 5 (1995), pp. 225–235.

[38] R. K. Walker and V. Srinivasan, *Creation and evolution of the ASME Y14.5.1M standard*, in Proc. 1993 Int. Forum on Dimensional Tolerancing and Metrology, 1993, pp. 19–30.

[39] C. K. Yap, *Towards exact geometric computation*, in Proc. 5th Canadian Conference on Computational Geometry (CCCG), 1993, pp. 405–419.

[40] ———, *Exact computational geometry and tolerancing metrology*, in Snapshots of Computational and Discrete Geometry, Vol. 3, Tech. Rep. SOCS-94.50, D. Avis and J. Bose, eds., McGill School of Comp. Sci., 1995.