Approximation of Planar Convex Sets from Hyperplane Probes

T. J. Richardson

Bell Laboratories, Lucent Technologies,
Murray Hill, NJ 07974, USA
tjr@research.bell-labs.com

Abstract. This paper studies a geometric probing problem. Suppose that an unknown convex set in $\mathbb{R}^2$ can be probed by an oracle which, when given a unit vector, will return the position of the supporting hyperplane of the convex set that has the given vector as an outward normal. We present an on-line algorithm for choosing probing directions so that, after n probes, an inner and an outer estimate of the convex set are obtained that are within $O(n^{-2})$ of each other in Hausdorff distance. This is optimal since there exist convex sets that, even if visible, cannot be approximated better than $O(n^{-2})$ with n -sided polygons, for example, a circle.

1. Introduction

How accurately can an unknown convex set be approximated by sampling its support function? Equivalently, if one can probe a convex set to obtain a supporting hyperplane whose outward normal is in a specified direction, then how should one select the sample directions in order to obtain a good approximation to the convex set as fast as possible? A number of similar *geometric probing* problems have been treated in the literature [1], [4], [3], [8]–[10], [12]–[21]. A geometric probing problem is concerned with determining a shape, or some aspect of a shape, from probes. A probe can be thought of as an oracle who, when given a parameter, returns some property of the shape depending on the parameter. In our problem, the parameter is a direction, i.e., a unit vector in the plane, and the property returned is the location of the supporting hyperplane of a convex set C having the given direction as the outward normal. There are many variations on this theme depending on the type of probe and the shape to be determined. For a survey, we refer to Skiena [20], [21]. Geometric probing problems bear on problems arising in robotics, medical instrumentation, optimization, integral and computational geometry, and in graph theory.

The paper which initiated the field, by Cole and Yap [4], considered the problem of determining a convex polygon with an unknown number of vertices V from *finger probes*. A finger probe returns the point of contact between a specified inward directed ray and the convex set. The goal is to use as few probes as possible in order to determine the convex polygon C . Cole and Yap [4] give a strategy guaranteed to determine the set after using no more than $3V$ finger probes, and they prove that any strategy will require that many finger probes in the worst case. Bernstein [2] shows that $2V + k$ finger probes, where k is a small constant, is sufficient to identify one out of m given convex polygons. In the case of *hyperplane probes*, the case we consider, Li [13] gives a strategy requiring no more than $3V + 1$ probes and proved that this is optimal.

Most of the early problems appearing in the literature seek exact determination of a shape that can be specified by a finite number of parameters, e.g., polygons. Lindenbaum and Bruckstein [15] raised the question of instead finding an approximation of an unknown shape with respect to some metric. In particular, they considered the problem of approximating an arbitrary bounded convex set from hyperplane probes, with respect to the Hausdorff metric.

Given two sets A and B , the Hausdorff distance $d_{\mathcal{H}}(A, B)$ between A and B is defined as

$$\max \left\{ \sup_{x \in B} \text{dist}(x, A), \sup_{x \in A} \text{dist}(x, B) \right\},$$

where $\text{dist}(x, A) := \inf_{y \in A} \|x - y\|$. There is an extensive literature on approximation of convex sets by polygons with respect to the Hausdorff metric [3], [6], [11], [12]. The Hausdorff distance is a natural one to consider because of its simple geometric nature and also because it can be used to bound other natural distance measures, such as the area of the symmetric difference.

Lindenbaum and Bruckstein [15] showed, by considering a circle, that any algorithm which produces a convex approximation $\hat{C}$ to an unknown convex set C with $d_{\mathcal{H}}(C, \hat{C}) < \varepsilon$ must sometimes require $\Omega((\text{Length}(\partial C)/\varepsilon)^{1/2})$ probes. Here, $\text{Length}(\partial C)$ is the total length of the boundary of C . More precisely, they showed for the circle that, for small ε , one requires essentially

$$\sqrt{\frac{\pi}{2}} \left(\frac{\text{Length}(\partial C)}{\varepsilon} \right)^{1/2}$$

support-function values in order to obtain an approximation to within ε . They proposed and analyzed an algorithm that, after making $O((\text{Length}(\partial C)/\varepsilon)^{1/2} \log(\text{Length}(\partial C)/\varepsilon))$ hyperplane probes, is guaranteed to produce a convex approximation $\hat{C}$ to a convex set C satisfying $d_{\mathcal{H}}(\hat{C}, C) \leq \varepsilon$. They also proved this bound tight for their strategy, i.e., for some convex sets their algorithm requires this many probes. Thus, to improve on their result a new algorithm is required. In this paper, we close the gap by giving an algorithm guaranteed to produce an approximation $\hat{C}$ with $d_{\mathcal{H}}(C, \hat{C}) < \varepsilon$ using $O((\text{Length}(\partial C)/\varepsilon)^{1/2})$ probes.

We consider several variations of the basic problem. An *off-line* algorithm is given a prespecified error ε ; it makes a series of probes in order to guarantee an approximation meeting the prespecified requirement. One tries to minimize the number of probes required. An *on-line* algorithm is not given a prespecified error, but rather repeatedly

queries the oracle until at some unpredictable time it is stopped by an outside agent. In this case, one seeks an estimate on the quality of the approximation obtained as a function of the number of probes made, and one tries to maintain a good estimate at all times, as a function of the number of probes made.

Let C be a closed convex set in the plane and fix an origin 0 . For each $\theta \in \mathbb{S}^1$, define

$$u(\theta) := \sup\{\langle x, \theta \rangle : x \in C\}.$$

The function u is the *support* function of C . A convex body is the intersection of its supporting half-spaces, hence it can be recovered from u . A hyperplane probe returns the value of $u(\theta)$ for any selected θ . Thus, in some sense, the goal is to approximate u from its samples.

We will consider the following three problems.

Problem 1 (Point of Tangency Probe). Given $\theta \in \mathbb{S}^1$, an oracle returns some $p(\theta) \in \partial C$ satisfying $\langle p(\theta), \theta \rangle = u(\theta)$. Devise an on-line probing strategy to approximate C with respect to the Hausdorff metric.

Problem 2 (Off-line Hyperplane Probe). Given $\theta \in \mathbb{S}^1$, an oracle returns the value of $u(\theta)$ for an unknown convex set C . Devise an off-line probing strategy to approximate C with respect to the Hausdorff metric to within ε .

Problem 3 (On-line Hyperplane Probe). Given $\theta \in \mathbb{S}^1$, an oracle returns the value of $u(\theta)$ for an unknown convex set C . Devise an on-line probing strategy to approximate C with respect to the Hausdorff metric.

We are primarily interested in the last problem, which is the hardest. We discuss the other two problems in order to introduce, in a simpler setting, ideas needed for the solution to Problem 3.

For Problem 1, we give an algorithm that requires (asymptotically)

$$\frac{3}{2} \sqrt{\frac{\pi}{2}} \left(\frac{\text{Length}(\partial C)}{\varepsilon} \right)^{1/2}$$

probes to guarantee an approximation to within ε . We conjecture that $\frac{3}{2}\sqrt{\pi/2}$ is in fact the optimal constant for this problem. For Problems 2 and 3, we require essentially twice the number of probes indicated for Problem 1. Proving optimality of the constant in this case seems much harder. In any case, the order as a function of $\text{Length}(\partial C)$ and ε is optimal, and the constants are close to optimal.

We state our results in the form of bounds on the Hausdorff distance between C_{inner} and C_{outer} , both convex estimates of C which satisfy $C_{\text{inner}} \subset C \subset C_{\text{outer}}$. This is for convenience, and gives a bound for the general case, because Lindenbaum and Bruckstein [15] showed that by splitting the difference between C_{inner} and C_{outer} one can obtain an estimate $\hat{C}$ satisfying

$$d_{\mathcal{H}}(\hat{C}, C) \leq \frac{1}{2} d_{\mathcal{H}}(C_{\text{inner}}, C_{\text{outer}}).$$

One of the main ideas in our strategy for Problems 2 and 3 is to probe in closely spaced pairs of directions to obtain good estimates of boundary points of C . Purely geometric considerations show that one obtains good estimates this way, but there is another motivation which is worth describing. Let $\mathbb{S}^1$ be oriented so that counterclockwise rotations are positive. Let $\star\theta$ denote the unit vector obtained by rotating θ through 90 degrees. Let C be a closed convex set and let $P(\theta) := \{x \in \partial C: \langle x, \theta \rangle = u(\theta)\}$ be the support set in direction θ . If $P(\theta)$ is a singleton, then we denote its element by $p(\theta)$. In general, we define $p(\theta-)$ to be the $x \in P(\theta)$ that minimizes $\langle x, \star\theta \rangle$ and $p(\theta+)$ to be the $x \in P(\theta)$ that maximizes $\langle x, \star\theta \rangle$. Note that $p(\theta-)$ is the first element of $P(\theta)$ encountered in a counterclockwise traversal of the boundary of C , and $p(\theta+)$ is the last element.

Theorem 1.1. *For any closed convex set C , the function $u(\theta)$ is absolutely continuous and it has left and right derivatives $\dot{u}(\theta-)$, $\dot{u}(\theta+)$ for all θ . Furthermore*

$$\begin{aligned} p(\theta-) &= u(\theta)\theta + \dot{u}(\theta-) \star\theta, \\ p(\theta+) &= u(\theta)\theta + \dot{u}(\theta+) \star\theta. \end{aligned}$$

Proof. We omit a proof. See [17] or [18].

By sampling u in pairs $\theta \pm \eta$, for small η , we can obtain a good estimate for both $u(\theta)$ and $\dot{u}(\theta)$. In view of Theorem 1.1, this means that we can obtain a good estimate of $p(\theta)$. A solution to Problem 1 is then fairly easily transformed into a solution to Problem 2. A solution to Problem 3 uses this idea also, but requires more work.

The probing problem is also connected to questions of metric entropy [3], [6]. Bronštein [3] showed that the ε -entropy, with respect to Hausdorff distance, of convex sets which are contained the unit ball in d -dimensions ($d \geq 2$) has order $\varepsilon^{(1-d)/2}$. This means that one can find approximately $2^{\varepsilon^{(1-d)/2}}$ convex sets in the unit disc that are mutually ε apart with respect to Hausdorff distance. The heart of the proof of the ε -entropy bound is the construction of polygonal approximations of convex sets. A simple and elegant construction appears in Dudley [6]. For dimension $d = 2$ this is the same as the number of probes required to determine the set to within ε . In view of the results obtained in this paper, this means that well-chosen hyperplane probes typically yield a constant number of bits of information concerning the unknown convex set.

The ideas in this paper generalize to higher dimensions. For a higher-dimensional result, one could probably combine the construction of polygonal approximations to C found in [6] with a multiple probing strategy (to mimic the effect of tangent point information) in a straightforward way to obtain an ε approximation to C after using $O(\varepsilon^{(1-d)/2} \log(1/\varepsilon))$ hyperplane probes. However, we conjecture that $O(\varepsilon^{(1-d)/2})$ probes are enough.

2. Approximation Estimates and Preliminaries

Let us fix a convex set C and let u denote its support function. Assume that u has been sampled in a cyclically ordered set of directions $\phi_i, i = 1, \dots, n$. An *outer estimate* C_{outer}

Fig. 1. Inner and outer estimates of a convex set.

of the convex set C is obtained by intersecting the associated supporting half-spaces. An *inner estimate* can be defined as the intersection of all closed convex sets with the given support lines. (Later, for Problem 1, we will define the inner estimate a little differently.) It can be explicitly constructed as follows. Let x_i denote the intersection point of the two support lines having outer normals φ_i and φ_{i+1} (subscripts are mod n). To each i , we associate a half-space H_i as follows. If x_{i-1} , x_i , and x_{i+1} are not collinear, then let H_i be the half-space not containing x_i whose boundary is the line through x_{i-1} and x_{i+1} . If x_{i-1} , x_i , and x_{i+1} are collinear, then $x_{i-1} = x_i = x_{i+1}$ and, in this degenerate case, we simply let H_i be the half-space with outer normal φ_i such that x_i is on the boundary of H_i . The inner estimate is precisely $C_{\text{inner}} := \cap_{i=1}^n H_i$. See Fig. 1.

Let d_i denote the Euclidean distance between x_i and H_i . If $|\langle \varphi_i, \varphi_{i+1} \rangle| \leq \pi/2$ for each i , then $d_{\mathcal{H}}(C_{\text{inner}}, C_{\text{outer}})$ is equal to the maximum d_i . (Here $|\langle \theta_1, \theta_2 \rangle|$ denotes the angle between the unit vectors θ_1 and θ_2 . When $|\langle \theta_1, \theta_2 \rangle| < \pi$, we will understand (θ_1, θ_2) to mean the set of unit vectors in the positive cone generated by θ_1 and θ_2 , i.e., the set of unit vectors lying on the shortest closed arc segment in $\mathbb{S}^1$ joining θ_1 with θ_2 .) To obtain bounds on d_i , we will use the following estimate.

Lemma 2.1. *Let a , b , and c be the lengths of the sides of a nondegenerate triangle. Let v_a , v_b , v_c be the vertices opposite their respective sides and let A_e be the exterior angle at v_a . Let d be the perpendicular height from v_a to the line through v_b and v_c . Then*

$$d \leq \frac{b+c}{2} \sin\left(\frac{A_e}{2}\right).$$

Proof. Let B and C denote the interior angles at v_b and v_c , respectively (see Fig. 2). We will prove the inequality by minimizing the ratio $(b+c)/d$ while holding A_e fixed. This ratio is equal to $(\sin(B) + \sin(A_e - B))^{-1}$. Differentiating with respect to B we

Fig. 2. Triangle notation.

obtain

$$\frac{\partial}{\partial B} \frac{b+c}{d} = \frac{\cos(A_e - B) - \cos(B)}{(\sin(B) + \sin(A_e - B))^2} = \frac{\cos(C) - \cos(B)}{(\sin(B) + \sin(C))^2}.$$

Thus, if $B < C$, then the ratio decreases by increasing B . We see that the minimum occurs at $B = C$. From this the lemma follows directly. $\square$

The following is another useful inequality.

Lemma 2.2. *Let a, b, c be the side lengths of a triangle and let A_e be the exterior angle opposite the side with length a . Then*

$$a \geq (b+c) \cos(A_e/2).$$

Proof. The inequality is a simple consequence of the cosine rule.

$$\begin{aligned} a^2 &= b^2 + c^2 + 2bc \cos(A_e) \\ &= (b+c)^2 \cos^2(A_e/2) + (b-c)^2 (1 - \cos^2(A_e/2)) \\ &\geq (b+c)^2 \cos^2(A_e/2). \end{aligned}$$

$\square$

Lemma 2.1 states the key geometric inequality used in the proofs of the performance guarantees of our algorithms. The second key ingredient is the following simple algebraic inequality which will be used in an inductive argument.

Lemma 2.3. *If $a_i b_i \geq \gamma c_i^2$ for $i = 1, 2$, where all quantities are nonnegative reals, then $(a_1 + a_2)(b_1 + b_2) \geq \gamma(c_1 + c_2)^2$.*

Proof. First note that

$$a_1 b_2 + a_2 b_1 \geq 2\sqrt{a_1 b_2 a_2 b_1}$$

by the Schwartz inequality. Using this, we obtain

$$\begin{aligned} (a_1 + a_2)(b_1 + b_2) &\geq (\sqrt{a_1 b_1} + \sqrt{a_2 b_2})^2 \\ &\geq \gamma(c_1 + c_2)^2. \end{aligned}$$

$\square$

Another geometric estimate we require is a relation between the Hausdorff distance of convex sets $C_1 \subset C_2$ and the difference of the length of their boundaries.

Lemma 2.4. *Let C_1, C_2 be bounded convex sets in the plane such that $C_1 \subset C_2$. Then $\text{Length}(\partial C_1) \leq \text{Length}(\partial C_2)$ and*

$$\text{Length}(\partial C_1) + 2\pi d_{\mathcal{H}}(C_1, C_2) \geq \text{Length}(\partial C_2). \quad (1)$$

Proof. Let u_1 and u_2 be the support functions of C_1 and C_2 , respectively. Crofton's formula [19] gives the following

$$\text{Length}(\partial C_i) = \int_0^\pi |u_i(\theta) - u_i(-\theta)| d\theta.$$

Thus, $\text{Length}(\partial C_1) \leq \text{Length}(\partial C_2)$ is immediate. Now, $|u_2(\theta) - u_2(-\theta)| - |u_1(\theta) - u_1(-\theta)| \leq 2d_{\mathcal{H}}(C_1, C_2)$ so we directly obtain (1). $\square$

3. Point of Tangency Problem

In this section, we present a solution to Problem 1. In this problem, we assume that the oracle gives us not merely the location of the supporting hyperplane, but rather a point from the boundary of the set that is contained in the supporting hyperplane. Thus, given $\theta \in \mathbb{S}^1$, the oracle returns a point $p(\theta) \in \partial C$ which satisfies $u(\theta) = \langle p(\theta), \theta \rangle$. Note, by this equation, that $p(\theta)$ also gives us the value of $u(\theta)$.

Initial samples are taken at four uniformly spaced directions yielding a bounding box for C . Let S^n denote the set of directions which have been sampled after the n th stage of the algorithm. In this section, n will also denote the number of sampled directions; thus we begin with $n = 4$. For this section only (Problem 1), we let C_{inner}^n denote the convex hull of the set $\{p(\varphi) : \varphi \in S^n\}$. As always, C_{outer}^n denotes the intersection of the known supporting half-spaces after stage n of the algorithm. We have $C_{\text{inner}}^n \subset C \subset C_{\text{outer}}^n$ for each n .

For each $\varphi \in S^n$, let $\varphi_{+n} \in S^n$ denote the sample direction succeeding φ in a counterclockwise cyclic ordering of S^n . Let x_φ^n denote the point of intersection between the support lines with outer normals φ and φ_{+n} . Define

$$L_\varphi^n := |p(\varphi) - x_\varphi^n| + |p(\varphi_{+n}) - x_\varphi^n|.$$

Note that $\sum_{\varphi \in S^n} L_\varphi^n = \text{Length}(\partial C_{\text{outer}}^n)$. Using Lemma 2.1 and noting $\sin(x) \leq x$ for $x \in [0, \pi/2]$ we have

$$4d_{\mathcal{H}}(C_{\text{inner}}^n, C_{\text{outer}}^n) \leq \max_{\varphi \in S^n} |(\varphi, \varphi_{+n})| L_\varphi^n. \quad (2)$$

To simplify the notation, we let h_n denote the quantity on the right in (2). We are now ready to give a formal definition of the algorithm.

Algorithm 1. Let $S^4 := \theta_0 + \{0, \pi/2, \pi, 3\pi/2\}^1$ (θ_0 is arbitrary) and let initial probes be made in the directions of S^4 .

Given S^n we generate S^{n+1} as follows.

Find $\hat{\varphi}^n \in S^n$ such that $h_n = |\langle \hat{\varphi}^n, \hat{\varphi}_{+n}^n \rangle| L_{\hat{\varphi}^n}^n$. Let φ_m denote the midpoint of $(\hat{\varphi}^n, \hat{\varphi}_{+n}^n)$. Make the $(n+1)$ th probe in direction φ_m and define $S^{n+1} := S^n \cup \varphi_m$.

Note that h_n is nonincreasing in n . If $h_n = 0$, then the algorithm has found C exactly. In this case, all of our performance claims become trivial. It simplifies the proofs to assume $h_n > 0$ for all n , hence, without loss of generality, we do so. Using Algorithm 1, we have the following performance guarantee.

Theorem 3.1. For $n \geq 10$, Algorithm 1 guarantees the estimate

$$d_{\mathcal{H}}(C_{\text{inner}}^n, C_{\text{outer}}^n) \leq \frac{9\pi}{4} \frac{\text{Length}(\partial C)}{n^2 - 9\pi^2}.$$

Remark. It is possible, with some tedious analysis, to obtain bounds for $n \in 4, \dots, 10$ and to tighten the bound for small $n \geq 10$.

The proof of Theorem 3.1 is inductive and the remainder of this section is devoted to it. To facilitate the proof, we represent the evolution of the algorithm using a nested sequence of binary trees $\{G^n\}$. The leaves of G^n will correspond with the intervals (φ, φ_{+n}) between sampled directions at the n th stage of the algorithm. When the midpoint of an interval is sampled the corresponding leaf generates two children corresponding with the two subintervals, thereby defining G^{n+1} . Thus, at any stage, the number of leaves in the tree is exactly equal to the number of samples which have been taken.

To each node p we associate two quantities both independent of n ; L_p , which denotes a length, and Φ_p , which denotes the angular interval associated to p .

Definition of Directed Graph Representation. Let $G^4 = (V^4, E^4)$ be the complete directed binary tree of depth 3. Its vertex set consists of six nodes

$$V^4 := \{\text{root}, t_1, t_2, q_1, r_1, q_2, r_2\}.$$

Its directed edge set is given by

$$E^4 := \{(\text{root}, t_i), (t_i, q_i), (t_i, r_i): i = 1, 2\}.$$

We will first define L_p and Φ_p for $p \in V^4$. (See Fig. 3.) From the four initial probes, we obtain a $l_1 \times l_2$ bounding box for C where $l_1 \geq l_2$. If $l_2 = 0$, then C is determined by the four initial samples, and $h_4 = 0$, hence, by our assumption on h_n , we have $l_2 > 0$. Let $\theta_0, \theta_0 + \pi$ be the outer normal directions associated with the sides of length l_2 . We then set $\Phi_{t_1} := (\theta_0, \theta_0 + \pi)$, $\Phi_{q_1} := (\theta_0, \theta_0 + \pi/2)$, and

¹ This is an abuse of notation to which we adhere. For a unit vector θ and a real number η the expression $\theta + \eta$ denotes the unit vector obtained by rotating θ through η radians.

 Fig. 3. Initial bounding box for algorithm 1 and graph G^4 .

$\Phi_{r_1} := (\theta_0 + \pi/2, \theta_0 + \pi)$. The intervals associated with t_2, q_2, r_2 are obtained by rotating those associated with t_1, q_1, r_1 , respectively, by π . For completeness we set $\Phi_{\text{root}} := \mathbb{S}^1$. We define $L_{\text{root}} = L_{t_1} = L_{t_2} := +\infty$, and $L_{q_1} := L_{\theta_0}^4$, $L_{r_1} := L_{\theta_0 + \pi/2}^4$, etc.

Given G^n , we generate G^{n+1} as follows. (See Fig. 4.) There is a unique leaf (by induction) p in V^n such that $\Phi_p = (\hat{\varphi}^n, \hat{\varphi}_{+n}^n)$. Introduce two new nodes q and r , i.e., set $V^{n+1} = V^n \cup \{q, r\}$, and set $E^{n+1} = E^n \cup \{(p, q), (p, r)\}$. This completes the definition of G^{n+1} . Corresponding to the new nodes, we define $L_q := L_{\hat{\varphi}^n}^{n+1}$ and $L_r := L_{\varphi_m}^{n+1}$, and also $\Phi_q := (\hat{\varphi}^n, \varphi_m)$ and $\Phi_r := (\varphi_m, \hat{\varphi}_{+n}^n)$. Note that if $(p, q) \in E^n$, then $L_p \geq L_q$ and $|\Phi_p| = 2|\Phi_q|$.

For each $n \geq 4$ let $\mathcal{T}^n$ denote the set of terminal nodes in V^n . For any node $p \in V^n$, let N_p^n denote the number of terminal nodes in the subtree of G^n rooted at p . Thus, if $p \in \mathcal{T}^n$, then $N_p^n = 1$. For each node $p \in V^n \setminus (\mathcal{T}^n \cup \{\text{root}, t_1, t_2\})$ let n_p be the maximal m such that $p \in \mathcal{T}^m$. This is the n corresponding to the stage of the algorithm in which

Fig. 4. Graph growth in Algorithm 1.

the interval associated with p was selected for splitting. Note that if n_p is defined, then $L_p = L_{\hat{\varphi}^n_p}^{n_p}$.

Lemma 3.2 (Algorithm 1). *For $n \geq 10$, we have $N_{t_1}^n, N_{t_2}^n \geq 3$.*

Proof. Because of our assignment of intervals to nodes in G^4 , we have $L_{q_i} + L_{r_i} \geq l_1$ for $i = 1, 2$, hence $\max\{h_{q_i}, h_{r_i}\} \geq \pi l_1/16$ for $i = 1, 2$. Assume $n = 9$ and $N_{t_1}^n = 2$. We will show that if s is a leaf in the subtree rooted at t_2 , then $h_s < \pi l_1/16$. This will prove the lemma.

Let u, v be the children either of q_2 or of r_2 . It is easy to see that $L_u, L_v < l_1 + l_2$, hence $h_u, h_v < \pi l_1/8$ since $|\Phi_u| = |\Phi_v| = \pi/4$. Thus, if w is a child of u or v , i.e., a grandchild of q_2 or r_2 , then $h_w < \pi l_1/16$. Since $\max\{h_{q_1}, h_{r_1}\} \geq \pi l_1/16$, there can be no great grandchildren of q_2 or r_2 . Therefore, at $n = 9$, both q_2 and r_2 must have children and three of these must in turn have children. Thus, it is enough to show that there is one child s of q_2 and r_2 which satisfies $h_s < \pi l_1/16$, or equivalently, that $L_s < l_1$ (since $|\Phi_s| = \pi/4$). But this follows from the fact that the sum of L_s over the four children of q_2 and r_2 is bounded above by $L_{q_2} + L_{r_2} \leq 2l_2 + l_1 \leq 3l_1$. $\square$

Fix $n \geq 10$ and consider the tree G^n . We now recursively define a functional $\hat{L}_p^n$ on the nodes p of G^n . The functional $\hat{L}_p^n$ is, roughly speaking, the length of the portion of $\partial C_{\text{outer}}^n$ associated with the subtree under p .

Definition of Length Estimate. If p is a terminal node of G^n , i.e., $N_p^n = 1$, then set $\hat{L}_p^n := L_p$. If p is any nonterminal node in G^n , then let q and r denote its children. If $N_p^n = 2$, then we set $\hat{L}_p^n := \max(\hat{L}_q^n + \hat{L}_r^n, h_n/|\Phi_p|)$. If $N_p^n > 2$, then we recursively set $\hat{L}_p^n := \hat{L}_q^n + \hat{L}_r^n$.

Lemma 3.3 (Algorithm 1). *For any $n \geq 10$, the following hold.*

- A. If $N_p^n = 2$, then $|\Phi_p| \hat{L}_p^n \geq h_n$.
- B. If $N_p^n \geq 2$, then $|\Phi_p| \hat{L}_p^n \geq \frac{2}{9} h_n (N_p^n)^2$.

Proof. Condition A follows by definition. Note also that A implies B for $N_p^n = 2$. Let q, r denote the children of p and assume $N_p^n = 3$. Then, modulo renaming the children, we have $N_r^n = 1$ and $N_q^n = 2$. Hence,

$$\frac{2}{9}h_n(N_p^n)^2 = \frac{1}{2}h_n(N_q^n)^2 \leq 2|\Phi_q|\hat{L}_q^n \leq |\Phi_p|\hat{L}_q^n \leq |\Phi_p|\hat{L}_p^n.$$

Thus B holds for $N_p^n \leq 3$.

Assume now that B holds for all nodes s such that $N_s^n \leq m$ where $m \geq 3$, and that $N_p^n = m + 1$. Without loss of generality, we assume $N_q^n \geq 2$. If $N_r^n \geq 2$, then B follows directly from Lemma 2.3. If $N_r^n = 1$, then $N_q^n = m \geq 3$ and we have $(N_p^n)^2 = (N_q^n + 1)^2 < 2(N_q^n)^2$. Therefore,

$$\frac{2}{9}h_n(N_p^n)^2 \leq \frac{2}{9}h_n(2(N_q^n)^2) \leq 2|\Phi_q|\hat{L}_q^n \leq |\Phi_p|\hat{L}_p^n,$$

completing the proof. $\square$

Since $N_{\text{root}}^n = n$ and $|\Phi_{\text{root}}| = 2\pi$, when $n \geq 10$ we now have

$$2\pi\hat{L}_{\text{root}}^n \geq \frac{2}{9}h_n n^2. \quad (3)$$

Now, $\hat{L}_{\text{root}}^n$ is nearly equal to $\text{Length}(\partial C)$.

Lemma 3.4. *For $n \geq 10$ we have $\hat{L}_{\text{root}}^n \leq \pi h_n + \text{Length}(\partial C)$.*

Before proving this we complete the proof of Theorem 3.1. From Lemma 3.4 and (3) we obtain

$$2\pi(\pi h_n + \text{Length}(\partial C)) \geq \frac{2}{9}h_n n^2,$$

or

$$\text{Length}(\partial C) \geq \left(\frac{1}{9\pi}n^2 - \pi \right) h_n.$$

Substituting the bound $4d_{\mathcal{H}}(C_{\text{inner}}^n, C_{\text{outer}}^n) \leq h_n$, obtained from (2), into the equation above and rearranging terms, we obtain Theorem 3.1.

We will now prove Lemma 3.4.

Proof of Lemma 3.4. For each p satisfying $N_p^n = 2$ define $\tilde{L}_p^n := \hat{L}_q^n + \hat{L}_r^n$, where q and r are the children of p . Let $V_2^n \subset V^n$ denote the set of nodes p satisfying $N_p^n = 2$ and also $\hat{L}_p^n > \tilde{L}_p^n$.

From the definition of $\hat{L}_p^n$, we have

$$\hat{L}_{\text{root}}^n = \sum_{p \in V_2^n} (\hat{L}_p^n - \tilde{L}_p^n) + \sum_{p \in T^n} L_p.$$

To bound the second term, we use Lemma 2.4 to obtain

$$\sum_{p \in T^n} L_p = \text{Length}(\partial C_{\text{outer}}^n) \leq \text{Length}(\partial C) + \frac{1}{2}\pi h_n. \quad (4)$$

From Lemma 2.2 we obtain $|p(\hat{\varphi}^{n_p}) - p(\hat{\varphi}_{+n_p}^{n_p})| \geq \cos(|\Phi_p|/2)L_p$. Since h_n is non-increasing we have $|\Phi_p|L_p \geq h_n$, so if $p \in V_2^n$, then $L_p \geq \hat{L}_p^n$. Hence,

$$\cos\left(\frac{|\Phi_p|}{2}\right)\hat{L}_p^n \leq \tilde{L}_p^n.$$

If $x \leq \pi/4$, then the inequality $(\cos(x))^{-1} - 1 \leq x^2$ holds. Since $n \geq 10$, we conclude from Lemma 3.2 that if $p \in V_2^n$, then $|\Phi_p| \leq \pi/2$. We can now proceed

$$\sum_{p \in V_2^n} \hat{L}_p^n - \tilde{L}_p^n \leq \sum_{p \in V_2^n} \left(\frac{|\Phi_p|}{2}\right)^2 \tilde{L}_p^n \leq \sum_{p \in V_2^n} \frac{1}{4} |\Phi_p| h_n \leq \frac{\pi}{2} h_n, \quad (5)$$

where for the penultimate inequality we used the fact that if $p \in V_2^n$, then $|\Phi_p| \tilde{L}_p^n \leq h_n$. $\square$

4. Hyperplane Probe with Prespecified Errors

We now consider Problem 2: hyperplane probing without tangent point information, but with prespecified approximation requirements. As before, the convex set C is initially probed in four uniformly spaced directions yielding a $l_1 \times l_2$, $l_1 \geq l_2 > 0$, bounding box for C . This gives us an upper bound $D := \sqrt{l_1^2 + l_2^2}$ on the diameter of C . Let ε denote the desired approximation accuracy. Define

$$\eta := \frac{\varepsilon}{10D}.$$

The main innovation introduced to compensate for the lack of knowledge of a boundary point in C is to sample the support function in closely spaced pairs of directions. The intersection point of the pair of support lines approximates a boundary point of C . Initial samples will be taken at angles $\{0 \pm \eta, \pi/2 \pm \eta, \pi \pm \eta, 3\pi/2 \pm \eta\}$. The bounding box described above can first be taken from the $\theta - \eta$ directions.

As before, we let S^n represent the set of sampled directions. The difference now is that $\varphi \in S^n$ means the directions $\varphi \pm \eta$ have been sampled by stage n . Let φ_{+n} have the same meaning as in Problem 1, i.e., the element of S^n subsequent to φ in a counterclockwise cyclic ordering of S^n .

Definition of p_φ . In Problem 1, we were given a point $p(\varphi)$; here we define p_φ to be the point of intersection between the support lines with outer normal directions $\varphi - \eta$ and $\varphi + \eta$. We now define x_φ^n to be the intersection point of the supporting lines with outer normal directions $\varphi + \eta$ and $\varphi_{+n} - \eta$. As before, we set $L_\varphi^n := |p_\varphi - x_\varphi^n| + |p_{\varphi_{+n}} - x_\varphi^n|$ and define

$$h_n := \max_{\varphi \in S^n} |(\varphi, \varphi_{+n})| L_\varphi^n.$$

See Fig. 5.

Fig. 5. Initial bounding box for Algorithm 2.

Algorithm 2. Let $S^4 := \theta_0 + \{0, \pi/2, \pi, 3\pi/2\}$ and let initial probes be made in the directions of $S^4 \pm \eta$.

Given S^n , we generate S^{n+1} as follows.

Find $\hat{\varphi}^n \in S^n$ such that $h_n = |(\hat{\varphi}^n, \hat{\varphi}_{+n}^n)|L_{\hat{\varphi}^n}^n$. Let φ_m denote the midpoint of $(\hat{\varphi}^n, \hat{\varphi}_{+n}^n)$. Make the $(2n + 1)$ th and the $(2n + 2)$ th probes in directions $\varphi_m \pm \eta$ and define $S^{n+1} := S^n \cup \varphi_m$.

Note that Algorithm 2 is identical to Algorithm 1 except for the pairwise sampling of the support function. The performance guarantee is also similar.

Theorem 4.1. *Algorithm 2 guarantees that if $n \geq 10$ and*

$$n \geq \frac{3\sqrt{\pi}}{2} \left(\frac{\text{Length}(\partial C)}{\varepsilon} + 2\pi \right)^{1/2},$$

then $d_{\mathcal{H}}(C_{\text{inner}}^n, C_{\text{outer}}^n) \leq \varepsilon$.

Remark. Note that, here, $2n$ is the number of probes actually made by stage n .

The proof of this theorem is very similar to the proof of Theorem 3.1. We will indicate only the new considerations which arise in this case.

It is not obvious that (2) still holds, but, in a restricted sense, it does.

Lemma 4.2 (Algorithm 2). *If $h_n \geq 4\varepsilon$, then $4d_{\mathcal{H}}(C_{\text{inner}}^n, C_{\text{outer}}^n) \leq h_n$.*

Proof. Let

$$\tilde{h}_n := \max_{\varphi \in S^n} (2\eta)(|p_{\varphi_{+n}} - x_{\varphi}^n| + |p_{\varphi_{+n}} - x_{\varphi_{+n}}^n|).$$

We have

$$4d_{\mathcal{H}}(C_{\text{inner}}^n, C_{\text{outer}}^n) \leq \max\{h_n, \tilde{h}_n\},$$

thus, we need only show $h_n \geq \tilde{h}_n$.

Since $L_\varphi^n \leq \sqrt{2}D$, we see that if $|(\varphi, \varphi_{+n})|L_\varphi^n \geq 4\varepsilon$, then $|(\varphi, \varphi_{+n})| \geq 4\varepsilon/\sqrt{2}D$. It follows that if $h_n \geq 4\varepsilon$, then for all $\varphi \in S^n$ we have

$$|(\varphi, \varphi_{+n})| \geq \frac{2\varepsilon}{\sqrt{2}D} > 10\eta. \quad (6)$$

Thus, we see that if $h_n \geq 4\varepsilon$, then the sample directions $\varphi \in S^n$ are spaced by significantly more than the gaps between sample pairs $\varphi \pm \eta$. Now,

$$\begin{aligned} (2\eta)(|p_{\varphi_{+n}} - x_\varphi^n| + |p_{\varphi_{+n}} - x_{\varphi_{+n}}^n|) &\leq (2\eta)(L_\varphi^n + L_{\varphi_{+n}}^n) \\ &\leq \frac{1}{5}(|(\varphi, \varphi_{+n})|L_\varphi^n + |(\varphi_{+n}, \varphi_{+n+n})|L_{\varphi_{+n}}^n) \\ &\leq h_n. \end{aligned}$$

Taking the maximum over φ , we obtain $h_n \geq \tilde{h}_n$. $\square$

We define the graph G^n exactly as for Problem 1 except for the rotation by $-\eta$ of the original bounding box directions. We also define N_p^n , L_p , and n_p as before. The new definition of Φ_p ignores η so the definition appears as before. Thus, if n_p is defined, then we have $\Phi_p = (\hat{\varphi}^{n_p}, \hat{\varphi}_{+n_p}^{n_p})$. The same basic properties hold. In particular, corresponding to Lemma 3.2 we have:

Lemma 4.3 (Algorithm 2). *For $n \geq 10$, we have $N_{t_1}^n, N_{t_2}^n \geq 3$.*

Proof. The proof given for Lemma 3.2 goes through verbatim with the new definitions. $\square$

As in Problem 1, we now recursively define a functional $\hat{L}_p^n$ on the nodes p of G^n . The definition is exactly the same as in Problem 1 so we will not repeat it here.

Lemma 4.4. *If $n \geq 10$ and $h_n \geq \varepsilon$, then the following hold.*

- A. *If $N_p^n = 2$, then $|\Phi_p|\hat{L}_p^n \geq h_n$.*
- B. *If $N_p^n \geq 2$, then $|\Phi_p|\hat{L}_p^n \geq \frac{2}{9}h_n(N_p^n)^2$.*

Proof. The proof for Lemma 3.3 goes through verbatim except that we invoke Lemma 4.2 to obtain the bound corresponding to (2). $\square$

Theorem 4.1 now follows directly using the same argument used to prove Theorem 3.1, with one small correction. Whereas before we had $|p(\hat{\varphi}^{n_p}) - p(\hat{\varphi}_{+n_p}^{n_p})| \geq \cos(|\Phi_p|/2)L_p$ from Lemma 2.2, we now have $|p_{\hat{\varphi}^{n_p}} - p_{\hat{\varphi}_{+n_p}^{n_p}}| \geq \cos((|\Phi_p| - 2\eta)/2)L_p$. However, we still obtain the estimate

$$(\cos(|\Phi_p|/2 - \eta))^{-1} - 1 \leq (|\Phi_p|/2)^2. \quad (7)$$

as long as $h_n \geq 4\varepsilon$. The inequality $\sum_{p \in V_2^n} (\hat{L}_p^n - \tilde{L}_p^n) \leq \frac{1}{2}\pi h_n$ is obtained as in (5), and the remainder of the argument goes through as before.

5. On-line Hyperplane Probing

In the final version, Problem 3, we do not have a prespecified error bound. The algorithm given for Problem 2 is inadequate here because the prior bound on the error associated with the interval $|(\varphi - \eta, \varphi + \eta)|$ will be too large if the algorithm runs long enough. Another way of expressing this is to say that the intervals $|(\varphi, \varphi_{+n})|$ will eventually become small compared to η .

The idea we introduce to handle this problem is to allow the distance between sample pairs to depend on the size of the interval being split. This will guarantee that, for each interval, one of the endpoint pairs has a small gap compared to the interval size. However, this does not provide such a guarantee for both endpoints. If one of the endpoint pairs has a large gap compared to the size of the interval, then we sample the support function in two directions so that this relatively widely spaced pair is converted into two pairs with small gaps. These additional samples do not come with an attendant guaranteed improvement in the approximation. There is a potential here for the number of samples required to balloon. Most of the difficulty of the proof is concerned with showing that this does not occur.

At the n th stage of the algorithm, the convex set C will have been probed in a collection of pairs of directions $\varphi \pm \eta$, where, now, η depends on φ . We will refer to η as the *offset* associated with φ . As before, we let S^n denote the collection of such φ corresponding to the n th stage of the algorithm. For any fixed direction $\varphi \in S^n$, there will be only one associated offset which will be independent of n . However, unlike before, S^n will not be strictly monotonically increasing in n . This is because, occasionally, it will be necessary to replace a pair $\varphi \pm \eta$ with two pairs $\varphi_1 \pm \eta_1, \varphi_2 \pm \eta_2$, where $\varphi - \eta = \varphi_1 - \eta_1$ and $\varphi + \eta = \varphi_2 + \eta_2$. Such a replacement requires only two new probes and no information is thrown away. However, φ is removed from S^n and $\{\varphi_1, \varphi_2\}$ is added when such a replacement occurs. At any stage of the algorithm, $|S^n| = n$ is half the number of probes required by the algorithm to complete the n th stage.

Let φ denote an element of S^n and let φ_{+n} denote the subsequent element in a cyclic ordering, as before. As a notational convenience, we will use η to denote the offset associated with φ wherever it occurs and η_{+n} to denote the offset associated with φ_{+n} . As in Problem 2, we define p_φ to be the point of intersection between the support line with outer normal direction $\varphi - \eta$ and that with outer normal direction $\varphi + \eta$. Similarly, we define x_φ^n to be the intersection point of the supporting lines with outer normal directions $\varphi + \eta$ and $\varphi_{+n} - \eta_{+n}$. We also define the following as before

$$\begin{aligned} L_\varphi^n &:= |x_\varphi^n - p_\varphi| + |x_{\varphi_{+n}}^n - p_{\varphi_{+n}}|, \\ h_n &:= \max_{\varphi \in S^n} |(\varphi, \varphi_{+n})| L_\varphi^n. \end{aligned}$$

As in Problem 2, we make initial probes in four uniformly spaced directions, giving a $l_1 \times l_2, l_1 \geq l_2 > 0$, bounding box for C .

The algorithm requires two constants α, β which satisfy $\alpha \geq 4$ and $\beta \geq \alpha + 2$. We define another constant γ by $2^{-\gamma} = 2^{1-\alpha} + 2^{-\beta}$. Note that $\gamma > 2$. We are now ready to describe the algorithm.

Algorithm 3. Initial probes are made for $\varphi \in \{0, \pi/2, \pi, 3\pi/2\}$ with offsets $\eta = 2^{-\beta}$ in all cases. Thus $S^4 = \{0, \pi/2, \pi, 3\pi/2\}$.

Given S^n , we perform the following steps to produce S^{n+1} .

Find $\hat{\varphi}^n \in S^n$ such that $h_n = |(\hat{\varphi}^n, \hat{\varphi}_{+n}^n)|L_{\hat{\varphi}^n}^n$. Let $\hat{\eta}$ and $\hat{\eta}_{+n}$ denote the offsets associated with $\hat{\varphi}^n$ and $\hat{\varphi}_{+n}^n$, respectively. If $|(\hat{\varphi}^n, \hat{\varphi}_{+n}^n)| > 2^\alpha \min\{\hat{\eta}, \hat{\eta}_{+n}\}$, then execute Step 1 only, otherwise execute Step 2 only.

Step 1. Probe C in the directions $\varphi_m \pm \eta_m$, where

$$\begin{aligned}\varphi_m &:= \text{midpoint of } (\hat{\varphi}^n, \hat{\varphi}_{+n}^n), \\ \eta_m &:= 2^{-(1+\beta)}|(\hat{\varphi}^n, \hat{\varphi}_{+n}^n)|.\end{aligned}$$

Define $S^{n+1} := S^n \cup \varphi_m$ and let η_m be the offset associated with φ_m .

Step 2. Set $\tilde{\eta} := 2^{-\beta}(1 + 2^{1-\beta})^{-1} \min\{\hat{\eta}, \hat{\eta}_{+n}\}$. For convenience assume $\hat{\eta} = \min\{\hat{\eta}, \hat{\eta}_{+n}\}$. (We will prove later that we cannot have $\hat{\eta} = \hat{\eta}_{+n}$ at this point.) Define $\varphi_a := \hat{\varphi}^n - \hat{\eta} + \tilde{\eta}$ and $\varphi_b := \hat{\varphi}^n + \hat{\eta} - \tilde{\eta}$. Probe C in the directions $\varphi_a + \tilde{\eta}$ and $\varphi_b - \tilde{\eta}$. Define

$$S^{n+1} := (S^n - \hat{\varphi}^n) \cup \{\varphi_a, \varphi_b\},$$

where $\tilde{\eta}$ is the offset associated with both φ_a and φ_b . (If $\hat{\eta}_{+n} = \min\{\hat{\eta}, \hat{\eta}_{+n}\}$, then carry out the analogous procedure for $\hat{\varphi}_{+n}^n, \hat{\eta}_{+n}$.)

Remark. The choice of $\tilde{\eta}$ is so that $\tilde{\eta} = 2^{-\beta}|(\varphi_a, \varphi_b)|$, see Lemma A.3.

The guaranteed performance for this algorithm is given by the following.

Theorem 5.1. *For any $\delta > 0$, we can choose α and β so that for $n \geq 10$ Algorithm 3 guarantees*

$$d_{\mathcal{H}}(C_{\text{inner}}^n, C_{\text{outer}}^n) \leq \frac{9\pi}{4} \frac{\text{Length}(\partial C)}{(1 - \delta)n^2 - 9\pi^2}.$$

The proof of this theorem parallels that of Theorem 4.1, but it is more delicate. Here we require certain intermediate results which control the effect of Step 2. We have relegated the proofs of these intermediate results to the Appendix to avoid distracting the reader from the parallelism with the proof of Theorem 4.1.

Lemma 5.2. *The following hold for all n and φ in S^n :*

$$\min\{\eta, \eta_{+n}\} \leq 2^{-\beta}|(\varphi, \varphi_{+n})|, \quad (8)$$

$$\eta + \eta_{+n} \leq 2^{-\gamma}|(\varphi, \varphi_{+n})|. \quad (9)$$

Proof. See the Appendix.

Again, we need a proof that h_n gives us the correct estimate for the Hausdorff distance between the inner and outer estimate.

Lemma 5.3 (Algorithm 3). *For $n \geq 4$, the following estimate holds:*

$$4d_{\mathcal{H}}(C_{\text{inner}}^n, C_{\text{outer}}^n) \leq h_n.$$

Proof. The proof closely parallels that of Lemma 4.2. Let

$$\tilde{h}_n := \max_{\varphi \in S^n} (2\eta_{+n})(|p_{\varphi_{+n}} - x_\varphi^n| + |p_{\varphi_{+n}} - x_{\varphi_{+n}}^n|).$$

We have $4d_{\mathcal{H}}(C_{\text{inner}}^n, C_{\text{outer}}^n) \leq \max\{h_{\text{max}}^n, \tilde{h}_{\text{max}}^n\}$ so we need only show $h_{\text{max}}^n \geq \tilde{h}_{\text{max}}^n$. Now,

$$\begin{aligned} (2\eta_{+n})(|p_{\varphi_{+n}} - x_\varphi^n| + |p_{\varphi_{+n}} - x_{\varphi_{+n}}^n|) &\leq (2\eta_{+n})(L_\varphi^n + L_{\varphi_{+n}}^n) \\ &\leq 2^{-(\gamma-1)}(|(\varphi, \varphi_{+n})|L_\varphi^n + |(\varphi_{+n}, \varphi_{+n})|L_{\varphi_{+n}}^n) \\ &\leq h_n, \end{aligned}$$

where the second to last inequality follows from Lemma 5.2. $\square$

As before, we represent evolution of the algorithm by a labeled acyclic directed graph. Nodes correspond to intervals demarked by the algorithm. The main difference here is that the graph need not be a tree; some nodes may have two parents. Each node of the graph will be either *type 1* or *type 2*, corresponding to the step in the algorithm that produced it. At the n th stage of the algorithm, the terminal nodes of the corresponding graph are in one-to-one correspondence with the arcs (φ, φ_{+n}) associated with S^n . Initially, the graph G^4 has six nodes, exactly as before, all of which are now type 1. Step 1 of the algorithm splits an interval, and we represent this in the graph as before, by the associated node spawning two children. A new node produced this way is a *type 1* node. See Fig. 6. Step 2 of the algorithm, on the other hand, generates three new nodes which share two parents. The reason for this is that the introduction of the new interval in between what was an endpoint pair of samples affects both of the intervals that shared that endpoint pair. The node associated with the new interval is a *type 2* node. The nodes associated with the neighboring intervals inherit their type from their immediate ancestor. (At least one of these must be type 1, but we do not need this.) See Fig. 7.

Given a node p in the graph, Φ_p denotes the interval associated with p , as before. With each interval, and hence with each node, we can associate the values of the offsets η corresponding to the two endpoints. We use η_p to denote the offset associated to the “left” endpoint φ . We now proceed with the formal definitions.

Fig. 6. Graph growth due to Step 1.

Fig. 7. Graph growth due to Step 2.

Initialization. Corresponding to S^4 , we define the directed graph $G^4 = (V^4, E^4)$. The definition is identical to that in Problem 2. All nodes in V^4 are type 1.

Step 1. Let p be the (terminal) node of V^n such that $\Phi_p = (\hat{\varphi}^n, \hat{\varphi}_{+n}^n)$. Define $V^{n+1} := V^n \cup \{q, r\}$, where both new nodes are type 1, and define $E^{n+1} := E^n \cup \{(p, q), (p, r)\}$. Define $\Phi_q := (\hat{\varphi}^n, \varphi_m)$ and $\Phi_r := (\varphi_m, \hat{\varphi}_{+n}^n)$.

Step 2. For convenience, assume $\hat{\eta} = \min\{\hat{\eta}, \hat{\eta}_{+n}\}$; the construction when $\hat{\eta}_{+n} = \min\{\hat{\eta}, \hat{\eta}_{+n}\}$ is analogous. Let $(\hat{\varphi}_{-n}^n, \hat{\varphi}^n)$ denote the interval adjacent to $(\hat{\varphi}^n, \hat{\varphi}_{+n}^n)$ as determined by S^n . Let s be the node of V^n satisfying $\Phi_s = (\hat{\varphi}_{-n}^n, \hat{\varphi}^n)$. Define $V^{n+1} := V^n \cup \{u, v, w\}$. Node u is given the same type as node s , node v is type 2, and node w is given the same type as node p (which must, in fact, be type 1). Set $\Phi_u := (\hat{\varphi}_{-n}^n, \varphi_a)$, $\Phi_v := (\varphi_a, \varphi_b)$, and $\Phi_w := (\varphi_b, \hat{\varphi}_{+n}^n)$. Define

$$E^{n+1} := E^n \cup \{(s, u), (s, v), (p, v), (p, w)\}.$$

Note that node v has two parents. Only type 2 nodes can have two parents. The quantity $|S^n| = n$ is equal to the number of terminal nodes in V^n . In Problems 1 and 2, all terminal nodes are type 1. A key step to proving the performance bound for Algorithm 3 is to show that the number of type 2 terminal nodes is small compared to the number of type 1 terminal nodes. Let $N_{1,p}^n$ denote the number of type 1 terminal nodes in V^n that can be reached from node p , and let $N_{2,p}^n$ denote the corresponding number of type 2 terminal nodes. (A node q can be *reached* from a node p in G^n if there exists a sequence

of edges $(p, r_1), (r_1, r_2), \dots, (r_k, q)$ in E^n .) As before, n_p , when it exists, denotes the maximal value of n for which $p \in \mathcal{T}^n$. Every node $p \neq \text{root}$ has either one or two *siblings*, these being in general the nodes born at the same stage as p . The initial nodes are a special case: t_1 and t_2 are siblings, as are r_1 and q_1 , and also r_2 and q_2 .

Let L_p be defined as in Problem 2. Note that if $(p, q) \in E^n$, then $L_p \geq L_q$. Let $\mathcal{T}_1^n$ and $\mathcal{T}_2^n$ be the type 1 and 2 terminal nodes, respectively. Then $|\mathcal{T}_1^n| = N_{1, \text{root}}^n$ and $|\mathcal{T}_2^n| = N_{2, \text{root}}^n$.

Corresponding to Lemmas 3.2 and 4.3 we have the following.

Lemma 5.4 (Algorithm 3). *For $n \geq 10$, we have $N_{1, t_1}^n, N_{1, t_2}^n \geq 3$.*

Proof. As in the proof of Lemma 4.3, we can transpose verbatim the proof of Lemma 3.2 to obtain a proof here. One point which deserves mention is that because $\beta \geq \alpha + 2$ and $\pi/2 > 1$, any type 2 node in G^n will be at depth at least 6 from the root. Thus, the possibility of type 2 nodes affecting the argument is eliminated. $\square$

As in Problems 1 and 2, the main arguments use induction over the graph G^n . Since type 2 nodes can have two parents, we need to modify the definitions. To this end, we introduce weights for edges in E^n .

Definition of Edge Weights. For each edge $(p, q) \in E^n$, we define $w_{p,q}$ as follows. If q has only one parent, then $w_{p,q} = 1$. If q has two parents and $N_{1,q}^n > 0$, or if $L_q = 0$, then set $w_{p,q} = \frac{1}{2}$. If q has two parents and $N_{1,q}^n = 0$ and $L_q > 0$, then let p_1 and p_2 be the parents of q , and let q_1 and q_2 be the siblings of q sharing parent p_1 and p_2 , respectively. Set

$$w_{p_i,q} := \frac{L_{p_i} - L_{q_i}}{L_{p_1} - L_{q_1} + L_{p_2} - L_{q_2}} \quad (10)$$

for $i = 1, 2$. (Note that since $L_q > 0$, the denominator is strictly positive.)

The weights w have been chosen first to satisfy $w_{p,q} \geq 0$, but also so that for each $q \neq \text{root}$ we have $\sum_{p: (p,q) \in E^n} w_{p,q} = 1$. We will use the weights to recursively define the function $\hat{L}^n$. The special definition given in (10) attempts to maintain $L_p \simeq \hat{L}_p^n$ when $N_{1,p}^n < 2$, see Lemma 8.

Definition of Length Estimate. If p is a terminal node, we set $\hat{L}_p^n := L_p$. If $N_{1,p}^n = 2$ and $N_{1,q}^n = 1$ for any child q of p , then we set $\hat{L}_p^n := \max(\sum_{\{q: (p,q) \in E^n\}} w_{p,q} \hat{L}_q^n, h_n/|\Phi_p|)$. Otherwise we recursively set $\hat{L}_p^n := \sum_{\{q: (p,q) \in E^n\}} w_{p,q} \hat{L}_q^n$.

Denote by V_2^n those nodes $p \in V^n$ such that $N_{1,p}^n = 2$, $N_{1,q}^n < 2$ for any child q of p , and $\sum_{\{q: (p,q) \in E^n\}} w_{p,q} \hat{L}_q^n < h_n/|\Phi_p|$. Note that $p \in V_2^n$ guarantees that p has two type 1 children.

Type 1 Terminal Node Counting Function. If p is a terminal node, then define $\hat{N}_p :=$

$N_{1,p}^n$. If p is not a terminal node, then we define

$$\hat{N}_p := \sum_{\{q: (p,q) \in E\}} w_{p,q} \hat{N}_q.$$

Note that for $q \neq \text{root}$, we have $\sum_{\{p: (p,q) \in E\}} w_{p,q} = 1$. Thus,

$$\hat{N}_{\text{root}} = |\mathcal{T}_1^n|.$$

Lemma 5.5. *If $(p, q) \in E^n$ and $N_{1,q}^n > 0$, then $|\Phi_p| \hat{L}_p^n \geq |\Phi_q| \hat{L}_q^n$.*

Proof. If p is the lone parent of q , then $w_{p,q} = 1$. It follows that $\hat{L}_p^n \geq \hat{L}_q^n$ (even if $p \in V_2^n$) and since $|\Phi_p| \geq |\Phi_q|$ the inequality follows.

If q has two parents, then it is type 2. Let p_1 and p_2 be the parents in this case. Since $N_{1,q}^n > 0$, we have $w_{p_1,q} = w_{p_2,q} = \frac{1}{2}$, hence $\hat{L}_{p_1}^n, \hat{L}_{p_2}^n \geq \hat{L}_q^n/2$. However, since q is type 2 with two parents, we have $|\Phi_{p_1}|, |\Phi_{p_2}| \geq 2^{\nu-1} |\Phi_q| \geq 2 |\Phi_q|$ by Lemma 5.2. Hence, $|\Phi_{p_1}| \hat{L}_{p_1}^n, |\Phi_{p_2}| \hat{L}_{p_2}^n \geq |\Phi_q| \hat{L}_q^n$. $\square$

Let 1_p be the indicator function that p is type 1, i.e.,

$$1_p := \begin{cases} 0, & p \text{ is type 2,} \\ 1, & p \text{ is type 1.} \end{cases}$$

Note that $\hat{N}_p \geq 1_p$.

Lemma 5.6. *If $p \in V^n$, then*

$$\hat{N}_p > 1_p \quad \Rightarrow \quad |\Phi_p| \hat{L}_p^n \geq \frac{2}{9} \hat{N}_p^2 h_n. \quad (11)$$

Proof. We will prove (11) using induction over G^n . However, to make the induction go through we append the following ancillary induction hypotheses:

$$\hat{N}_p > 2 \quad \Rightarrow \quad |\Phi_p| \hat{L}_p^n \geq \frac{25}{18} h_n. \quad (12)$$

Note that (11) and (12) hold trivially for all terminal nodes.

We first prove the following implication:

$$\hat{N}_p > 1_p \quad \Rightarrow \quad |\Phi_p| \hat{L}_p^n \geq h_n. \quad (13)$$

The condition $\hat{N}_p > 1_p$ implies $N_{1,p}^n \geq 2$, as is easily verified. If $N_{1,p}^n = 2$ and $N_{1,q}^n = N_{1,r}^n = 1$ where q and r are the children of p , then we have $|\Phi_p| \hat{L}_p^n \geq h_n$ directly. The general case now follows from Lemma 5.5. Note that (13) implies (12) and (11) for $\hat{N}_p \leq 2$. Henceforth we assume $\hat{N}_p > 2$. In particular, this implies $p \notin V_2^n$.

Let q and r denote the children of p . We assume that (11) and (12) hold for q and r . If $|\Phi_q| \hat{L}_q^n \geq \frac{2}{9} \hat{N}_q^2 h_n$ and $|\Phi_r| \hat{L}_r^n \geq \frac{2}{9} \hat{N}_r^2 h_n$, then we obtain $|\Phi_p| \hat{L}_p^n \geq \frac{2}{9} \hat{N}_p^2 h_n$, by

applying Lemma 2.3. Thus, we need only verify (11) when either $\hat{N}_q = 1_q$ or $\hat{N}_r = 1_r$. We proceed by cases.

Case 1: Both children of node p are type 1. If $\hat{N}_q > 1_q$ and $\hat{N}_r > 1_r$, then since $|\Phi_p| = 2|\Phi_r| = 2|\Phi_q|$, we obtain $|\Phi_p|\hat{L}_p^n \geq 4h_n$ from (13). Thus, (12) follows directly.

Henceforth we assume $\hat{N}_q = 1$. Since we assume $\hat{N}_p > 2$, it follows that $\hat{N}_r > 1$. By (13) we have $|\Phi_p|\hat{L}_p^n \geq 2h_n$. We see that (12) holds in this case and (11) holds for $\hat{N}_p \leq 3$. Assuming $\hat{N}_p > 3$ we have $\hat{N}_r > 2$ hence, using the induction hypothesis (12) for r , we obtain $|\Phi_p|\hat{L}_p^n \geq \frac{25}{9}h_n$. We see that (11) holds for $\hat{N}_p \leq 3.5$. Assuming $\hat{N}_p > 3.5$ we have $\hat{N}_r > 2.5$ and $\hat{N}_p^2 = (\hat{N}_q + \hat{N}_r)^2 < 2\hat{N}_r^2$. Hence, $|\Phi_p|\hat{L}_p^n \geq 2|\Phi_r|\hat{L}_r^n \geq \frac{4}{9}\hat{N}_r^2h_n \geq \frac{2}{9}\hat{N}_p^2h_n$.

Case 2: At least one child of node p is type 2. Without loss of generality, we assume that r is the (type 2) child of p with two parents. It follows from Lemma 5.2 that $|\Phi_p| \geq 2^{\gamma-1}|\Phi_r| \geq 2|\Phi_r|$.

Assume $\hat{N}_r > 0$, then $w_{p,r} = \frac{1}{2}$ and we obtain

$$|\Phi_p|\hat{L}_p^n \geq |\Phi_r|\hat{L}_r^n + |\Phi_q|\hat{L}_q^n. \quad (14)$$

If $\hat{N}_q = 1_q$, then $\hat{N}_p = 1_q + \frac{1}{2}\hat{N}_r$ hence $\hat{N}_r > 2$, but this implies $\hat{N}_p < \hat{N}_r$ and (12) and (11) follow trivially from the induction hypothesis. Assuming $\hat{N}_q > 1_q$ we obtain $|\Phi_p|\hat{L}_p^n \geq 2h_n$ from (13), and (12) follows.

If $\hat{N}_r = 0$, then $\hat{N}_p = \hat{N}_q$. But $|\Phi_p| \geq |\Phi_q|$ and $\hat{L}_p^n \geq \hat{L}_q^n$, so this case is trivial. $\square$

We are now ready to complete the proof of Theorem 5.1.

Proof of Theorem 5.1. Since $\hat{N}_{\text{root}} = N_{1,\text{root}}^n = |\mathcal{T}_1^n|$, assuming $n \geq 10$, we now have

$$2\pi\hat{L}_{\text{root}}^n \geq \frac{2}{9}|\mathcal{T}_1^n|^2h_n. \quad (15)$$

To proceed, we need to estimate $\hat{L}_{\text{root}}^n$ as before, but we also need to relate n to $|\mathcal{T}_1^n|$. For the latter, we prove the following in the Appendix.

Lemma 5.7. *For each n , we have*

$$|\mathcal{T}_1^n| \geq (\beta - \alpha - 1)|\mathcal{T}_2^n|.$$

Since $n = |\mathcal{T}^n| = |\mathcal{T}_1^n| + |\mathcal{T}_2^n|$, we have $n \leq (1 + (\beta - \alpha - 1)^{-1})|\mathcal{T}_1^n|$.

To estimate $\hat{L}_{\text{root}}^n$, we can proceed much as in Problem 2. We again have the following

$$\hat{L}_{\text{root}}^n = \sum_{p \in V_2^n} (\hat{L}_p^n - \tilde{L}_p^n) + \sum_{p \in \mathcal{T}^n} L_p.$$

As before, we need to bound $\hat{L}_p^n - \tilde{L}_p^n$ in the equation above. We require the following estimate whose proof is in the Appendix.

Lemma 5.8. *If $N_{1,q}^n < 2$, then $\hat{L}_q^n \geq (1 + 2^{-\beta-1}|\Phi_q|^2)^{-1}L_q$.*

Assume that $p \in V_2^n$ and let q and r be the two type 1 children of p . Then $|\Phi_p| = 2|\Phi_r| = 2|\Phi_q|$ so, by Lemma 5.8,

$$\begin{aligned}\tilde{L}_p^n &= \hat{L}_q^n + \hat{L}_r^n \\ &\geq (1 + 2^{-\beta-3}|\Phi_p|^2)^{-1}(L_q + L_r).\end{aligned}$$

We now apply Lemma 2.2 to obtain

$$\tilde{L}_p^n \geq (1 + 2^{-\beta-3}|\Phi_p|^2)^{-1} \cos\left(\frac{|\Phi_p|}{2}\right) L_p.$$

Assuming $n \geq 10$ we have $|\Phi_p| \leq \pi/2$ by Lemma 5.4, hence

$$\frac{(1 + 2^{-\beta-1}(|\Phi_p|/2)^2)}{\cos(|\Phi_p|/2)} - 1 \leq \left(\frac{|\Phi_p|}{2}\right)^2.$$

Since $p \in V_2^n$ and h_n is monotonic in n we have $\hat{L}_p^n \leq L_p$, hence

$$\frac{\hat{L}_p^n}{\tilde{L}_p^n} - 1 \leq \frac{L_p}{\tilde{L}_p^n} - 1 \leq \left(\frac{|\Phi_p|}{2}\right)^2.$$

We now proceed as in (5) to obtain $\sum_{p \in V_2^n} (\hat{L}_p^n - \tilde{L}_p^n) \leq \frac{1}{2}\pi h_n$. We also have $\sum_{p \in \mathcal{T}^n} L_p = \text{Length}(\partial C_{\text{outer}}^n) \leq \text{Length}(\partial C) + \frac{1}{2}\pi h_n$, hence,

$$\hat{L}_{\text{root}}^n \leq \pi h_n + \text{Length}(\partial C).$$

Substituting into (15), we obtain

$$(\pi h_n + \text{Length}(\partial C)) \geq \frac{1}{9\pi} |\mathcal{T}_1^n|^2 h_n$$

or, rearranging terms,

$$\text{Length}(\partial C) \geq \left(\frac{1}{9\pi} |\mathcal{T}_1^n|^2 - \pi\right) h_n.$$

Finally, substituting from Lemma 5.7, we have

$$\text{Length}(\partial C) \geq \left(\frac{1}{9\pi} \left(1 + \frac{1}{\beta - \alpha - 1}\right)^2 n^2 - \pi\right) h_n.$$

Since $(1 + 1/(\beta - \alpha - 1))^2$ can be made arbitrarily close to 1 by making $\beta - \alpha$ large enough, the proof is complete. $\square$

Acknowledgments

I would like to thank Professor Alfred Bruckstein for introducing me to probing problems and also Professor Michael Lindenbaum for probing discussions. I would also like to thank Professor Richard Dudley for providing me with a preprint from his forthcoming book *Uniform Central Limit Theorems* [7] which contains a rewriting of the work by Bronštein [3].

Appendix

Proof of Lemma 5.2. The proof is by induction. The inequalities are easily verified for $n = 4$. Assume they are satisfied for $m < n$, where $n > 4$.

Assume Step 1 of the algorithm is invoked at the n th stage. We see from the definition of η_m that $\eta_m = 2^{-\beta} |(\hat{\varphi}^n, \varphi_m)| = 2^{-\beta} |(\varphi_m, \hat{\varphi}_{+n}^n)|$ so (8) is satisfied. Since Step 1 was invoked, we have $\max(\hat{\eta}, \hat{\eta}_{+n}) \leq 2^{-\alpha} |(\hat{\varphi}^n, \hat{\varphi}_{+n}^n)|$, hence $\hat{\eta} + \eta_m \leq (2^{-\beta} + 2^{1-\alpha}) |(\hat{\varphi}^n, \varphi_m)|$ and $\hat{\eta}_{+n} + \eta_m \leq (2^{-\beta} + 2^{1-\alpha}) |(\varphi_m, \hat{\varphi}_{+n}^n)|$, completing the proof for this case.

Assume now that Step 2 of the algorithm is invoked at the n th stage. We see from the definition of $\tilde{\eta}$ that $\tilde{\eta} = 2^{-\beta} |(\varphi_a, \varphi_b)|$, so both inequalities follow easily for this interval.

We now consider the interval $|(\varphi_b, \hat{\varphi}_{+n}^n)|$. We have $|(\hat{\varphi}^n, \hat{\varphi}_{+n}^n)| \geq 2^\gamma \hat{\eta} \geq 3\hat{\eta}$ from the inductive hypothesis. Since $2\hat{\eta} > |(\varphi_a, \varphi_b)|$, we obtain $|(\varphi_b, \hat{\varphi}_{+n}^n)| \geq |(\hat{\varphi}^n, \hat{\varphi}_{+n}^n)| - \hat{\eta} > |(\varphi_a, \varphi_b)|$. Thus, $\tilde{\eta} \leq 2^{-\beta} |(\varphi_b, \hat{\varphi}_{+n}^n)|$. Now,

$$\begin{aligned} \hat{\eta}_{+n} + \tilde{\eta} &= \hat{\eta}_{+n} + \hat{\eta} - (\hat{\eta} - \tilde{\eta}) \\ &\leq 2^{-\gamma} |(\hat{\varphi}^n, \hat{\varphi}_{+n}^n)| - (\hat{\eta} - \tilde{\eta}) \\ &\leq 2^{-\gamma} (|(\hat{\varphi}^n, \hat{\varphi}_{+n}^n)| - (\hat{\eta} - \tilde{\eta})) \\ &= 2^{-\gamma} |(\varphi_b, \hat{\varphi}_{+n}^n)|, \end{aligned}$$

completing the proof for this interval.

The interval $|(\hat{\varphi}_{-n}^n, \varphi_a)|$ can be handled analogously. □

Proof of Lemma 5.7 (Counting Type 2 Terminal Nodes). For any n , a terminal node p is associated with an interval $\Phi_p = (\varphi, \varphi_{+n})$. Let $\eta^l(p)$ denote the offset associated with the “left” endpoint φ and let $\eta^r(p)$ denote the offset associated with the “right” endpoint φ_{+n} . Consider the following functions defined on these nodes:

$$\begin{aligned} R^l(p) &:= \log \left(\frac{|\Phi_p|}{\eta^l(p)} \right), \\ R^r(p) &:= \log \left(\frac{|\Phi_p|}{\eta^r(p)} \right). \end{aligned}$$

By examining the two steps of the algorithm, we can verify the following relations, where we have used the notation from the description of Algorithm 3. In the construction, we assume that $\eta^l(p) = \min\{\eta^l(p), \eta^r(p)\}$ for Step 2, the other case is analogous.

Step 1.

$$R^l(q) = R^l(p) - 1, \quad (16)$$

$$R^r(r) = R^r(p) - 1, \quad (17)$$

$$R^l(r) = R^r(q) = \beta. \quad (18)$$

Step 2.

$$R^l(u) \geq R^l(s) - 2^{1-\gamma}, \quad (19)$$

$$R^r(u) \geq R^r(s) + 2^{1-\gamma}, \quad (20)$$

$$R^l(v) \geq \beta, \quad (21)$$

$$R^r(v) \geq \beta, \quad (22)$$

$$R^l(w) \geq \beta - 1 + 2^{1-\gamma}, \quad (23)$$

$$R^r(w) \geq R^r(p) - 2^{1-\gamma}. \quad (24)$$

Most of these are obvious and several of the inequalities have been purposefully made weak. We will prove the stronger inequalities (19) and (24). To obtain (19), we use $|\Phi_u| \geq |\Phi_s| - \eta^r(s)$ to derive

$$\log\left(\frac{|\Phi_u|}{|\Phi_s|}\right) \geq \log(1 - 2^{-\gamma}) \geq -2^{1-\gamma}.$$

To obtain (24), we use $|\Phi_w| \geq |\Phi_p| - \eta^l(p)$ in the same way to derive

$$\log\left(\frac{|\Phi_w|}{|\Phi_p|}\right) \geq -2^{1-\gamma}.$$

Now, we recursively define functions $\hat{R}^r, \hat{R}^l$. First, if s is a terminal node corresponding to $n = 4$, i.e., one of the initial terminal nodes, then we set $\hat{R}^l(s) := \beta - 1$ and $\hat{R}^r(s) := \beta$. For subsequent terminal nodes, we define $\hat{R}^r$ and $\hat{R}^l$ inductively by setting the inequalities in Step 2 above, i.e., in (19)–(24), to equalities and using these and the equalities in Step 1, (16)–(18), as definitions. It follows by induction that $\hat{R}^r \leq R^r$ and $\hat{R}^l \leq R^l$. We also see that if p is a type 1 terminal node, then $\hat{R}^r(p) + \hat{R}^l(p) \leq 2\beta - 1$ and, if p is a type 2 terminal node, then $\hat{R}^r(p) + \hat{R}^l(p) \leq 2\beta$. Let $W^n := \sum_{p \in \mathcal{T}^n} (\hat{R}^r(p) + \hat{R}^l(p))$. For each $n \geq 4$, we have

$$W^n \leq |\mathcal{T}_1^n|(2\beta - 1) + |\mathcal{T}_2^n|(2\beta).$$

We claim that the following inequality is also valid:

$$|\mathcal{T}_1^n|(2\beta - 2) + |\mathcal{T}_2^n|(3\beta - \alpha - 1) \leq W^n. \quad (25)$$

The lemma then follows easily from the preceding inequalities.

The proof of the claim is by induction. It obviously holds for $n = 4$. Assume it holds for $n \leq m$ where $m \geq 4$. Let $p \in \mathcal{T}^m$ be given by $\Phi_p = (\hat{\varphi}^m, \hat{\varphi}_{+m}^m)$.

We assume first that Step 1 is invoked at the m th stage of the algorithm. It follows then that $W^{m+1} - W^m = 2\beta - 2$. If p is type 1, then $|\mathcal{T}_1^{m+1}| = |\mathcal{T}_1^m| + 1$ and $|\mathcal{T}_2^{m+1}| = |\mathcal{T}_2^m|$

and (25) follows. If p is type 2, then $|T_1^{m+1}| = |T_1^m| + 2$ and $|T_2^{m+1}| = |T_2^m| - 1$, and, since $(4\beta - 4) - (3\beta - \alpha - 1) = \beta + \alpha - 3 \leq 2\beta - 2$, (25) again follows.

We assume now that Step 2 is invoked at the m th stage of the algorithm. In this case, we have $|T_1^{m+1}| = |T_1^m|$ and $|T_2^{m+1}| = |T_2^m| + 1$. From the definitions of $\hat{R}^r$ and $\hat{R}^l$ we obtain $W^{m+1} - W^m = 3\beta - 1 - \hat{R}^l(p)$. Since Step 2 was invoked, we have $R^l(p) \leq \alpha$ and, since $\hat{R}^l(p) \leq R^l(p)$, we have $W^{m+1} - W^m \geq 3\beta - 1 - \alpha$. The proof is complete. $\square$

Proof of Lemma 5.8. We will prove the following stronger version of Lemma 5.8.

Lemma A.1. *If $N_{1,p}^n < 2$, then $L_p \leq (1 + 2^{-1-\beta}(|\Phi_p|)^2)\hat{L}_p^n$ and if, furthermore, p is type 2, then $L_p \leq (1 + 2^{2-3\beta}(|\Phi_p|)^2)\hat{L}_p^n$.*

The proof of this lemma is based on the following inequality:

Lemma A.2. *Consider the triangle as in Lemma 2.1, see Fig. 2. Consider also the triangle with side lengths a, b', c' having corresponding angles $A + 2\varepsilon, B - \varepsilon$, and $C - \varepsilon$, where $\varepsilon < \min\{B, C\}$. The following inequality holds:*

$$b + c \leq (1 + \varepsilon \tan(A_e/2))(b' + c').$$

Proof. By the sine rule, we have

$$b' + c' = a \frac{\sin(B - \varepsilon) + \sin(C - \varepsilon)}{\sin(A + 2\varepsilon)}.$$

Note that setting $\varepsilon = 0$ gives a formula for $b + c$. The first step in the proof is to show the surprising fact that, for fixed a and A , the ratio $(b + c)/(b' + c')$ is independent of B . We use the identity $\sin(u) + \sin(v) = 2 \sin((u + v)/2) \cos((u - v)/2)$ to obtain

$$\begin{aligned} \frac{b + c}{b' + c'} &= \frac{\sin(B) + \sin(C)}{\sin(B - \varepsilon) + \sin(C - \varepsilon)} \frac{\sin(A + 2\varepsilon)}{\sin(A)} \\ &= \frac{\sin(A_e/2)}{\sin(A_e/2 - \varepsilon)} \frac{\sin(A_e - 2\varepsilon)}{\sin(A_e)} \\ &= \frac{\cos(A_e/2 - \varepsilon)}{\cos(A_e)}. \end{aligned}$$

Calculus proves the inequality $\cos(A_e/2 - \varepsilon)/\cos(A_e/2) \leq 1 + \tan(A_e/2)\varepsilon$, completing the proof. $\square$

Lemma A.2 yields the following inequality corresponding to Step 2 of the algorithm, see the denominator in (5.10).

$$L_p - L_w + L_s - L_v \leq (1 + \tan(|\Phi_v|/2)2\tilde{\eta})L_v. \quad (26)$$

The following will also be helpful.

Lemma A.3. *If q is a type 2 node with two parents p_1 and p_2 , where p_1 is type 2, then $|\Phi_{p_1}| \geq 2^{\beta-1}|\Phi_q|$.*

Proof. It follows by induction that any type 2 node p_1 with $|\Phi_{p_1}| = |(\varphi, \varphi_{+n})|$ satisfies $|\Phi_{p_1}| \geq 2^\beta \max\{\eta, \eta_{+n}\}$. Let q and p_1 be as in the condition of the lemma. Then $|\Phi_q| < 2 \max\{\eta, \eta_{+n}\}$, completing the proof. $\square$

Proof of Lemma A.1. The proof is by induction. The stated inequalities clearly hold for all terminal nodes. Assume that $N_{1,p}^n < 2$ and that p is not a terminal node. Assume that the required inequalities hold for the children q and r of p . Exactly one of the children of p must be type 2 with two parents. Let r be this child, then $N_{1,r}^n = 0$. Note that since $|\Phi_r| \leq 2^{1-\beta}$, we have $\tan(|\Phi_r|/2) \leq (0.6)|\Phi_r|$. From Lemma A.2, via (26), and the definition of w (5.10), we now obtain

$$L_p - L_q \leq (1 + (0.6)2^{1-\beta}(|\Phi_r|)^2)w_{p,r}L_r.$$

Applying the inductive hypothesis yields

$$\begin{aligned} L_p &\leq L_q + (1 + (2^{2-3\beta}(|\Phi_r|)^2))(1 + (0.6)2^{1-\beta}(|\Phi_r|)^2)w_{p,r}\hat{L}_r^n \\ &\leq L_q + (1 + 2^{1-\beta}(|\Phi_r|)^2)w_{p,r}\hat{L}_r^n. \end{aligned}$$

Assume first that p is type 1. In this case, q must be type 1. Since $|\Phi_p| \geq 2|\Phi_r|$, we now have

$$\begin{aligned} L_p &\leq (1 + 2^{-1-\beta}(|\Phi_q|)^2)\hat{L}_q^n + (1 + 2^{-1-\beta}(|\Phi_p|)^2)w_{p,r}\hat{L}_r^n \\ &\leq (1 + 2^{-1-\beta}(|\Phi_p|)^2)\hat{L}_p^n. \end{aligned}$$

Assume now that p is type 2. In this case q is type 2. By Lemma A.3 we have $|\Phi_r| \leq 2^{1-\beta}|\Phi_p|$. We now proceed

$$\begin{aligned} L_p &\leq (1 + 2^{2-3\beta}(|\Phi_q|)^2)\hat{L}_q^n + (1 + 2^{2-3\beta}(|\Phi_p|)^2)w_{p,r}\hat{L}_r^n \\ &\leq (1 + 2^{2-3\beta}(|\Phi_p|)^2)(\hat{L}_q^n + w_{p,r}\hat{L}_r^n) \\ &= (1 + 2^{2-3\beta}(|\Phi_p|)^2)\hat{L}_p^n. \end{aligned} \quad \square$$

References

1. P. D. Alevizos, J. D. Boissonnat, and M. Yvinec. On the order induced by a set of rays: Application to then probing of non-convex polygons. *Proc. IEEE Internat. Conf. Robotics Automat.*, Phoenix, AZ (1989).
2. H. J. Bernstein. Determining the shape of a convex n -sided polygon by using $2n + k$ tactile probes. *Information Processing Letters* **22**, 255–260 (1986).
3. E. M. Bronštejn. ε -Entropy of convex sets and functions. *Siberian Mathematical Journal* **17**, 393–398 (1976).
4. R. Cole and C. K. Yap. Shape from probing. *Journal of Algorithms* **8** (1), 19–38 (1987).
5. D. Dobkin, H. Edelsbrunner, and C. K. Yap. Probing convex polytopes. *Proc. 18th ACM Symp. Theory of Comput.*, pp. 424–432 (1986).

6. R. M. Dudley. Metric entropy of some classes of sets with differentiable boundaries. *Journal of Approximation Theory* **10**, 227–236 (1974).
7. R. M. Dudley. *Uniform Central Limit Theorems*. To appear.
8. H. Edelsbrunner and S. S. Skiena. Probing convex polygons with X-rays. *SIAM Journal of Computing* **17**, 870–882 (1988).
9. J. P. Greschak. Reconstructing convex sets. Ph.D. dissertation, Dept. E.E. Comput. Sci. MIT, Cambridge, MA (1985).
10. W. E. L. Grimson and T. Lozano-Perez. Model based recognition and localization from sparse range or tactile data. *International Journal of Robotics Research* **3**(3), 3–35 (1984).
11. P. M. Gruber. Approximation of convex bodies. In: *Convexity and its Applications* (P. M. Gruber and J. M. Wills, eds.). Birkhäuser, Basel, pp. 131–162 (1983).
12. P. S. Kenderov. Polygonal approximation of plane convex compacts. *Journal of Approximation Theory* **38**, 221–239 (1983).
13. S. Y. R. Li. Reconstruction of polygons from projections. *Information Processing Letters* **28**, 235–240 (1988).
14. M. Lindenbaum and A. Bruckstein. Parallel strategies for geometric probing. *Journal of Algorithms* **13**, 320–349 (1992).
15. M. Lindenbaum and A. Bruckstein. Blind approximation of planar convex sets. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **10** (4), 517–529 (1994).
16. J. L. Prince and A. S. Wilsky. Reconstructing a convex set from noisy support line measurements. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **12** (4), 377–389 (1990).
17. T. J. Richardson. Total curvature and intersection tomography. *Advances in Mathematics*. To appear.
18. T. J. Richardson. Planar rectifiable curves are determined by their projections. *Discrete and Computational Geometry* **16**, 21–31 (1996).
19. L. A. Santaló. *Integral Geometry and Geometric Probability*. Addison-Wesley, Reading, MA (1976).
20. S. S. Skiena. Problems in geometric probing. *Algorithmica* **4**, 599–605 (1989).
21. S. S. Skiena. Geometric probing. Ph.D. dissertation, Technical Report No. UIUCDCS-R-88-1425, Dept. Comput. Sci., University of Illinois, Urbana, IL (1988).

Received April 18, 1995, and in revised form March 28, 1996.