Probing a Set of Hyperplanes by Lines and Related Problems

Yasukazu Aoki*, Hiroshi Imai*, Keiko Imai† and David Rappaport‡

Abstract

Suppose that for a set H of n unknown hyperplanes in the Euclidean d-dimensional space, a line probe is available which reports the set of intersection points of a query line with the hyperplanes. Under this model, this paper investigates the complexity to find a generic line for H and further to determine the hyperplanes in H. This problem arises in factoring the u-resultant to solve systems of polynomials (e.g., Renegar [12]). We prove that $d+1$ line probes are sufficient to determine H. Algorithmically, the time complexity to find a generic line and reconstruct H from $O(dn)$ probed points of intersection is important. It is shown that a generic line can be computed in $O(dn \log n)$ time after d line probes, and by an additional d line probes, all the hyperplanes in H are reconstructed in $O(dn \log n)$ time. This result can be extended to the d-dimensional complex space. Also, concerning the factorization of the u-resultant using the partial derivatives on a generic line, we touch upon reducing the time complexity to compute the partial derivatives of the u-resultant represented as the determinant of a matrix.

1 Introduction

Let H be a finite set of n hyperplanes $h_i = \{\mathbf{x} \in \mathbf{R}^d \mid \mathbf{a}_i^{\mathrm{T}}\mathbf{x} = b_i\}$ $(i = 1, \ldots, n)$ in $\mathbf{R}^d$. The arrangement of H is denoted by $\mathcal{A}(H)$. A function $f_H : \mathbf{R}^n \to \mathbf{R}$ defined by

$$f_H(\mathbf{x}) = \prod_{i=1}^{n} (\mathbf{a}_i^{\mathrm{T}}\mathbf{x} - b_i)$$

is called the *defining polynomial* of the arrangement $\mathcal{A}(H)$ (Orlik and Terao [11]), and also plays an important role in the interior-point method for linear programming (Karmarkar [7], Sonnevend [14]).

In modern elimination theory for algebraic equations such as Lazard's method [9], some way of computing the u-resultant, which is the defining polynomial of n hyperplanes in $\mathbf{C}^d$, of a given system of algebraic equations is presented, and solutions are obtained by decomposing the u-resultant into linear factors, that is, determining the hyperplanes. This factorization is done by probing the hyperplanes by lines (Canny [2]) or computing the gradient on a generic line to the hyperplanes by partially using probes (Kobayashi, Fujise and Furukawa [8], Renegar [12], Murao [10]). Thus arises the problem of probing hyperplanes by lines, a problem which does not seem to have been well studied in computational geometry. In contrast with this, probing a convex polyhedron is a well-studied problem (Cole and Yap [3], Dobkin, Edelsbrunner and Yap [4]). For a

*Department of Information Science, University of Tokyo, Hongo, Tokyo 113, Japan
†Department of Information and System Engineering, Chuo University, Kasuga, Tokyo 112, Japan
‡Department of Computing and Information Science, Queen's University, Kingston, Ontario, Canada K7L 3N6

survey of results in geometric probing see [13]. The theme of this paper is the previously unstudied problem of probing hyperplanes.

Let us define the problem in more detail. For simplicity, we consider the problem in real d-space $\mathbf{R}^d$ and all the hyperplanes in H are distinct. The cases of complex space and degenerate hyperplanes can be handled with slight modifications, which will be described in the full version of this paper. Suppose that a procedure to compute the defining polynomial $f_H(\mathbf{x})$ of the set H of n unknown hyperplanes in $\mathbf{R}^d$ is available in term of d variables $x_1, \ldots, x_d$ ($\mathbf{x} = (x_i)$), as in the case of the u-resultant. Using this procedure, for any point $\mathbf{p}$ in $\mathbf{R}^d$, we can determine whether $\mathbf{p}$ is on some hyperplane in H by checking if $f_H(\mathbf{p})$ is zero or not. This operation is called a *point probe.* For a line $\mathbf{p}+t\mathbf{q}$ passing through $\mathbf{p}$ with direction $\mathbf{q}$ ($\mathbf{q} \in \mathbf{R}^d$), $f_H(\mathbf{p}+t\mathbf{q})$ is a polynomial of degree at most n in terms of t. The points of intersection of the line with hyperplanes in H are computed by solving this univariate polynomial. This operation is called a *line probe.* It should be noted that a line probe obtains the points of intersection as a set, with no information regarding which point lies on which hyperplane. Concerning these fundamental operations, we assume that values of $f_H(\mathbf{p})$, $f_H(\mathbf{p}+t\mathbf{q})$ and the zeros, with their multiplicities, of a single-variable polynomial can be calculated exactly in a unit time (in section 5, we will discuss complexities related to compute $f_H(\mathbf{p}+t\mathbf{q})$). Also, fundamental operations $(+, -, \times, /)$ on real numbers are assumed to be executed exactly in a unit time. These assumptions allow us to focus on the number of necessary probes.

The following problems are considered when probing to reconstruct the set of hyperplanes H.

(1) Finding a generic line: Finding a *generic line*, a line intersecting all the hyperplanes in H at distinct points by line probes.

(2) Probing hyperplanes by lines: Constructing the hyperplanes in H by line probes.

(3) Computing the gradient of the defining polynomial: Computing the gradient $\nabla f_H(\mathbf{x})$ of $f_H(\mathbf{x})$ on a generic line efficiently.

Concerning (1), a randomly chosen line is generic with probability 1, but this is a probabilistic method [2, 8], and we want to devise a deterministic one. The algorithm by Renegar [12] is deterministic, but it uses $(d-1)n(n-1)/2+1$ line probes (a fixed direction) after finding a *generic point*, which is a point on no hyperplane in H, by $dn+1$ point probes. This paper shows that, assuming a generic point is at hand, the number of line probes that are needed to determine a generic line is d in the worst case, and given at most dn probed points, a generic line can be computed in $O(dn \log n)$ time. If probed points are provided sorted on a probe line then the time complexity is reduced to $O(dn)$.

Concerning (2), Canny [2] shows that, given a generic line, performing d general line probes and $dn(n+1)/2$ special line probes which verify whether a probing line is included in some hyperplane in H, the hyperplanes can be reconstructed in $O(dn^2)$ time. This paper shows that, by executing, $2d$ probes from scratch the hyperplanes can be reconstructed in $O(dn \log n)$ time.

Thus in both cases of (1) and (2), by treating the problems from the viewpoint of computational geometry, the probing and time complexities are greatly improved. Our results can be readily extended to the d-dimensional complex space $\mathbf{C}^d$.

Concerning (3), in [8, 12], it is shown that, given a generic line, all the hyperplanes are recovered by computing the gradient of the defining polynomial at each intersection points of the generic line with the hyperplanes. Here, it is required to compute the gradient quickly. This paper presents an efficient way of computing the gradient when the u-resultant is represented as the determinant of a matrix ([8, 10]) by making use of the idea in [1, 6]. Our improvement reduces the time complexity by a factor of degree 2. This result is somehow a computational algebraic result, but would be interesting from the viewpoint of computational geometry since the defining polynomial of the arrangement may be utilized further in the field.

2 Preliminaries

We have already defined fundamental concepts such as point probe, line probe, generic point, and generic line for the set H of n distinct hyperplanes in $\mathbf{R}^d$. In connection with line probes, we need some more definitions. A line is *parallel* to a hyperplane if the normal vector of the hyperplane is orthogonal to the direction of the line. The direction of a line which is not parallel to any hyperplane in H is called a *generic direction*. Of course, the direction of a generic line is a generic direction. Finally, the *multiplicity* of a point in $\mathbf{R}^d$ with respect to H is defined to be the number of hyperplanes in H containing the point. As was assumed, by a line probe, intersection points of the line with hyperplanes in H are obtained with their multiplicities. If the multiplicities of intersection points of a probing line sum up to n, then the probing line is in a generic direction. When a probing line is contained in a hyperplane in H, we only obtain this fact and no other information. To avoid this problem, a generic point is useful, since any line passing the generic point is not contained in any hyperplane in H. Renegar [12] gives a simple solution to find a generic point by point probes using the moment curve $(1, t, t^2, t^3, \ldots, t^{d-1})$ $(t \in \mathbf{R})$ in $\mathbf{R}^d$.

Lemma 1 (Renegar [12]) *For $dn+1$ distinct points on the moment curve, at least one point is a generic point.*

Hence, with $dn+1$ point probes, a generic point $\mathbf{q}$ is obtained. To compare point probe and line probe and also to explain the approach in [12], we describe how to find a generic line by a similar approach. If the hyperplanes in H are homogeneous, that is, all the hyperplanes pass the origin, the direction $\mathbf{q}$ is not parallel to any hyperplane in H, and so is a generic direction. Even if the hyperplanes are not homogeneous, by means of line probes, a generic direction can be easily found, as stated in the following.

Lemma 2 *By considering $(d-1)n+1$ distinct directions from the origin to points on the moment curve, at least one direction is a generic direction.*

Then, we can find a generic line by line probes conceptually considering the projection of the whole arrangement to a hyperplane perpendicular to the generic direction.

Lemma 3 (Renegar [12]) *For a given generic direction* $\mathbf{q}$*, considering* $(d-1)n(n-1)/2+1$ *distinct points* $\mathbf{p}_i(\neq \mathbf{q})$ *on the moment curve, one of lines represented by* $\mathbf{p}_i + t\mathbf{q}$ $(t \in \mathbf{R})$ *is a generic line.*

Thus, with $dn+1$ point probes (or $(d-1)n+1$ line probes) and $(d-1)n(n-1)/2+1$ line probes, a generic line can be found. Here, a crucial observation is that the numbers in Lemmas 1, 2 are tight in the worst case, which can be seen by considering a simple adversary, while the bound in Lemma 3 seems rather loose. We will show, in the next section, that much fewer line probes are sufficient to find a generic line.

Since a point probe is much less costly than a line probe, as may be seen in section 5, in the sequel, we assume that a generic point is pre-computed by point probes, and will not count for this complexity.

3 Tight Bounds for the Number of Necessary Probes

Making a given generic point as the origin, form the orthogonal $(x_1, x_2, \ldots, x_d)$-coordinate by choosing d independent directions in $\mathbf{R}^d$. Consider d line probes by these d axes. Denote by S_i the set of distinct intersection points of the x_i-axis with hyperplanes in H, where the multiplicities are ignored. Let $n_i = |S_i|$, where $n_i \leq n-$(the number of hyperplanes in H parallel to the x_i-axis). Denote the set of all probed points by S which is the disjoint union of S_i $(i = 1, \ldots, d)$. In total, we have $|S| = \sum_{i=1}^{d} n_i (\leq dn)$ probed points.

For a set S' of points and a hyperplane h in H, let $S' \cap h$ denote the set of points in S' which are contained in h. Since the probing lines are independent and are not contained in any hyperplane in H, we have the following lemma.

Lemma 4 *For any hyperplane* h *in* H *and* $i = 1, \ldots, d$*,* $|S \cap h| \geq 1$ *and* $|S_i \cap h| \leq 1$*.*

For $h \in H$, we represent the probed points by $I_h = \{ i \mid |S_i \cap h| = 1,\ i = 1, \ldots, d \}$. Then, for any $i \notin I_h$, the x_i-axis does not intersect with h, that is, the x_i-axis is parallel to h. Thus, h contains $|I_h|$ independent points in $S \cap h$ and is parallel to $d - |I_h|$ independent directions each of which is orthogonal to the affine hull of points in $S \cap h$. This implies that h is uniquely determined, in a reverse manner, if I_h and $|I_h|$ points, one from each S_i $(i \in I_h)$, are given.

Now, consider the set $\widetilde{H}$ of hyperplanes which are determined by I_h and $|I_h|$ points, one from each S_i $(i \in I_h)$, in the above manner for all possible I_h and points in S_i $(i \in I_h)$. Apparently, there are $\prod_{i=1}^{d}(n_i+1)-1$ possible ways of taking I_h and $|I_h|$ points, one from each S_i $(i \in I_h)$, and, for each of these, a distinct hyperplane is uniquely determined. Note that $I_h \neq \emptyset$. The next lemma follows from the discussion thus far.

Lemma 5 *$H \subseteq \widetilde{H}$, and $|\widetilde{H}| = \prod_{i=1}^{d}(n_i+1) - 1 < (n+1)^d$.*

We are now ready to prove the following theorem.

Theorem 1 *The number of line probes necessary to determine a generic line is at most d, and this bound is tight in the worst case.*

Proof. To show that d line probes are sufficient to find a generic line, we can use Lemma 5. First use line probes to obtain the superset $\widetilde{H}$. Then, we construct the arrangement $\mathcal{A}(\widetilde{H})$. (Although the size of $\mathcal{A}(\widetilde{H})$ is $O(n^{d^2})$ we will subsequently see how to make implicit use of $\mathcal{A}(\widetilde{H})$ without incurring a huge computational expense.) Given $\mathcal{A}(\widetilde{H})$, it is easy to compute a generic line for $\widetilde{H}$. Since $H \subseteq \widetilde{H}$, this line is also a generic line with respect to H.

We now prove this bound is tight in the worst case. For any set of at most $d-1$ line probes, there is a direction orthogonal to each probing line. Suppose there is a hyperplane in H which is orthogonal to this direction. Then, even if the hyperplane is moved orthogonally along the direction, the collection of probed points does not change. Hence, we cannot necessarily distinguish two different sets of hyperplanes by at most $d-1$ line probes. Furthermore, there may be a hyperplane not distinguishable by the first $d-1$ probes which can adversely effect any proposed generic line. $\square$

Although the size of $\mathcal{A}(\widetilde{H})$ is $O(n^{d^2})$ we will subsequently see how to make use of the regular structure of $\widetilde{H}$ to obtain a generic line without incurring the huge computational expense of constructing the entire arrangement.

A further key result is proved leading to an algorithm to reconstruct H from probes. An efficient implementation of the algorithm is discussed in the next section.

Theorem 2 *The number of line probes necessary to determine H is at most $d+1$, and this bound is tight in the worst case.*

Proof. As in Theorem 1, execute d line probes, and further execute a line probe by using the computed generic line. Since this generic line is generic to both H and $\widetilde{H}$, each probed point on the line for H coincides with exactly one probed point for $\widetilde{H}$. Then, hyperplanes in $\widetilde{H}$ each of which uniquely contains a probed point on the generic line for H are exactly those in H.

The lower bound can be shown in a way similar to the proof of Theorem 1. $\square$

Although we have so far assumed that hyperplanes in H are distinct to one another, we can generalize the above theorems easily even to degenerate case. These bounds are a big improvement to those found in [2, 12].

4 Efficient Probing Algorithm for Hyperplanes

In this section, we give an efficient algorithm for finding a generic line and identifying the hyperplanes. A main idea is to make use of some regular structure of the superset $\widetilde{H}$ of H discussed in the preceding section, specifically its grid structure in the dual space. To find a generic line, this algorithm only needs d line probes. Using at most dn probed points of intersection, a generic line is computed in $O(dn \log n)$ time.

First, as in the preceding section, make a generic point as the origin, and choose an arbitrary orthogonal basis to form the $(x_1, \ldots, x_d)$-coordinate. Using each x_i-axis as a probing line, find all the points of intersection. Then, the superset $\widetilde{H}$ of H is formed from these probed points.

We now transform the original space (primal space) to the dual space so that the regular structure of the superset $\widetilde{H}$ may be more easily seen in the dual space. Since the hyperplanes in H do not pass the origin, each hyperplane may be represented as $\mathbf{a}^{\mathrm{T}}\mathbf{x} = 1$. We may then consider a dual transform which maps a point $\mathbf{a}$ in the primal space to a hyperplane $\mathbf{a}^{\mathrm{T}}\mathbf{x} = 1$ in the dual space. We will use the same coordinate system in the dual space. By this duality, a hyperplane in the primal space is mapped to a point in the dual space. Note that a hyperplane parallel to the x_i-axis in the primal space is mapped to a point in the hyperplane $x_i = 0$ in the dual space.

By this duality transform, each probed point on the x_i-axis is mapped to a hyperplane orthogonal to the same axis. By these hyperplanes corresponding to probed points and hyperplanes $x_i = 0$ $(i = 1, \ldots, n)$ in the dual space, $\prod_{i=1}^{d}(n_i + 1)$ grid points are formed in the dual space. Denote this set of at most $(n+1)^d$ grid points by $\widetilde{H}_D$. Hyperplanes in H are mapped to points in the dual space, whose set is denoted by H_D. Then, by the duality, the following lemma holds.

Lemma 6 *The dual of $\widetilde{H}$ is $\widetilde{H}_D$ minus the origin point, and $H_D \subseteq \widetilde{H}_D$.*

$\widetilde{H}_D$ has a nice regular structure as grid points. Although its total size is at most $(n+1)^d$, it is originally produced by d sets of at most $n+1$ parallel hyperplanes, and hence has a compact representation.

We will search for, L, a generic line to $\widetilde{H}$, passing through the origin in the primal space. Then this problem in the primal space is transformed to the problem, in the dual space, of finding a hyperplane orthogonal to the line L such that when moving the hyperplane orthogonally along the line no two grid points of $\widetilde{H}_D$ are hit simultaneously. Note that no points in $\widetilde{H}_D$ lie on L implying that there are no hyperplanes in H parallel to L.

Lemma 7 *A line L passing through the origin, with the property that every two points from $\widetilde{H}_D$ have distinct orthogonal projections onto L, is generic for $\widetilde{H}$.*

Let the set of $n_i + 1$ distinct grid points in $\widetilde{H}_D$ incident to the dual x_i-axis be represented by their x_i coordinate as $\{\, x_i = \overline{x}_{i,j} \mid j = 1, \ldots, n_i + 1 \,\}$ with $\overline{x}_{i,0} < \overline{x}_{i1} < \cdots < \overline{x}_{i,n_i}$ (one of $\overline{x}_{i,j}$

($j = 0, \dots, n_i$) is zero). For each $i = 1, \dots, d$, define the width w_i of $\overline{x}_{i,j}$ ($j = 0, \dots, n_i$) to be $\overline{x}_{i,n_i} - \overline{x}_{i,0}$. Also, for each $i = 1, \dots, d$, define the minimum gap g_i by $g_i = \min_{j=1,\dots,n_i} (\overline{x}_{i,j} - \overline{x}_{i,j-1})$. We here assume that $n_i > 1$ for each i. Using these, we will try to find a line L, passing through the origin such that any hyperplane orthogonal to the line L contains at most one grid point in $\widetilde{H}_D$.

Theorem 3 *Let r_i be a real number with $r_i \geq 2$ ($i = 1, \dots, d$). Define the ith element q_i of $\mathbf{q} \in \mathbf{R}^d$ for $i = 1, \dots, d$ in this order by*

$$\begin{aligned} q_1 &= \frac{4}{r_1} \\ q_i &= \frac{g_{i-1}}{r_i w_i} q_{i-1} \quad (i = 2, \dots, d) \end{aligned}$$

Then, a line represented by $t\mathbf{q}$ ($t \in \mathbf{R}$) passing through the origin is a generic line in the primal space.

Proof. Consider two distinct grid points $\mathbf{x}'$ and $\mathbf{x}''$ in $\widetilde{H}_D$. Let l be the minimum l such that the x_l-coordinate values of $\mathbf{x}'$ and $\mathbf{x}''$ are different, that is, $x'_i = x''_i$ ($i = 1, \dots, l-1$) and $x'_l \neq x''_l$. Without loss of generality, we can assume $x'_l > x''_l$. Then, we have

$$\begin{aligned} \mathbf{q}^{\mathrm{T}}\mathbf{x}' - \mathbf{q}^{\mathrm{T}}\mathbf{x}'' &= \sum_{i=l}^{d} q_i(x'_i - x''_i) \geq q_i(x'_l - x''_l) - \sum_{i=l+1}^{d} q_i|x'_i - x''_i| \\ &\geq q_l g_l - \sum_{i=l+1}^{d} q_i w_i = q_l g_l - \sum_{i=l+1}^{d} \left(w_i q_{l+1} \prod_{j=l+1}^{i-1} \frac{g_j}{r_{j+1} w_{j+1}} \right) \\ &\geq q_l g_l - q_{l+1} g_{l+1} \sum_{i=l+1}^{d} \left(\frac{1}{2^{i-l-1}} \prod_{j=l+2}^{i-1} \frac{g_j}{w_j} \right) \geq q_l g_l - q_{l+1} g_{l+1} \sum_{i=l+1}^{d} \frac{1}{2^{i-l-1}} \\ &> q_l g_l - 2 q_{l+1} g_{l+1} \geq q_l g_l \left(1 - \frac{g_{l+1}}{w_{l+1}} \right) \geq 0 \end{aligned}$$

Hence, two distinct grid points are not on the same hyperplane whose normal vector is $\mathbf{q}$, and the theorem follows. □

Given $\{\overline{x}_{i,j}\}$, w_i and g_i ($i = 1, \dots, d$) are computed in $O(dn \log n)$ time by sorting $\overline{x}_{i,j}$ ($j = 0, \dots, n_i$) for each $i = 1, \dots, d$. If probed points on a line in the primal space are given in a sorted order, these values can be computed in $O(dn)$ time. After computing these values, $\mathbf{q}$ is found in $O(d)$ time. Hence, we have the following theorem.

Theorem 4 *Given a generic point, from d line probes, a generic line can be found in $O(dn \log n)$ time, and, if intersection points on a probing line are given in a sorted order, the complexity is reduced to $O(dn)$.*

We now thus establish an efficient way of finding a generic line. We can generalize this result for the complex space $\mathbf{C}^d$. Concerning the problem of finding the hyperplanes in H by line probes, we have the following.

Theorem 5 *Given a generic point, from $2d$ line probes in total, all the hyperplanes in H can be determined in $O(dn \log n)$ time.*

Proof. In Theorem 3, $\mathbf{q}$ is computed for some r_i $(i = 1, \ldots, d)$ with $r_i \geq 2$. Let $\mathbf{q}_i$ be $\mathbf{q}$ in the theorem obtained for r_j $(j = 1, \ldots, d)$ defined by $r_j = 4$ for $j = i$ and $r_j = 2$ for $j \neq i$. Then, by the theorem, each line $t\mathbf{q}_i$ $(t \in \mathbf{R})$ is a generic line. Furthermore these generic lines hit the hyperplanes in H in the same order, because, for each $\mathbf{q}_i$, by the proof of the theorem, we see that all grid points on $x_1 = \overline{x}_{1,j_1}$ are traversed before grid points on $x_1 = \overline{x}_{1,j_1+1}$ are done, and, among grid points on $x_1 = \overline{x}_{1,j_1}$, grid points on $x_2 = \overline{x}_{2,j_2}$ are traversed before grid points on $x_2 = \overline{x}_{2,j_2+1}$, and so on. Finally, since grid points on $d-1$ hyperplanes $x_i = \overline{x}_{i,j_i}$ $(i = 1, \ldots, d-1)$ have distinct x_d-coordinate values, grid points in $\widetilde{H}_D$ are passed over by the hyperplane with normal vector $\mathbf{q}_i$ in a unique order.

Hence, denoting the points of intersection of line $t\mathbf{q}_i$ and hyperplanes in H by $b_{i,j}\mathbf{q}_i$ $(j = 1, \ldots, n)$ with $b_{i,1} < b_{i,2} < \cdots < b_{i,n}$ $(i = 1, \ldots, d)$, the jth hyperplane in H contains $b_{1,j}\mathbf{q}_1,\ b_{2,j}\mathbf{q}_2, \ldots, b_{d,j}\mathbf{q}_d$.

Let A be a $d \times d$ matrix whose ith column vector is $\mathbf{q}_i$. Defining α_i by $\alpha_1 = 1$ and $\alpha_i = \dfrac{g_{i-1}}{2w_i}\alpha_{i-1}$ $(i = 2, \ldots, d)$, A is expressed as follows.

$$A = \begin{pmatrix} \alpha_1 & 2\alpha_1 & 2\alpha_1 & \cdots & 2\alpha_1 \\ \alpha_2 & \alpha_2 & 2\alpha_2 & \cdots & 2\alpha_2 \\ \alpha_3 & \alpha_3 & \alpha_3 & \cdots & 2\alpha_3 \\ \cdot & \cdot & \cdot & \cdots & \cdot \\ \alpha_{d-1} & \alpha_{d-1} & \alpha_{d-1} & \cdots & 2\alpha_{d-1} \\ \alpha_d & \alpha_d & \alpha_d & \cdots & \alpha_d \end{pmatrix}$$

As is readily seen, A is nonsingular. Its inverse is given by

$$A^{-1} = \begin{pmatrix} -1/\alpha_1 & 0 & 0 & \cdots & 0 & 2/\alpha_d \\ 1/\alpha_1 & -1/\alpha_2 & 0 & \cdots & 0 & 0 \\ 0 & 1/\alpha_2 & -1/\alpha_3 & \cdots & 0 & 0 \\ \cdot & \cdot & \cdot & \cdots & \cdot & \cdot \\ 0 & 0 & 0 & \cdots & 1/\alpha_{d-1} & -1/\alpha_d \end{pmatrix}.$$

Let B_j be a $d \times d$ diagonal matrix whose diagonal elements are $b_{1,j}, b_{2,j}, \ldots, b_{d,j}$. The ith column of AB_j is $b_{i,j}\mathbf{q}_i$, and, since A is nonsingular and $b_{i,j}$ $(i = 1, \ldots, d)$ are not zeros, AB_j is nonsingular.

Let the jth hyperplane in H be $\mathbf{a}_j^{\mathrm{T}}\mathbf{x} = 1$ (recall that this hyperplane does not pass the origin). Then, $\mathbf{a}_j^{\mathrm{T}}AB_j = (1, 1, \ldots, 1)$. Since AB_j is nonsingular, $\mathbf{a}_j$ is determined uniquely from A and B_j, that is, the jth hyperplane in H can be uniquely determined by points $b_{1,j}\mathbf{q}_1,\ b_{2,j}\mathbf{q}_2, \ldots, b_{d,j}\mathbf{q}_d$ on the hyperplane.

All the hyperplanes in H can thus be obtained. About the time complexity, A^{-1} is simple enough as above, and, computing $\alpha_1, \ldots, \alpha_d$ once and computing $\mathbf{q}_1, \ldots, \mathbf{q}_d$ from them, we can compute $\mathbf{a}_j$ $(j = 1, \ldots, n)$ in $O(dn \log n)$ time in total. □

5 Computing the Gradient of the u-Resultant

Due to the space limitations, we will only sketch the results on the u-resultant (e.g., see [15]). We illustrate our ideas through a small example taken from [9].

Consider a system of algebraic equations

$$
\begin{aligned}
x^2 + xy + 2x + y - 1 = 0 \\
x^2 - y^2 + 3x + 2y - 1 = 0
\end{aligned}
$$

which has a finite number (4 in this case) of solutions. The solution of this system is $(0,1)$, $(1,-1)$, $(-3,0)$ and a solution at infinity in the direction $(-1,1)$. There are several methods to compute the u-resultant, and, by the algorithm in [9], the u-resultant is expressed as the determinant of a 4×4 matrix as follows:

$$
f_H(u_1,u_2,u_3) = \begin{vmatrix} u_1+u_2-u_3 & 0 & 0 & 0 \\ -u_2+u_3 & -u_1+2u_2-u_3 & u_2 & 0 \\ u_3 & -2u_2 & u_1-u_2+u_3 & 0 \\ 0 & 0 & 0 & -u_2+u_3 \end{vmatrix}
$$

Since in this case the problem is small and well-behaved, we can directly expand the determinant and decompose it as follows:

$$
f_H(u_1,u_2,u_3) = (u_1+u_2-u_3)(u_1-3u_2+u_3)(u_1+u_3)(u_2-u_3)
$$

A linear term $a_1u_1 + a_2u_2 + a_3u_3$ in the decomposition corresponds to a solution $(u_2/u_1, u_3/u_1)$ by the theory of u-resultant. Hence, four terms in the right-hand side in this case corresponds to $(1,-1)$, $(-3,1)$, $(0,1)$ and a solution at infinity in the direction $(-1,1)$.

In this simple example, the determinant can be expanded and its decomposition can be found very easily. However, if the system becomes larger, expanding the determinant of a large matrix having d (in the above case, $d=3$) indeterminates is very time-consuming, and, even in case the expansion is obtained, its direct decomposition is again very hard.

In general cases, $f_H(\mathbf{u})$ is given as the determinant of an $n \times n$ matrix $R(\mathbf{u})$ of d variables $u_1, \ldots, u_d$, and the ij-element of $R(\mathbf{u})$ is given by $r_{ij}(\mathbf{u}) = \mathbf{a}_{ij}^{\mathrm{T}}\mathbf{u}$ [8, 9]. As mentioned above, expanding the determinant of $R(\mathbf{u})$ takes much time. Roughly, it takes $O(n^{3+d})$ time by the interpolation method. However, to perform a line probe for line $\mathbf{p}+t\mathbf{q}$, we first compute the determinant $f_H(\mathbf{p}+t\mathbf{q})$ of $R(\mathbf{p}+t\mathbf{q})$ with a single variable t to obtain a single-variable polynomial, and then compute its zeros. Here, the determinant of $R(\mathbf{p}+t\mathbf{q})$ can be computed in $O(n^4)$ time, independent of d. Therefore, instead of expanding the original determinant with many variables once, which is almost impossible even for moderate-size problems, expanding the determinant of a single variable $d+1$ or $2d$ times by executing $d+1$ or $2d$ line probes, respectively, is definitely economical in total, and this is one of the most practical approaches to solve the problem. Probing by lines are thus useful.

Now, suppose all the solutions are distinct, or in other words in the corresponding arrangement there are no two identical hyperplanes (the general case will be discussed in the full version). Consider a generic line, represented by $\mathbf{p}+t\mathbf{q}$, and let $\mathbf{p}+t_i\mathbf{q}_1$ $(i = 1, \ldots, n)$ be points of intersection of the generic line with hyperplanes. Then, from the form of the defining polynomial,

$$\left(\left.\frac{\partial f_H(\mathbf{u})}{\partial u_1}\right|_{\mathbf{u}=\mathbf{p}+t_i\mathbf{q}}, \left.\frac{\partial f_H(\mathbf{u})}{\partial u_2}\right|_{\mathbf{u}=\mathbf{p}+t_i\mathbf{q}}, \ldots, \left.\frac{\partial f_H(\mathbf{u})}{\partial u_d}\right|_{\mathbf{u}=\mathbf{p}+t_i\mathbf{q}} \right)$$

is the direction of normal vector of the ith hyperplane [8, 12] (note that $\mathbf{p} + t_i\mathbf{q}$ is on only the ith hyperplane). From this, the ith solution of the original equation can be readily computed. Thus, computing the gradient of $f_H(\mathbf{u})$ with respect to $\mathbf{u}$ is a main step of this approach.

We here give an efficient way of doing this. A basic idea is from [1, 6].

Theorem 6 *Suppose that the determinant $|R(u(t))|$ of $R(u(t))$ with t as a variable is not vanishing. For $\mathbf{u} = \mathbf{u}(t) = \mathbf{p} + t\mathbf{q}$ with t as a variable,*

$$\left.\frac{\partial f_H(\mathbf{u})}{\partial u_l}\right|_{\mathbf{u}=\mathbf{u}(t)} = \mathrm{trace}\left[\frac{\partial R(\mathbf{u})}{\partial u_l}\left(|R(\mathbf{u}(t))|R(\mathbf{u}(t))^{-1}\right)\right]$$

where $\partial R(\mathbf{u})/\partial u_l$ is an $n \times n$ matrix whose ij-element is $\partial r_{ij}(\mathbf{u})/\partial u_l$, which is the lth element of $\mathbf{a}_{ij}$.

The proof is omitted in this version. Note that the trace in Theorem 6 can be computed in $O(n^3)$ time by summing up n^2 products of a real constant and a single-variable polynomial of degree at most $n-1$. The inverse $M(\mathbf{u}(t))^{-1}$ is computed just once, which takes $O(n^4)$ time by the interpolation method, and applying the above theorem for each i we have the following.

Theorem 7 *Given the defining polynomial $f_H(\mathbf{u})$ as the determinant of a matrix $R(\mathbf{u})$, the gradient of $f_H(\mathbf{u})$ with respect to $\mathbf{u}$ on a line $\mathbf{p} + t\mathbf{q}$ can be computed in $O(n^4)$ time.*

Acknowledgment

The work by the fourth author was done while he was visiting Department of Information Science, University of Tokyo during September, 1992, by the bilateral program between NSERC (Canada) and JSPS (Japan). The work by the second and third authors was partially supported by the Grant-in-Aid of the Ministry of Education, Science and Culture of Japan.

References

[1] Baur, W., and V. Strassen, "The Complexity of Partial Derivatives," *Theoretical Computer Science*, Vol.22 (1983), pp.317–330.

[2] Canny, J., "A New Algebraic Method for Robot Motion Planning and Real Geometry," *Proceedings of the 28th IEEE Annual Symposium on Foundations of Computer Science*, 1987, pp.39-48.

[3] Cole, R., and C. Yap, "Shape from Probing," *Journal of Algorithms*, Vol.8 (1987), pp.19–38.

[4] Dobkin, D., H. Edelsbrunner, and C. K. Yap, "Probing Convex Polytopes," *Proceedings of the 18th Annual ACM Symposium on Theory of Computing*, 1986, pp.424–432.

[5] Edelsbrunner, H., "*Algorithms in Combinatorial Geometry*," Springer-Verlag, 1987.

[6] Iri, M., "Simultaneous Computation of Functions, Partial Derivatives and Estimates of Rounding Errors — Complexity and Practicality," *Japan Journal of Applied Mathematics*, Vol.1, No.2 (1984), pp.223–252.

[7] Karmarkar, N., "A New Polynomial-Time Algorithm for Linear Programming," *Combinatorica*, Vol.4 (1984), pp.373–395.

[8] Kobayashi, H., T. Fujise and A. Furukawa, "Solving Systems of Algebraic Equations by a General Elimination Method," *Journal of Symbolic Computation*, Vol.5, No.3 (1988), pp.303–320.

[9] Lazard, D., "Résolution des Systèmes d'Équations Algébriques," *Theoretical Computer Science*, Vol.15 (1981), pp.77–110.

[10] Murao, H., *Development of Efficient Algorithms in Computer Algebra*," Doctoral Thesis, Department of Information Science, University of Tokyo, 1992.

[11] Orlik, P., and H. Terao, "*Arrangements of Hyperplanes*," Springer-Verlag, 1991.

[12] Renegar, J., "On the Worst-Case Arithmetic Complexity of Approximating Zeros of Systems of Polynomials," *SIAM Journal on Computing*, Vol.18, No.2 (1989), pp.350–370.

[13] Skiena S., "Interactive reconstruction via probing," *IEEE Proceedings*, to appear.

[14] Sonnevend, Gy., "An "Analytical Centre" for Polyhedrons and New Classes of Global Algorithms for Linear (Smooth, Convex) Programming," *Lecture Notes in Control and Information Sciences*, Vol.84, Springer-Verlag, 1986, pp.866–876.

[15] van der Waerden, B. L., "*Moderne Algebra*," Vol.II, Springer-Verlag, 2nd Edition, 1940.