Efficient Algorithms for Bichromatic Separability

PANKAJ K. AGARWAL

Duke University

BORIS ARONOV

Polytechnic University

AND

VLADLEN KOLTUN

Stanford University

Abstract. A closed solid body *separates* one point set from another if it contains the former and the closure of its complement contains the latter. We present a near-linear algorithm for deciding whether two sets of n points in $\mathbb{R}^3$ can be separated by a prism, near-quadratic algorithms for separating by a slab or a wedge, and a near-cubic algorithm for separating by a double wedge. The latter three algorithms improve the previous best known results by an order of magnitude, while the prism separability algorithm constitutes an improvement of two orders of magnitude.

Categories and Subject Descriptors: F.2.2 [**Analysis of Algorithms and Problem Complexity**]: Nonnumerical Algorithms and Problems

General Terms: Algorithms, Theory

Additional Key Words and Phrases: Geometric algorithms, separability, arrangements

---

P. K. Agarwal was supported in part by the National Science Foundation (NSF) under grants CCR-00-86013, EIA-98-70724, EIA-99-72879, EIA-01-31905, and CCR-02-04118, and by a grant from the U.S.-Israeli Binational Science Foundation.

B. Aronov was supported in part by NSF grants CCR-99-72568 and ITR CCR-00-81964 and by a grant from the U.S.-Israeli Binational Science Foundation.

V. Koltun was supported in part by NSF grant CCR-01-21555.

Authors' addresses: P. K. Agarwal, Department of Computer Science, Levine Science Research Center D315, Box 90129, Duke University, Durham, NC 27708-0129, e-mail: pankaj@cs.duke.edu; B. Aronov, Department of Computer and Information Science, Dibner Building 236, Polytechnic University, Six MetroTech Center, Brooklyn, NY 11201, e-mail: aronov@poly.edu; V. Koltun, Computer Science Department, 353 Serra Mall, Gates 464, Stanford University, Stanford, CA 94305, e-mail: vladlen@cs.stanford.edu.

1. *Introduction*

Many problems in supervised machine learning can be formulated as classifying objects into a finite set of categories, based on a given training set. In its simplest form, each object in the training set is labeled $+1$ or -1, and the goal is to build a *predictor* based on this training set that enables us to classify a new object into one of these two classes. Because of its wide range of applications, this fundamental classification problem and its generalizations have been extensively studied in machine learning. Numerous powerful learning techniques such as decision trees, boosting, support vector machines, logistic regression, etc. have been developed in the last two decades. See the recent book by Hastie et al. [2001] for a discussion on these and other techniques. Despite these advances in statistical techniques, relatively little progress has been made on combinatorial techniques for classification, especially in three and higher dimensions.

In this article, we study the following simple geometric version of the above classification problem in $\mathbb{R}^3$, also known as the *separability* problem: Let R be a set of n points in $\mathbb{R}^3$, and let B be another set of n points in $\mathbb{R}^3$. Given a family of closed three-dimensional bodies F, determine whether there exists $F \in \mathrm{F}$ such that $\mathrm{R} \subseteq F$ and $\mathrm{B} \cap \operatorname{int} F = \emptyset$ and if so return such a body F. This separability problem is obviously an instance of supervised learning, as the separator F can be used to predict the class of an arbitrary point in $\mathbb{R}^3$. Namely, if a point lies in F, we assign it to R, otherwise to B.

1.1. RELATED WORK. If F is the class of all half-spaces, the separability problem can be reduced to linear programming and solved in linear time for any fixed dimension [Megiddo 1983], and algorithms are known for finding an "optimal" separating hyperplane [Vapnik 1996]. Most other work in computational geometry on separability and/or classification has focused on two-dimensional problems. A series of papers studied the problem of separating two sets in the plane by a circle (or a disk, in above terminology), eventually leading to a linear-time algorithm by O'Rourke et al. [1986]. Recently, researchers have studied the problem of separating two planar point sets by other objects such as a convex polygon with the minimum number of edges [Edelsbrunner and Preparata 1988], a double wedge [Hurtado et al. 2004], a wedge, or a strip [Hurtado et al. 2004]. Arkin et al. [2001] proved an $\Omega(n \log n)$ lower bound for many of these separability problems. The problem of separating two planar sets by a simple polygon with the minimum number of edges is known to be NP-complete [Fekete 1992] and an approximation algorithm is given by Mitchell [1993]. See Agarwal and Suri [1998] for several other related separation results in the plane.

Relatively little is known about the separability problems in three and higher dimensions. The algorithm by O'Rourke extends to higher dimensions. Recently, Hurtado et al. [2003] proposed cubic or slightly super-cubic separability algorithms in $\mathbb{R}^3$ when F is the family of slabs (regions bounded by pairs of parallel planes), convex dihedral wedges (intersections of two half-spaces), infinite convex prisms, and infinite convex cones. They also developed an $O(n^4)$ algorithm for deciding separability with a double wedge and algorithms with running times ranging from $O(n^5)$ to $O(n^8)$ for other three-dimensional separability problems. Researchers have also studied the problem of separating geometric objects other than points. See Boissonnat et al. [2000] and Brönnimann and Goodrich [1995] and references therein for several such results.

1.2. OUR RESULTS. In this article, we develop separability algorithms for several families of separators in $\mathbb{R}^3$. In particular, we obtain the following results:

(i) An $O(n \log^4 n)$ deterministic algorithm for separating by a (convex infinite) prism.
(ii) An $O(n^2 \log n)$ expected-time randomized algorithm for separating by a slab; a deterministic algorithm with the same worst-case running time also exists.
(iii) An $O(n^2 \log^2 n \log^2(\log n))$ expected-time randomized or $O(n^2 \text{ polylog}(n))$-time deterministic algorithm for separating by a (convex dihedral) wedge.
(iv) An $O(n^3 \log^2 n \log^2(\log n))$ expected-time randomized or $O(n^3 \text{ polylog}(n))$-time deterministic algorithm for separating by a double wedge.

Finally, we also study the problem of separating by a (convex infinite) cone. Currently, we do not have a subcubic algorithm for cone separability, but we prove a number of structural properties of cone separability that, we believe, should yield an $O(n^2 \text{ polylog}(n))$ algorithm for this problem.

Our results rely on techniques from the algorithmic theory of arrangements and on geometric data structures. These results improve the previously known bounds by at least an order of magnitude and, in the case of prisms, by two orders of magnitude. We believe that our approach can be extended to derive improved algorithms for other separability problems in $\mathbb{R}^3$, including ones for which no non-trivial solutions are currently known, such as the cases of circular cylinder or circular cone separators, which were stated as open problems in Hurtado et al. [2003]. We consider our results to be of particular appeal due to the conceptual simplicity of the problems studied.

2. *General Approach*

Before presenting our specific results, we devote this section to describing the general approach taken in this article, as well as a few geometric concepts that will be crucial for our algorithms.

2.1. MINIMAL SEPARATORS AND THEIR REPRESENTATION. We call a body $F \in \mathcal{F}$ a *container* (with respect to $\mathcal{R}$) if $\mathcal{R} \subseteq F$. A container is called *minimal* if there is no other container G in $\mathcal{F}$ with $G \subset F$. Finally, a (minimal) container F is called a (*minimal*) *separator* if $\mathcal{B} \cap \text{int}(F) = \emptyset$.

The families $\mathcal{F}$ that we study in this article have the property that a minimal container can be represented as a point in a low-dimensional space despite the possibility that its combinatorial complexity may be $\Omega(n)$. For example, a minimal prism container can be represented by the direction of its axis. Indeed, if we fix a direction u on the unit sphere $\mathbb{S}^2$, then there is a unique minimal prism container with this axis direction, namely, the one formed by the Minkowski sum of $\text{conv}(\mathcal{R})$ with the line through the origin in direction u. Hence, the set of minimal prism containers can be represented by $\mathbb{S}^2$. Similarly, if $\mathcal{F}$ is the family of cones in $\mathbb{R}^3$, then a minimal cone container can be represented by its apex, a point in $\mathbb{R}^3$.

Let $\mathcal{C} = \mathcal{C}(\mathcal{F}, \mathcal{R})$ be the parametric space that represents the set of minimal containers in $\mathcal{F}$, that is, each point in $\mathcal{C}$ corresponds to a container in $\mathcal{F}$ that is minimal with respect to $\mathcal{R}$, such that every minimal container is represented by a point in $\mathcal{C}$ in this way. With a slight abuse of notation, we will not distinguish between a point in $\mathbb{C}$ and the corresponding minimal container $F \in \mathbb{F}$. Given a set $\mathbb{B}$ of points in $\mathbb{R}^3$, let $\Sigma = \Sigma(\mathbb{B}, \mathbb{R}, \mathbb{F}) \subseteq \mathbb{C}$ denote the set of minimal separators. For a point $p \in \mathbb{R}^3$, let $\mathbb{K}_p \subseteq \mathbb{C}$ denote the set of minimal containers that also contain p, that is, for each $C \in \mathbb{K}_p$, $\mathbb{R} \cup \{p\} \subseteq C$. Then, $\Sigma = \mathbb{C} \setminus \bigcup_{b \in \mathbb{B}} \text{int}(\mathbb{K}_b)$.[1] The problem of determining whether there exists a body $F \in \mathbb{F}$ that separates $\mathbb{B}$ from $\mathbb{R}$ is equivalent to determining whether $\Sigma \neq \emptyset$.

One possible approach is to bound the combinatorial complexity (i.e., the total number of vertices, edges, and facets) of Σ and to compute a boundary representation of Σ in time roughly proportional to its complexity. However, in most cases we do not compute Σ in its entirety because this could be quite expensive. Instead, we prove various geometric and topological properties of Σ and design faster algorithms for detecting whether $\Sigma = \emptyset$. For example, we show that if $\mathbb{F}$ is the set of prisms in $\mathbb{R}^3$, then $\Sigma(\mathbb{B}, \mathbb{R}, \mathbb{F})$ can be represented as the intersection of two regions X and Y on the unit sphere. We compute a compact representation of each of X and Y and using this representation determine in $O(n \log^4 n)$ time whether $X \cap Y = \emptyset$, even though the worst-case complexity of $\Sigma(\mathbb{B}, \mathbb{R}, \mathbb{F})$ is $\Omega(n^2)$. Similarly, for the case of wedges and double wedges, we search over various cross sections of $\partial\Sigma$ to detect its non-emptiness.

We make the following assumptions for the sake of simplicity. The algorithms can easily be modified to work for the general case.

(A1) The points $\mathbb{R} \cup \mathbb{B}$ are in general position, in the sense that no two points have the same x-, y-, or z-coordinate, no two points lie on a line passing through the origin, no three points are collinear or lie in a common vertical plane, and no four points are coplanar. We also assume that the origin o lies in the interior of conv($\mathbb{R}$).

(A2) $\Sigma = \text{cl}(\mathbb{C} \setminus \bigcup_{b \in \mathbb{B}} \mathbb{K}_b)$. (This assumption emphasizes that, for the sake of consistency in exposition, we choose to view Σ as a closed set. In particular, a minimal container $C \in \mathbb{C}$, such that $\mathbb{B} \cap \text{int}(C) = \emptyset$ but $\mathbb{B} \cap \text{cl}(C) \neq \emptyset$ is still considered to be a separator.)

(A3) Since $\mathbb{F}$ is a family of convex bodies in Sections 3–5 (note that double wedges are not convex), we assume that $\text{int}(\text{conv}(\mathbb{R})) \cap \mathbb{B} = \emptyset$ in these sections. Indeed if $\text{int}(\text{conv}(\mathbb{R})) \cap \mathbb{B} \neq \emptyset$, then no body in $\mathbb{F}$ can separate $\mathbb{R}$ from $\mathbb{B}$ and, since we are dealing with point sets in $\mathbb{R}^3$, we can check this condition in $O(n \log n)$ time.

2.2. ARRANGEMENTS. Let $\mathbb{H}$ be a set of n hyperplanes in $\mathbb{R}^d$. The *arrangement* of $\mathbb{H}$, denoted by $\mathbb{A}(\mathbb{H})$, is the decomposition of $\mathbb{R}^d$ into (relatively open) *cells* so that each cell is a maximal connected region that lies in the same subset of $\mathbb{H}$. Each cell of $\mathbb{A}(\mathbb{H})$ is a convex polyhedron. The *combinatorial complexity* of a k-dimensional cell C, denoted as $|C|$, is the number of cells of $\mathbb{A}(\mathbb{H})$ of dimension less than k that are contained in ∂C, the boundary of C; see Agarwal and Sharir [2000] for a survey on arrangements. It is well known that the complexity of a single cell in $\mathbb{A}(\mathbb{H})$ is $O(n^{\lfloor d/2 \rfloor})$ and that the complexity of the entire arrangement is $O(n^d)$. The *lower* (respectively, *upper*) *envelope* of $\mathbb{H}$ is the boundary of the d-dimensional cell of $\mathbb{A}(\mathbb{H})$ that lies below (respectively, above) all hyperplanes of

[1] Strictly speaking, by $\text{int}(\mathbb{K}_b)$ we mean interior of $\mathbb{K}_b$ relative to $\mathbb{C}$, but we will ignore this technicality.

H. The *zone* of H with respect to a surface Γ, denoted as $\mathrm{Z} = \mathrm{Z}(\mathrm{H}, \Gamma)$, is the set of d-dimensional cells of $\mathcal{A}(\mathrm{H})$ that intersect Γ. $\sum_{C \in \mathrm{Z}} |C| = O(n^{d-1} \log n)$ if Γ is a convex surface and $O(n^{d-1})$ if Γ is a hyperplane [Aronov et al. 1993].

3. *Prism Separability*

In this section, we assume F to be the set of (convex infinite) prisms in $\mathbb{R}^3$. Recall that a *prism* F is the Minkowski sum of a closed, convex body C with a line ℓ. The direction of ℓ is referred to as the *direction* of F. Since F is a family of convex bodies, we describe the algorithm under Assumption (A3). As mentioned in Section 2, for a direction $u \in \mathbb{S}^2$, the Minkowski sum of conv(R) and a line in direction u through the origin is the minimal prism container $F(u)$ in direction u. Note that $F(u)$ and $F(-u)$ are the same, but for simplicity we distinguish them. We thus represent C, the set of minimal prism containers, by $\mathbb{S}^2$. We first analyze the structure of Σ and then describe the algorithm for determining whether $\Sigma = \emptyset$.

3.1. STRUCTURE OF Σ. For a point $p \in \mathbb{R}^3 \setminus \mathrm{conv}(\mathrm{R})$, let C_p be the union of rays emanating from p and intersecting conv(R), that is, C_p is the cone with apex p spanned by conv(R). $C_p - p$ denotes the cone C_p translated by the vector $\vec{po}$, so its apex lies at the origin. Set $S_p := (C_p - p) \cap \mathbb{S}^2$, that is, the set of directions u such that the ray emanating from p in direction u intersects conv(R). The following lemma is straightforward.

LEMMA 3.1. *For any point $p \in \mathbb{R}^3 \setminus \mathrm{conv}(\mathrm{R})$, $\mathrm{K}_p = S_p \cup -S_p$.*

Hence,

$$\Sigma = \mathbb{S}^2 \setminus \bigcup_{b \in \mathrm{B}} \mathrm{int}(S_b \cup -S_b) = \mathrm{cl}\left(\mathbb{S}^2 \setminus \bigcup_{b \in \mathrm{B}} (S_b \cup -S_b)\right).$$

The second equality follows from Assumption (A2). Let $\mathrm{K} := \bigcup_{b \in \mathrm{B}} S_b$. Then

$$\Sigma = \mathrm{cl}(\mathbb{S}^2 \setminus (\mathrm{K} \cup -\mathrm{K})).$$

Hurtado et al. [2003] have proved that the complexity of Σ is $O(n^2 \alpha(n))$. Here we present a topological property of Σ that will be useful in developing the algorithm for testing its emptiness, but we first prove the following property of S_p.

LEMMA 3.2. *Let p and q be two points in $\mathbb{R}^3 \setminus \mathrm{conv}(\mathrm{R})$. $S_q \subseteq S_p$ if and only if $p \in C_q$ and the segment pq does not intersect* conv(R). (*See Figure* 1(i).)

PROOF. Suppose $p \in C_q$ and the segment pq does not intersect conv(R). Let u be a direction in C_q, that is, the ray emanating from q in direction u intersects conv(R). Let ρ be the ray emanating from p in direction u. Then, $\rho \subseteq C_q$ since $p \in C_q$. Note that $C_q \setminus \mathrm{conv}(\mathrm{R})$ consists of two connected components, one of which is bounded and contains both p and q. Since ρ is unbounded and emanates from p, it must intersect conv(R), thereby implying that $u \in S_p$, as claimed.

Conversely, suppose $p \in C_q \setminus \mathrm{conv}(\mathrm{R})$ and pq intersects conv(R), then the ray emanating from p in direction $\vec{qp}$ does not intersect conv(R), thereby implying that $S_q \not\subseteq S_p$. Next, assume that $p \notin C_q$. Let ℓ be the line pq. If ℓ meets conv(R), then $\vec{qp}$ misses conv(R) because $p \notin C_q$. So the complementary ray of ℓ must intersect conv(R), proving that $S_q \not\subseteq S_p$. So we may assume that ℓ does not intersect conv(R).

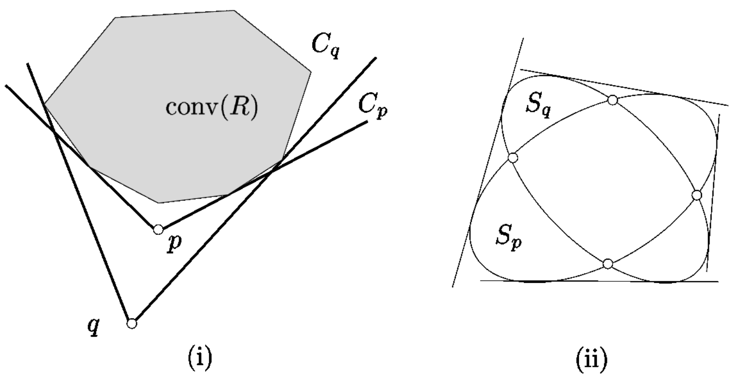

FIG. 1. (i) $p \in C_q$, so $S_q \subseteq S_p$; (ii) S_p and S_q cross.

Without loss of generality, we assume that ℓ is the x-axis and p lies to the right of q. Let π be a plane containing ℓ and intersecting conv(R). Identify π with the xy-plane and assume it meets conv(R) in the half-plane $y > 0$. Let u be the direction of the right tangent of conv(R)$\cap\pi$ from q. Then, the ray from p in direction u does not intersect conv(R), implying that $u \in S_q \setminus S_p$. Hence, $S_q \not\subseteq S_p$. In fact, we can prove an even stronger claim in this case, which we will need in Lemma 3.6 below. Let γ_p (respectively, γ_q) be the set of directions u for which the ray emanating from p (resp. q) in direction u and lying in the plane π intersect conv(R). We have already shown that $\gamma_q \not\subseteq \gamma_p$. Similarly, we can argue that the direction of the left tangent of conv(R)$\cap\pi$ from p does not belong to γ_q. Hence, $\gamma_p \not\subseteq \gamma_q$, and γ_p and γ_q are not nested. $\square$

Consider a collection S of closed, simply connected regions on a topological disk or a sphere. We say that $A, B \in \mathrm{S}$ *cross* if both $A \setminus B$ and $B \setminus A$ are disconnected; see Figure 1(ii). S is considered a *family of pseudo-disks* if no two of its members cross. Let $\mathrm{S} := \{S_b \mid b \in \mathrm{B}\}$. For simplicity, we assume that ∂S_p and ∂S_q, for any $p \neq q \in \mathrm{B}$, intersect transversally, if at all.

LEMMA 3.3. S *is a family of pseudo-disks.*

PROOF. It is sufficient to argue, for $p, q \in \mathrm{B}$, that S_p and S_q do not cross. Suppose S_p and S_q cross and thus their boundaries intersect in at least four points; see Figure 1(ii). (Since S_p and S_q are closed curves, they cannot intersect three times.) Then there are at least four planes, say, $\pi_1, \ldots, \pi_4$, through the origin that are common outer tangents to the cones $C_p - p$ and $C_q - q$. (Each of the two cones has its apex at the origin.) Let n_i be the outer normal of π_i, that is, the vector orthogonal to π_i and pointing to the half-space that does not contain $C_p - p$ and $C_q - q$.

We claim that, for each π_i, there exists a plane π_i' with outer normal n_i, passing through p and q, and tangent to C_p and C_q. Indeed consider two planes parallel to π_i: π_i' passing through p and π_i'' passing through q. Since the oriented plane π_i passes through the origin and is tangent to $C_p - p$, π_i' is tangent to C_p, and thus to conv(R), and its outer normal is n_i. Similarly, π_i'' must be tangent to conv(R) as well with n_i as its outer normal. This is impossible unless π_i' and π_i'' are the same oriented plane because a convex polytope has a unique tangent plane for every outer normal. This completes the proof of the claim.

We thus have four planes, each passing through the line pq and tangent to conv(R), but there are at most two such planes, a contradiction. Hence, S is a family of pseudo-disks. $\square$

3.2. THE ALGORITHM. We first describe a simple near-quadratic algorithm for computing Σ and then an $O(n \log^4 n)$-time procedure for testing whether $\Sigma = \emptyset$.

Compute S_b, for each $b \in \mathrm{B}$, identifying the facets of C_b (or equivalently the facets of conv($\mathrm{R} \cup \{b\}$) incident upon b) by a brute-force traversal of ∂(conv(R)). This yields, in $O(n^2)$ time, the family $\mathrm{S} = \{S_b \mid b \in \mathrm{B}\}$ of n convex spherical polygons of at most n edges each, for a total of $O(n^2)$ edges. Since S is a family of pseudo-disks, any two boundaries cross at most twice, and the arrangement $\mathrm{A(S)}$ has at most $2\binom{n}{2} = O(n^2)$ boundary intersection vertices. Thus, it has total complexity $O(n^2)$ and can be computed in $O(n^2 \log n)$ time by a standard procedure [Agarwal and Sharir 2000]. The union $\mathrm{K} = \bigcup S_b$ can be extracted from the arrangement in $O(n^2)$ time by a single traversal. Next, we can compute $-\mathrm{K}$ in additional $O(n^2)$ time. Finally, we compute $\mathrm{K} \cup -\mathrm{K}$ by identifying the intersection points of $\partial\mathrm{K}$ and $-\partial\mathrm{K}$ and then traversing the overlay of the two regions. Since $\mathrm{K} \cup -\mathrm{K}$ has $O(n^2\alpha(n))$ vertices [Hurtado et al. 2003], the total time spent in this step is $O(n^2\alpha(n) \log n)$. Finally, we set $\Sigma = \mathrm{cl}(\mathbb{S}^2 \setminus (\mathrm{K} \cup -\mathrm{K}))$. Thus, we have proven the following.

THEOREM 3.4. *Given two sets of points* R *and* B *in* $\mathbb{R}^3$*, each of cardinality* n*, we can compute the set of all minimal prism separators in* $O(n^2\alpha(n) \log n)$ *time.*

Next, we show that we can detect the emptiness of Σ in $O(n \log^4 n)$ time by computing an implicit representation of K and deciding whether $\mathbb{S}^2 = \mathrm{K} \cup -\mathrm{K}$. Using the Dobkin–Kirkpatrick hierarchy of convex polyhedra [Dobkin and Kirkpatrick 1990] on conv(R) and noting that it can be viewed as an implicit hierarchical representation of C_p, for any $p \in \mathbb{R}^3$ (see Dobkin and Kirkpatrick [1990]), we have the following tool needed in the algorithms below.

LEMMA 3.5. *We can preprocess* conv(R) *in* $O(n \log n)$ *time so that the following operations can be performed in* $O(\log n)$ *time.*

(i) *Given a direction* $u \in \mathbb{S}^2$ *and a point* $p \in \mathbb{R}^3$*, determine whether* $u \in S_p$*.*

(ii) *Given a direction* $u \in \mathbb{S}^2$ *and a point* $p \in \mathbb{R}^3$*, determine the at most two great circles passing through* u *and tangent to* S_p*.*

Note that (i) is equivalent to testing whether the line through p in direction u meets conv(R) and (ii) is equivalent to computing the planes tangent to conv(R) and containing the line through p in direction u. Using the above lemma, we prove the following.

LEMMA 3.6. *We can preprocess* conv(R) *in* $O(n \log n)$ *time so that for any two points* $p, q \in \mathbb{R}^3$*, we can determine in* $O(\log^2 n)$ *time which of the following mutually exclusive situations takes place:*

(i) S_p *and* S_q *are disjoint,*

(ii) $S_p \subset S_q$ *or* $S_q \subset S_p$*, or*

(iii) ∂S_p *and* ∂S_q *cross at two points.*

In the third case, we can compute the two intersections in the same amount of time.

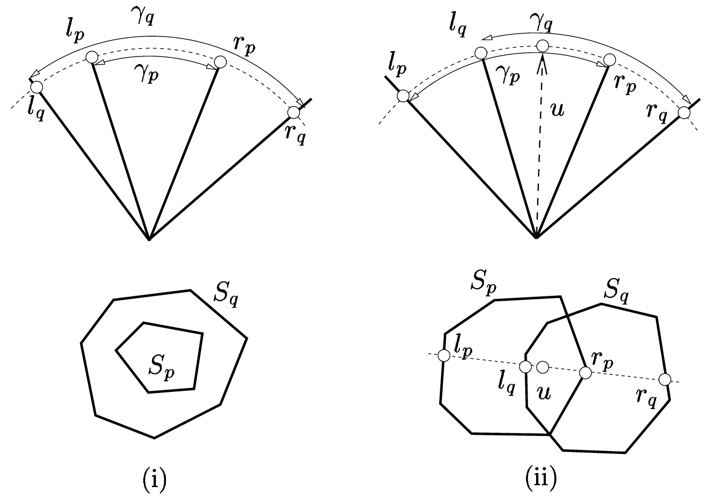

FIG. 2. Computing the intersection points of ∂S_p and ∂S_q; the top row shows γ_p and γ_q in the plane π, and the bottom row depicts the central projection of S_p and S_q: (i) $\gamma_p \subseteq \gamma_q$; (ii) γ_p and γ_q partially overlap.

PROOF. We construct in $O(n \log n)$ time the Dobkin–Kirkpatrick hierarchy on conv(R). By Lemma 3.2, case (2) occurs if and only if the line pq meets conv(R), while the segment pq does not. This can be tested in $O(\log n)$ time using the hierarchy.

On the other hand, S_p and S_q are disjoint if and only if $C_p - p$ and $C_q - q$ are disjoint. As mentioned above, the Dobkin–Kirkpatrick hierarchy on conv(R) can be used as a hierarchical representation of $C_p - p$ and $C_q - q$. It thus follows immediately, given the results of Dobkin and Kirkpatrick [1990], that we can determine in $O(\log^2 n)$ time whether $C_p - p$ and $C_q - q$ are disjoint. This handles case (i). Henceforth we assume that S_p and S_q are neither disjoint nor nested and thus their boundaries intersect twice. (Recall that our assumption on transversal intersection excludes the possibility of ∂S_p and ∂S_q touching each other at one point.) Moreover, the failed disjointness test returns a *witness direction* $u \in \mathrm{int}(S_p) \cap \mathrm{int}(S_q)$; refer to Figure 2.

Consider the plane π passing through pq and parallel to u (pq and u cannot be parallel) and the great circle γ on $\mathbb{S}^2$ parallel to it. Since γ passes through $u \in \mathrm{int}\, S_p \cap \mathrm{int}\, S_q$, it must intersect S_p and S_q in two arcs $\gamma_p := l_p r_p = S_p \cap \gamma$ and $\gamma_q := l_q r_q = S_q \cap \gamma$ since $u \in \gamma_p \cap \gamma_q$. The two arcs cannot be nested by Lemma 3.2, unless the line pq meets conv R outside the segment pq, which would contradict the assumption that we are in case (iii). Hence, the two arcs overlap partially (note that they cannot cover the circle as they share a point and each is strictly smaller than half a great circle). Therefore, up to symmetries there is only one possible ordering of the five points along γ, namely l_p, l_q, u, r_p, r_q. In particular, $l_p \notin S_q$, $r_p \in S_q$, $l_q \in S_p$, and $r_q \notin S_p$. Therefore, these four points partition each of ∂S_p, ∂S_q into two convex arcs ∂S_p^+, ∂S_p^- and ∂S_q^+, ∂S_q^- so that ∂S_p^+, ∂S_q^+ intersect at one point and ∂S_p^-, ∂S_q^- intersect at one point. By traversing a path in the Dobkin–Kirkpatrick hierarchy on conv(R) and using Lemma 3.5 at each node on the path to determine which child of that node should be visited next, we can determine the intersection point of ∂S_p^+ and ∂S_q^+, and the intersection point of ∂S_p^- and ∂S_q^- in $O(\log^2 n)$ time. □

We now describe how to detect whether Σ is empty. To simplify the presentation, we view $\mathbb{S}$ as a set of planar convex polygons; obvious adjustments have to be made to handle the fact that they lie on a sphere. For any subset $X \subseteq \mathbb{S}$, we represent each connected component of $\partial(\bigcup X)$ as a circular sequence of maximal x-monotone arcs, each lying on the top or the bottom boundary of a single polygon $S_b \in X$. For each arc γ, we store:

(i) the coordinates of its endpoints,
(ii) the point $b \in \mathbb{B}$ such that $\gamma \subset \partial S_b$,
(iii) the feature of conv($\mathbb{R}$) that determines γ, and
(iv) a bit specifying whether γ lies on the top or the bottom boundary of ∂S_b.

We refer to γ as a *top* arc in the former case and as a *bottom* arc in the latter case. Let X^* denote the set of arcs in the implicit representation of X. The endpoints of arcs in X^* are the leftmost and the rightmost points of polygons in X and the intersection points between the boundaries of two polygons of X that appear on $\partial(\bigcup X)$. Since there are $O(|X|)$ such points (see Lemma 3.3 and Kedem et al. [1986]), our implicit representation requires $O(n)$ space. Suppose we have such an implicit representation X^*, Y^* for two subsets $X, Y \subseteq \mathbb{S}$. Then we can compute the implicit representation $(X \cup Y)^*$ of the union of polygons in $X \cup Y$ by a sweep-line algorithm, as shown in Kedem et al. [1986]. The primitive steps in this procedure are:

—computing the intersection points between two arcs $\gamma \in X^*$ and $\gamma' \in Y^*$, and
—determining whether the endpoint of one arc lies below or above another arc.

Using Lemma 3.6, we can compute such intersection points and test such above/below relationships in time $O(\log^2 n)$, so the sweep-line algorithm takes $O((|X| + |Y|)\log^3 n)$ time. Hence, using a divide-and-conquer technique, we can compute an implicit representation of $\mathbb{K}$ in $O(n \log^4 n)$ time. After having computed $\mathbb{K}^*$, we can compute the implicit representation $(-\mathbb{K})^*$ of $-\mathbb{K}$ in linear time.

We would like to perform another sweep to compute an implicit representation of $\mathbb{K} \cup (-\mathbb{K})$. Unfortunately, ∂S_p and $-\partial S_q$ can intersect as many as a linear number of times, so we cannot compute the same implicit representation of Σ. Instead, we detect the emptiness of Σ using the following observation. If $\Sigma = \text{cl}(\mathbb{S}^2 \setminus (\mathbb{K} \cup (-\mathbb{K}))) \neq \emptyset$, then the leftmost point of a connected component of Σ is either an endpoint of an arc in $\mathbb{K}^* \cup (-\mathbb{K})^*$ or an intersection point of a top arc of $\mathbb{K}^*$ (respectively $(-\mathbb{K})^*$) with a bottom arc of $(-\mathbb{K})^*$ (respectively $\mathbb{K}^*$). We first check in a total of $O(n \log^2 n)$ time, by a sweep-line algorithm, whether an endpoint of any arc in $\mathbb{K}^*$ lies outside $-\mathbb{K}$, or vice versa. Next, we observe that, provided the previous test did not find a point in Σ, a sweep-line algorithm can detect whether $\partial\mathbb{K}$ and $\partial(-\mathbb{K})$ intersect. It is sufficient to only test the top (respectively bottom) arcs of $\mathbb{K}^*$ against the bottom (respectively, top) arcs of $-\mathbb{K}^*$. We stop the sweep as soon as such an intersection is found because we can conclude that $\Sigma \neq \emptyset$. If no intersection is found, we conclude that $\Sigma = \emptyset$. The intersection test between a top arc of $\mathbb{K}^*$ and a bottom arc of $-\mathbb{K}^*$ (or vice-versa) is particularly simple, as one of them is convex and the other is concave. However, there arcs are represented implicitly, so using the Dobkin–Kirkpatrick hierarchy and following the same approach as in Lemma 3.6, we detect such an intersection in $O(\log^2 n)$

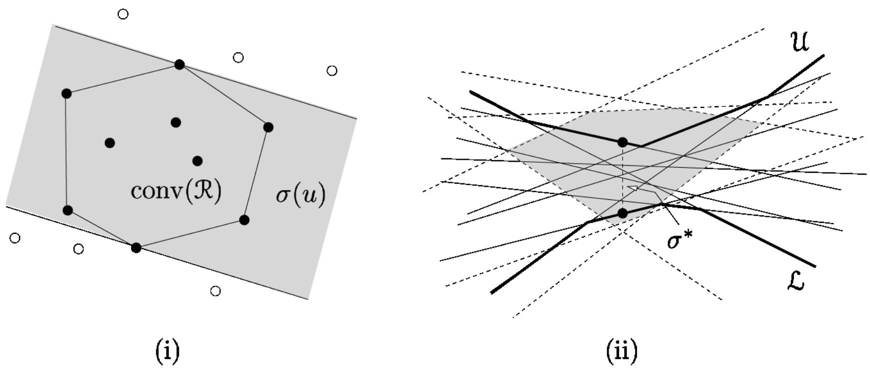

FIG. 3. Separating by a slab: (i) A slab separating R from B in the primal setting. (ii) Dual setting: solid lines form R^*, dashed lines form B^*, vertical segment is σ^*, and the shaded region is the cell of $A(B^*)$ that contains the vertical segment σ^*.

time. Thus the detection of an intersection of K^* and $(-K)^*$ can be implemented in $O(n \log^2 n)$ time. Putting everything together, we obtain the following.

THEOREM 3.7. *Given two sets of points* R *and* B *in* $\mathbb{R}^3$*, each of cardinality* n*, we can determine in* $O(n \log^4 n)$ *time the existence of a prism that contains* R *but whose interior is disjoint from* B *and return such a prism if it exists.*

4. *Slab Separability*

We now let F be the family of all *slabs*, that is, closed regions delimited by two parallel planes. For a given direction $u \in \mathbb{S}^2$, there is a unique minimal slab container $\sigma(u)$: the planes bounding $\sigma(u)$ are normal to u and support conv(R); one of these two planes lies above conv(R) and the other lies below conv(R). (Vertical slabs can be handled separately—use a two-dimensional algorithm for separating planar point sets (xy-projections of R and B) by a strip [Hurtado et al. 2004]. Hereafter, we will assume that the slab separator is nonvertical.) If $\sigma(u)$ is a separator, then none of the points in B lie in the interior of $\sigma(u)$. Using a standard duality transform, we can map R (respectively, B) to a set of planes R^* (respectively, B^*). The dual σ^* of a minimal separating slab is a vertical segment one of whose endpoints lies on the lower envelope L of $A(R^*)$ and the other on the upper envelope U of $A(R^*)$. Moreover, σ^* lies completely inside a (closed) cell of $A(B^*)$ as it does not intersect any plane of B^*. See Figure 3. Based on these observations, we proceed as follows:

Let Z be the set of cells in $A(B^*)$ that intersect both L and U. Since each cell $C \in Z$ intersects the convex surface L, $\sum_{C \in Z} |C| = O(n^2 \log n)$ [Aronov et al. 1993]. For a cell $C \in Z$, let $C^+ := C \cap U$ and $C^- := C \cap L$.

LEMMA 4.1. $\sum_{C \in Z}(|C^+| + |C^-|) = O(n^2)$.

PROOF. Each vertex of ∂C^+ is an intersection point of an edge of U and a plane of B^*, or a face of U and a line formed by the intersection of two planes of B^*. Since U has $O(n)$ edges, there are $O(n^2)$ vertices of the first type. As an intersection line of two planes of B^* intersects the convex surface U at most twice, there are $O(n^2)$ intersections of the second type as well. Sets C^- are treated symmetrically. □

For a subset $\mathrm{N} \subseteq \mathrm{B}$, let $\mathrm{Z}(\mathrm{N}^*)$ denote the set of cells in $\mathrm{A}(\mathrm{N}^*)$ that intersect both L and U, and let $\mathrm{Z}^\nabla(\mathrm{N}^*)$ denote the bottom-vertex triangulation, as defined in de Berg et al. [1995], of the cells in $\mathrm{Z}(\mathrm{B}^*)$. We compute $\mathrm{Z}^\nabla(\mathrm{B}^*)$ as well as C^+, C^-, for each cell $C \in \mathrm{Z}(\mathrm{B}^*)$, in $O(n^2 \log n)$ expected time, using a lazy randomized incremental algorithm [de Berg et al. 1995]. More precisely, we choose a random subset $\mathrm{N}^* \subseteq \mathrm{B}^*$ of $n/2$ planes and compute $\mathrm{Z}^\nabla(\mathrm{N}^*)$ recursively. For each simplex $\Delta \in \mathrm{Z}^\nabla(\mathrm{N}^*)$, we compute the subset $\mathrm{B}^*_\Delta \subseteq \mathrm{B}^*$ of planes that intersect Δ; set $n_\Delta = |\mathrm{B}^*_\Delta|$. For each Δ, we compute $\mathrm{A}(\mathrm{B}^*_\Delta)$, clipped within Δ, in $O(n^3_\Delta)$ time using a naive algorithm. Next, we traverse the cells of $\mathrm{A}(\mathrm{B}^*_\Delta)$, over all Δ, identify those cells which lie in $\mathrm{Z}(\mathrm{B}^*)$, and merge two of them if they are portions of the same cell of $\mathrm{Z}(\mathrm{B}^*)$. Finally, we compute the bottom-vertex triangulation of each cell in $\mathrm{Z}(\mathrm{B}^*)$. The total time spent in these steps is $O(n^2 \log n + \sum_{\Delta \in \mathrm{Z}^\nabla(\mathrm{N}_*)} n^3_\Delta)$. After having computed $\mathrm{Z}^\nabla(\mathrm{B}^*)$, we can also compute C^+, C^-, for each cell $C \in \mathrm{Z}(\mathrm{B}^*)$, in additional $O(n^2 \log n)$ time.

In order to compute the sets B^*_Δ efficiently, we maintain a history dag, as described in de Berg et al. [1995]. Roughly speaking, let $\mathrm{N}_0 \subseteq \mathrm{N}_1 \subseteq \cdots \subseteq \mathrm{B}^*$ be the sequence of random subsets chosen by the algorithm. For each simplex $\Delta \in \mathrm{Z}^\nabla(\mathrm{N}^*_i)$, we maintain a pointer to a simplex $\Delta' \in \mathrm{Z}^\nabla(\mathrm{N}^*_{i+1})$ if Δ and Δ' intersect. Assuming that we know the simplices of $\mathrm{Z}^\nabla(\mathrm{N}^*_i)$ that a plane $h \in \mathrm{B}^*$ intersects, we can quickly compute the simplices of $\mathrm{Z}^\nabla(\mathrm{N}^*_{i+1})$ that it intersects. The analysis in de Berg et al. [1995, Sect. 5.4] shows that the expected running time of the overall algorithm, including the time spent in computing the sets B^*_Δ, is $O(n^2 \log n)$.

Next, for each cell $C \in \mathrm{Z}$, we project C^+ and C^- onto the xy-plane, yielding two families of disjoint convex polygons. Using a sweep-line algorithm, we determine in $O((|C^+| + |C^-|) \log n)$ time whether the xy-projections of C^+ and C^- intersect. If they intersect at a point ξ, then the slab dual to the vertical segment $\mathrm{U}(\xi)\mathrm{L}(\xi)$ is a minimal separating slab, where $\mathrm{U}(\xi)$ (respectively, $\mathrm{L}(\xi)$) is the intersection point of U (respectively, L) with the vertical line through ξ. Repeating this step for all cells of Z, we can determine whether there exists a (minimal) separating slab. By Lemma 4.1, we obtain the following:

THEOREM 4.2. *Given two sets of points* R *and* B *in* $\mathbb{R}^3$, *each of cardinality* n, *a randomized algorithm can determine the existence of a separating slab and return such a slab if it exists in expected time* $O(n^2 \log n)$.

We note here that the computation can also be carried out deterministically, using the algorithm of Aronov and Iacono [2004]. We only outline the procedure here. We first compute L explicitly and then intersect every plane of B^* with L and construct the arrangement A_1 of xy-projections of these n convex curves. By the reasoning of the proof of Lemma 4.1, the size of A_1 is $O(n^2)$ and it can be computed by a sweep-line algorithm in $O(n^2 \log n)$ time. Each face in A_1 corresponds to a connected component of C^- for some $C \in \mathrm{A}(\mathrm{B}^*)$ that intersects L. A similar procedure computes the arrangement A_2 representing $\{C^+ \mid C \in \mathrm{A}(\mathrm{B}^*),\ C \cap \mathrm{U} \neq \emptyset\}$. Once we identify which faces of A_1 and A_2 correspond to the same cell C, the algorithm proceeds as above, checking for nonemptiness of $C^- \cap C^+$ for all C. We perform this identification as follows: We conceptually label each cell $C \in \mathrm{Z}$ with a bit vector identifying the side of each plane of B^* on which C lies. If one traverses A_1 along a tour visiting all vertices of its dual graph, neighboring bit vectors differ

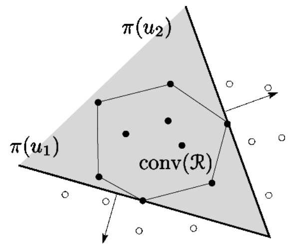

FIG. 4. Separating by a wedge.

by a single bit. We do not represent the vectors explicitly, but instead use the algorithm of Aronov and Iacono [2004] that can store them in constant amortized space per vector and detect duplicates, thereby grouping the faces corresponding to different connected components of the same C^- together. Duplicate detection can be performed in a single traversal of the data structure, so that faces are output in groups, with each group corresponding to the same bit vector. Applying the procedure to $\mathcal{A}_1$ and $\mathcal{A}_2$ together allows us to match C^- and C^+ as they correspond to faces with identical bit vectors. We omit further details.

THEOREM 4.3. *Given two sets of points $\mathcal{R}$ and $\mathcal{B}$ in $\mathbb{R}^3$, each of cardinality n, a separating slab, if it exists, can be computed deterministically in $O(n^2 \log n)$ worst-case time.*

5. *Wedge Separability*

We now let $\mathcal{F}$ be the family of *wedges*, that is, intersections of two closed half-spaces. If the two bounding planes are parallel, then the resulting wedge is a slab. Since we have already studied slab-separability, we assume that the wedges in $\mathcal{F}$ are bounded by non-parallel planes. As in the case of slabs, the two boundary planes of a minimal separating wedge support conv($\mathcal{R}$); see Figure 4. For a direction $u \in \mathbb{S}^2$, let $\pi(u)$ denote the (oriented) plane with outward normal u that supports conv($\mathcal{R}$), and let $\mathcal{B}(u) \subseteq \mathcal{B}$ denote the subset of points that lie on the same side of $\pi(u)$ as $\mathcal{R}$. A minimal wedge container can be represented by a pair of outward normals $(u_1, u_2) \in \mathbb{S}^2 \times \mathbb{S}^2$. If (u_1, u_2) is a separator, then $\pi(u_2)$ separates conv($\mathcal{R}$) and conv($\mathcal{B}(u_1)$). As we vary the direction u, $\mathcal{B}(u)$ remains the same until $\pi(u)$ passes through a point of $\mathcal{B}$. We proceed as follows to compute a minimal separating wedge, if one exists.

Let $\mathcal{N}$ be the *normal diagram* (also known as the *Gauss map*) of conv($\mathcal{R}$), that is, the subdivision of $\mathbb{S}^2$ into maximal regions so that for all directions u within each region, the plane $\pi(u)$ is tangent to the same feature (vertex, edges, or face) of conv($\mathcal{R}$). $\mathcal{N}$ can be computed in linear time from conv($\mathcal{R}$). For a point $b \in \mathcal{B}$, let $\gamma_b \subset \mathbb{S}^2$ be the locus of directions u so that $\pi(u)$ passes through b; if $b \notin \partial\,\mathrm{conv}(\mathcal{R})$, γ_b is the boundary of a convex polygon on $\mathbb{S}^2$, namely it is the set of outer normals to the planes tangent to C_b (defined in Section 3); b cannot lie on $\partial\,\mathrm{conv}(\mathcal{R})$ by our assumption of general position. The curve γ_b consists of arcs of great circles, the arcs end at points lying on the edges of $\mathcal{N}$; we call these points *breakpoints*. We compute the arrangement of $\Gamma := \{\gamma_b \mid b \in \mathcal{B}\}$. The same argument as in the proof of Lemma 3.3 implies that Γ is a family of pseudo-disks, therefore $\mathcal{A}(\Gamma)$ has $O(n^2)$ vertices and it can be computed in $O(n^2 \log n)$ time using the first algorithm

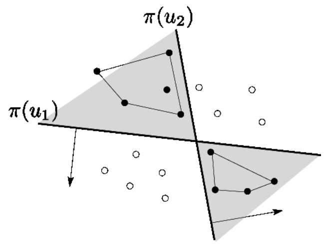

FIG. 5. Separating by a double wedge.

described in Section 3.

By construction and the above discussion, for all directions u within the same face ϕ of $\mathrm{A}(\Gamma)$, $\mathrm{B}(u)$ is the same, which we denote by $\mathrm{B}(\phi)$. Moreover, if ϕ, ϕ' are two adjacent faces of $\mathrm{A}(\Gamma)$, then $|\mathrm{B}(\phi) \oplus B(\phi')| \leq 1$, where $\oplus$ denotes symmetric difference. We wish to determine whether there is a face $\phi \in \mathrm{A}(\Gamma)$ for which $\mathrm{conv}(R)$ and $\mathrm{conv}(\mathrm{B}(\phi))$ are disjoint (i.e., weakly linearly separable). Roughly speaking, we traverse the faces of $\mathrm{A}(\Gamma)$, updating the set $\mathrm{B}(\phi)$ as we go from one face to another, and use a dynamic data structure to determine whether $\mathrm{conv}(\mathrm{R})$ and $\mathrm{conv}(\mathrm{B}(\phi))$ ever become disjoint.

In more detail, we compute a tour visiting all vertices of the dual graph of $\mathrm{A}(\Gamma)$, which is a sequence $\Phi = \langle \phi_1, \phi_2, \ldots, \phi_k \rangle$, $k = O(n^2)$, of faces of $\mathrm{A}(\Gamma)$. We traverse the sequence Φ, and at each i, we maintain $\mathrm{conv}(R) \cap \mathrm{conv}(\mathrm{B}(\phi_i))$ in a dynamic data structure. Since we know the sequence of insertions and deletions in advance and only want to determine whether $\mathrm{conv}(R) \cap \mathrm{conv}(\mathrm{B}(\phi_i))$ ever becomes empty, we can use the off-line data structure of Eppstein [1992], which can perform an update in amortized expected time $O(\log^2 n \log^2(\log n))$. He also described another data structure that can perform an update in polylogarithmic amortized deterministic time. We conclude the following.

THEOREM 5.1. *Given two sets of points* R *and* B *in* $\mathbb{R}^3$, *each of cardinality* n, *an algorithm can determine the existence of a separating convex dihedral wedge, and return such a wedge if it exists, in randomized expected time* $O(n^2 \log^2 n \log^2(\log n))$ *or in* $O(n^2\ \mathrm{polylog}(n))$ *deterministic worst-case time.*

6. *Double-Wedge Separability*

In this section, F is the family of all *double wedges*, that is, (closures of) the symmetric difference of two half-spaces bounded by nonparallel planes. Without loss of generality, we consider only those double wedges that do not contain a vertical plane.

Once again, consider the dual version of the problem, where we are required to determine, given two collections of planes R^*, B^* whether there exists a segment σ^* that intersects all of the planes in R^* but crosses none of those in B^*. Let C be a 3-dimensional cell of $\mathrm{A}(\mathrm{R}^* \cup \mathrm{B}^*)$. For a plane $h \in \mathrm{R}^* \cup \mathrm{B}^*$, let h_C^{in} (respectively, h_C^{out}) be the closed half-space bounded by h that contains (respectively, does not contain) C. Let $B_C^{\mathrm{in}} := \{h_C^{\mathrm{in}} \mid h \in \mathrm{B}^*\}$, and let $R_C^{\mathrm{out}} := \{h_C^{\mathrm{out}} \mid h \in \mathrm{R}^*\}$. If σ^* is the dual of a separating double wedge and one endpoint ξ of σ^* lies in C, then the other endpoint ζ of σ^* has to lie in $P_C := \bigcap g$, where the intersection is taken over all half-spaces $g \in B_C^{\mathrm{in}} \cup R_C^{\mathrm{out}}$. Indeed, ξ and ζ lie on the same side of all planes in B^* and on the opposite sides of all planes in R^*; see Figure 5. Hence, the segment

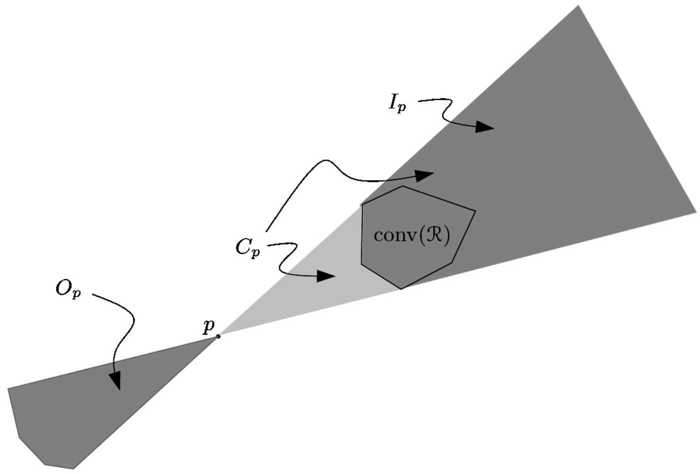

FIG. 6. The definitions of I_p and O_p.

$\xi\zeta$ does not intersect any plane of B^* and intersects all planes of R^*.

As in the previous section, consider a dynamic data structure that can maintain P_C through insertions and deletions in $B_C^{\text{in}} \cup R_C^{\text{out}}$. Traverse the cells C of $\mathrm{A}(\mathrm{R}^* \cup \mathrm{B}^*)$ and maintain P_C in this fashion. If for some cell C during the traversal P_C is nonempty we return a segment $\sigma^* = \xi\zeta$ such that $\xi \in C$ and $\zeta \in P_C$, otherwise we conclude that a segment σ^* as desired does not exist. The running time in the theorem below follows from the analysis in the preceding section.

THEOREM 6.1. *Given two sets of points* R *and* B *in* $\mathbb{R}^3$*, each of cardinality* n*, an algorithm can determine the existence of a separating dihedral double wedge and return such a double wedge if it exists in randomized expected time* $O(n^3 \log^2 n \log^2(\log n))$ *or* $O(n^3 \operatorname{polylog}(n))$ *deterministic worst-case time.*

7. *Cone Separability*

In this section, F is the family of all (closed, convex, infinite) cones. Recall that for $p \notin \mathrm{conv}(\mathrm{R})$, C_p is the cone with apex p spanned by the points in $\mathrm{conv}(\mathrm{R})$. Hereafter, we assume for simplicity that the apex of the container cone is always at a point of $\mathbb{R}^3$ strictly outside of $\mathrm{conv}(\mathrm{R})$. For any point $p \in \mathbb{R}^3 \setminus \mathrm{conv}(\mathrm{R})$, there is a unique minimal cone container (with respect to R) with apex p, namely the cone C_p, so we represent the set C of minimal cone containers by points in $\mathbb{R}^3 \setminus \mathrm{conv}(\mathrm{R})$.

Unlike the cases considered in previous sections, here we only discuss the structure of the space Σ of minimal cone separators. This investigation yields an $O(n^2)$ bound on the combinatorial complexity of Σ. We do not at this point have a near-quadratic algorithm for constructing Σ, or an efficient way of testing whether it is empty. The fastest known algorithm for computing it is roughly cubic [Hurtado et al. 2003]; see the discussion below.

7.1. STRUCTURE OF Σ. Fix a point $p \in \mathrm{B}$ and examine the set $\mathrm{K}_p := \{x \mid p \in C_x\}$. Let the *outer cone* O_p of p be the cone with apex p and antipodal to C_p. Let the *inner cone* I_p of p be C_p less the connected component of $C_p \setminus \mathrm{conv}(\mathrm{R})$ containing p. See Figure 6 and note that we have slightly abused the terminology

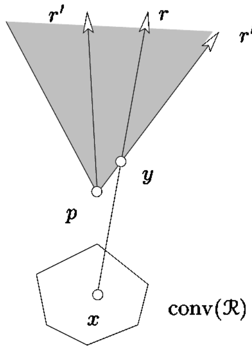

FIG. 7. Illustration to the proof of Lemma 7.1; intersection with the plane spanned by r and p is depicted; the case where r and p are collinear is easier.

in that the inner cone is not exactly a cone due to the presence of a "cap," a portion of $\partial\,\mathrm{conv}(\mathcal{R})$, on its boundary. It is easy to see that $\mathcal{K}_p = O_p \cup I_p$. In particular, putting $\mathcal{O} := \bigcup O_p$ and $\mathcal{I} := \bigcup I_p$, where the union is taken over $p \in \mathcal{B}$, Σ is the complement of $\mathcal{O} \cup \mathcal{I}$ and we focus on this latter set.

We call a ray r *outgoing* if it emanates from a point x in $\mathrm{conv}(\mathcal{R})$ and *strictly outgoing* if x lies in the interior of $\mathrm{conv}(\mathcal{R})$. To avoid excessive notation, we think of a ray as being traversed from its origin to infinity. A *tail* of a ray r is any ray $r' \subseteq r$.

LEMMA 7.1. *The following properties of outgoing rays and inner and outer cones hold:*

(*a*) *If an outgoing ray meets an outer cone O_p, it stays in O_p.*
(*b*) *If a strictly outgoing ray meets an outer cone O_p, it enters the interior of O_p and remains in the interior of O_p.*
(*c*) *An outgoing ray starts inside every inner cone I_p and, if it leaves I_p, it never re-enters it.*
(*d*) *An outer cone is a union of tails of outgoing rays, disjoint except at its apex.*
(*d'*) *The boundary of an outer cone is a union of tails of nonstrictly outgoing rays, disjoint except at its apex.*
(*e*) *An inner cone is a disjoint union of outgoing rays.*
(*e'*) *The boundary of an inner cone, except for its cap* (*i.e.*, $(\partial I_p) \setminus \mathrm{conv}(\mathcal{R})$)*, is a disjoint union of nonstrictly outgoing rays.*

PROOF. We start with (c), as it is trivially true: An outgoing ray r starts in $\mathrm{conv}(\mathcal{R})$ by definition and every inner cone I_p includes $\mathrm{conv}(\mathcal{R})$ by definition. So r must start in I_p and if it ever leaves this convex set, it can of course never re-enter it.

As for (a) and (b), an outer cone O_p is the union of all rays with origin p antipodal to those that meet $\mathrm{conv}(\mathcal{R})$. Consider an outgoing ray r originating at $x \in \mathrm{conv}(\mathcal{R})$ and first encountering O_p at a point $y \in \partial O_p$. Let r' be the ray emanating from p in direction $\vec{xp}$ (directed away from x) and r'' be the ray py; refer to Figure 7. By definition of O_p, $r' \subset O_p$. By convexity of O_p, $r'' \subset O_p$ (in fact, $r'' \subset \partial O_p$), and thus the angular wedge W defined by r' and r'' is also contained in O_p. Since the line containing r' passes through x, r' and the tail of r starting at y are disjoint, thereby

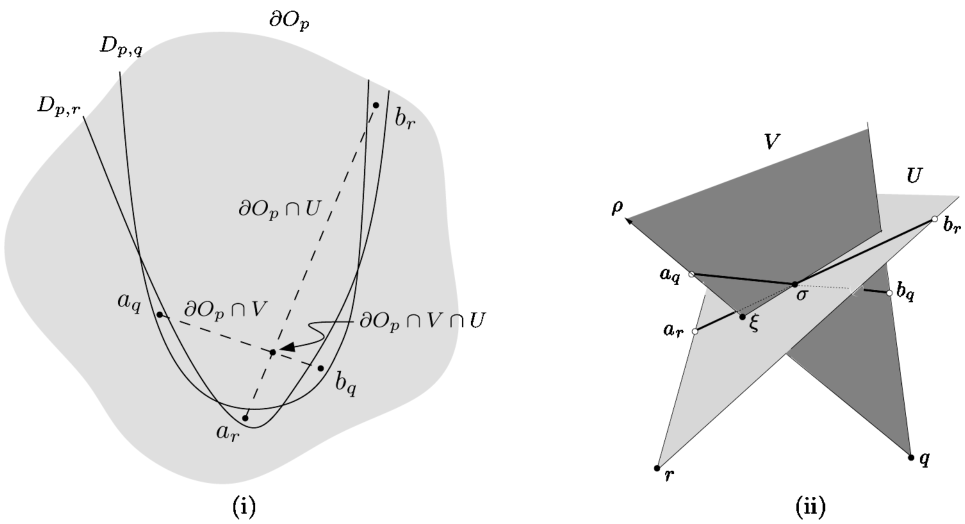

FIG. 8. Proof of Lemma 7.2.

implying that the latter lies inside $W \subseteq O_p$. This completes the argument for (a). If r is strictly outgoing, x lies in the interior of conv(R), thus r' lies completely in interior of O_p (except for p) and moreover W lies in int(O_p) except for r''. Thus r stays in the interior of W after entering it at y, completing the proof of (b).

Properties (d), (e), and (e′) hold by definition. Finally, (d′) holds almost by definition: O_p is the cone antipodal to C_p which in turn is the cone with apex p spanned by conv(R). So ∂O_p consists of such rays that their antipodal rays touch but do not penetrate conv(R), as claimed. □

We now prove the main structural property of O and I. For a pair of points $p, q \in \mathrm{B}$, let $D^O_{p,q} := \partial O_p \cap O_q$, $D^I_{p,q} := \partial O_p \cap I_q$, $E^O_{p,q} := \partial I_p \cap O_q$, and $E^I_{p,q} := \partial I_p \cap I_q$. Set

$$\mathbb{D}^O_p := \{D^O_{p,q} \mid q \in \mathrm{B} \setminus \{p\}\}, \qquad \mathbb{D}^I_p := \{D^I_{p,q} \mid q \in \mathrm{B} \setminus \{p\}\},$$
$$\mathbb{E}^O_p := \{E^O_{p,q} \mid q \in \mathrm{B} \setminus \{p\}\}, \qquad \mathbb{E}^I_p := \{E^I_{p,q} \mid q \in \mathrm{B} \setminus \{p\}\}.$$

LEMMA 7.2. *For any point $p \in \mathrm{B}$, $\mathbb{D}^O_p$ is a family of pseudo-disks on ∂O_p. The same is true for $\mathbb{D}^I_p$ on ∂O_p, and for each of $\mathbb{E}^O_p$ and $\mathbb{E}^I_p$ on ∂I_p.*

PROOF. To prove the first statement, it is sufficient to argue that $D^O_{p,q}$ and $D^O_{p,r}$ do not cross, for $q \neq r$, as defined in Section 3. Suppose to the contrary they do cross. Then there exist two-dimensional cones $V \subseteq O_q$ with apex q and $U \subseteq O_r$ with apex r, such that the two endpoints a_q, b_q of the curve $\partial O_p \cap V \subset D^O_{p,q}$ lie outside $D^O_{p,r}$, the two endpoints a_r, b_r of the curve $\partial O_p \cap U \subset D^O_{p,r}$ lie outside $D^O_{p,q}$ and the intersection point σ of $\partial O_p \cap V \cap U$ lies inside $D^O_{p,q} \cap D^O_{p,r}$; refer to Figure 8(i). It easily follows, without loss of generality, that a boundary ray ρ, say, of V intersects U at a point ξ that lies between the origin q of ρ and the point $a_q := \rho \cap \partial O_p$. (See Figure 8(ii).) Note that q must lie outside O_r, for otherwise $O_q \subset O_r$ and ∂O_q does not cross ∂O_r, contrary to the assumption. On the other hand, $\xi \in U \subseteq O_r$. Since $a_q \in \partial O_p$ lies outside O_r, the ray ρ is outside O_r at q

and a_q and inside it at ξ, contradicting Lemma 7.1(a), as ρ is a tail of an outgoing ray, by Lemma 7.1(d).

Next, consider the family $\mathbb{E}_p^O$. Assume for the sake of contradiction that $E_{p,q}^O$ and $E_{p,r}^O$ cross. This implies, as above, the existence of two-dimensional cones $V \subseteq I_q$ with apex q and $U \subseteq I_r$ with apex r, such that the two endpoints a_q, b_q of the curve $\partial O_p \cap V$ lie outside $E_{p,r}^O$ (and thus outside I_r), the two endpoints a_r, b_r of the curve $\partial O_p \cap U$ lie outside $E_{p,q}^O$ (and thus outside I_q), and the intersection point σ of $\partial O_p \cap V \cap U$ lies in $E_{p,q}^O \cap E_{p,r}^O$. Lemma 7.1(a) implies that the tails of the two boundary rays of V (respectively, U) originating at a_q and b_q (respectively, a_r and b_r) are completely disjoint from I_r (respectively, from I_q). Thus the endpoints α_q, β_q (respectively, α_r, β_r) of the arc $\mathbb{S}^2 \cap (V - q)$ (respectively, $\mathbb{S}^2 \cap (U - r)$) lie outside S_r (respectively S_q); see definitions in Section 3. On the other hand, consider the intersection $\gamma := V \cap U$, which is a ray. Clearly, the point on $\mathbb{S}^2$ corresponding to the direction of γ lies on both segments $\alpha_q\beta_q$ and $\alpha_r\beta_r$ and thus in $S_q \cap S_r$. In other words, S_q and S_r cross on $\mathbb{S}^2$, contradicting Lemma 3.3.

The proof for the remaining two families, $\mathbb{D}_p^I$ and $\mathbb{E}_p^I$, is similar. □

We are now in position to prove the main result of this section, a tight bound on the complexity of Σ.

THEOREM 7.3. *The complexity of the space* Σ *of all minimal separator cones is* $O(n^2)$.

PROOF. Since each O_p and I_p is a polyhedral cone, our general-position assumption implies that every 1- or 2-dimensional face of Σ is incident upon at least one vertex of Σ (that is not a cone vertex). Therefore, by the general-position assumption, the complexity of Σ is proportional to the number of its vertices. As Σ is the complement of the union of the cones I_p, O_p, its vertices are of three types:

(i) vertices of the cones,
(ii) vertices formed by the intersections of two cone surfaces, and
(iii) vertices formed by the interaction of three cone surfaces.

There are only $O(n)$ vertices of type (i). A type (ii) vertex if formed when a cone edge e meets another cone on $\partial\Sigma$. An edge of an inner cone is an outgoing ray and an edge of an outer cone is a tail of an outgoing ray. Hence, once e enters another outer cone, it never appears on $\partial\Sigma$ again, and also, before the last time e leaves another inner cone it never appears on $\partial\Sigma$. To summarize, e can only appear on $\partial\Sigma$ in at most one connected portion and contributes at most two vertices of the second type to $\partial\Sigma$. As the number of cone edges is quadratic, so is the number of such vertices.

It remains to count the vertices that lie on three cone surfaces. Suppose such a vertex ξ occurs on the boundary of two outer cones O_p, O_q and of another cone X_r, which can be either an inner or an outer cone. If X_r is the outer (respectively, inner) cone O_r (respectively, I_r), then ξ is a vertex of the union of pseudo-disks in $\mathbb{D}_r^O$ (respectively, $\mathbb{D}_r^I$). By a result of Kedem et al. [1986], each of $\bigcup \mathbb{D}_r^O$ and $\bigcup \mathbb{D}_r^I$ has $O(n)$ vertices. Hence, the number of vertices that lie on three cone surfaces two of which are outer-cone surfaces is $O(n^2)$. Similarly, the number of vertices that lie on three cone surfaces two of which are inner is also $O(n^2)$. This completes the proof of the theorem. □

7.2. THE (LACK OF AN EFFICIENT) ALGORITHM. As already mentioned, we do not have a near-quadratic algorithm for computing Σ. A near-cubic one can be obtained by computing the inner and outer cones explicitly in roughly quadratic total time. This yields a collection of n convex polyhedra each of complexity $O(n)$. As observed in Hurtado et al. [2003], their union can now be computed by the algorithm of Aronov et al. [1997] in roughly cubic randomized expected time. Any attempt to tune the union-of-polyhedra algorithm of Aronov et al. [1997] to this case that represents the cones explicitly is doomed to failure, as that algorithm starts by computing the intersection of every face of every polyhedron with the remaining polyhedra and the total complexity of all such intersections can be cubic in the worst case for the problem under consideration. A natural attempt to circumvent this is to represent the intersections implicitly, as was done in Section 3. Indeed, this would be possible and a near-quadratic algorithm would result if one could compute in, say, polylogarithmic time, the at most two (by Lemma 7.2) points of intersection of any three cones. This would allow one to compute, as in Section 3, by recursive pairwise sweep-merge $\bigcup \mathbb{D}_p^O$ and $\bigcup \mathbb{D}_p^I$, for every p, in total time roughly proportional to the maximum number of such intersection points on the union; the same would apply to the sets $\mathbb{E}_p^O$ and $\mathbb{E}_p^I$. Testing, for every p, whether $\bigcup \mathbb{D}_p^I \cup \bigcup \mathbb{D}_p^O = \partial O_p$ as in Section 3 (and the analogous operations for the inner cone surfaces) would allow efficient emptiness testing for Σ.

8. *Conclusion*

In this article, we studied the problem of separating two point sets by a solid body in $\mathbb{R}^3$ and presented improved algorithms for a number of cases. However, several problems remain open. One obvious problem is to develop a near-quadratic algorithm for cone separability. The only missing step is a data structure that can preprocess a convex polytope in $\mathbb{R}^3$ so that for three query points p, q, r one can determine in polylogarithmic time the intersection points of ∂C_p, ∂C_q, and ∂C_r.

There has been extensive work on computing "large-margin" hyperplane separators, but little is known for such separators by other solid bodies. Can our algorithms be extended to find maximum-margin separators? What about handling outliers and higher-dimensional data?

ACKNOWLEDGMENTS. The authors thank Carlos Seara for bringing the problem to our attention and for useful discussions and the two referees for their helpful comments.

REFERENCES

AGARWAL, P. K., AND SHARIR, M. 2000. Arrangements and their applications. In *Handbook of Computational Geometry*, J.-R. Sack and J. Urrutia, Eds. Elsevier Science Publishers B.V., North-Holland, Amsterdam, The Netherlands, pp. 49–119.

AGARWAL, P. K., AND SURI, S. 1998. Surface approximation and geometric partitions. *SIAM J. Comput. 27*, 1016–1035.

ARKIN, E., HURTADO, F., MITCHELL, J., SEARA, C., AND SKIENA, S. 2001. Some lower bounds on geometric separability problems. In *Proceedings of the 11th Fall Workshop on Computational Geometry*.

ARONOV, B., AND IACONO, J. 2004. Detecting duplicates among similar bit vectors. In *Proceedings of the 14th Fall Workshop on Computational Geometry*.

ARONOV, B., PELLEGRINI, M., AND SHARIR, M. 1993. On the zone of a surface in a hyperplane arrangement. *Disc. Comput. Geom. 9*, 177–186.

ARONOV, B., SHARIR, M., AND TAGANSKY, B. 1997. The union of convex polyhedra in three dimensions. *SIAM J. Comput. 26*, 1670–1688.

BOISSONNAT, J.-D., CZYZOWICZ, J., DEVILLERS, O., URRUTIA, J., AND YVINEC, M. 2000. Computing largest circles separating two sets of segments. *Internat. J. Comput. Geom. Appl. 10*, 41–54.

BRÖNNIMANN, H., AND GOODRICH, M. T. 1995. Almost optimal set covers in finite VC-dimension. *Disc. Comput. Geom. 14*, 463–479.

DE BERG, M., DOBRINDT, K., AND SCHWARZKOPF, O. 1995. On lazy randomized incremental construction. *Disc. Comput. Geom. 14*, 261–286.

DOBKIN, D. P., AND KIRKPATRICK, D. G. 1990. Determining the separation of preprocessed polyhedra—A unified approach. In *Proceedings of the 17th International Colloquium Automata Language Programming*. Lecture Notes in Computer Science, vol. 443. Springer-Verlag, New York, 400–413.

EDELSBRUNNER, H., AND PREPARATA, F. P. 1988. Minimum polygonal separation. *Inf. Comput. 77*, 218–232.

EPPSTEIN, D. 1992. Dynamic three-dimensional linear programming. *ORSA J. Comput. 4*, 360–368.

FEKETE, S. 1992. On the complexity of min-link red-blue separation. Unpublished manuscript.

HASTIE, T., TIBSHIRANI, R., AND FRIEDMAN, J. 2001. *The Elements of Statistical Learning*. Springer-Verlag, Berlin, Germany.

HURTADO, F., MORA, M., RAMOS, P. A., AND SEARA, C. 2004. Separability by two lines and by nearly straight polygonal chains. *Disc. Appl. Math. 144*, 110–122.

HURTADO, F., NOY, M., RAMOS, P. A., AND SEARA, C. 2001. Separating objects in the plane with wedges and strips. *Disc. Appl. Math. 109*, 109–138.

HURTADO, F., SEARA, C., AND SETHIA, S. 2003. Red-blue separability problems in 3D. In *Proceedings of the 3rd International Workshop on Computational Geometry and Applications*.

KEDEM, K., LIVNE, R., PACH, J., AND SHARIR, M. 1986. On the union of Jordan regions and collision-free translational motion amidst polygonal obstacles. *Disc. Comput. Geom. 1*, 59–71.

MEGIDDO, N. 1983. Linear-time algorithms for linear programming in R^3 and related problems. *SIAM J. Comput. 12*, 759–776.

MITCHELL, J. S. B. 1993. Approximation algorithms for geometric separation problems. Tech. Rep. Dept. Applied Mathematics, SUNY Stony Brook, NY.

O'ROURKE, J., KOSARAJU, S. R., AND MEGIDDO, N. 1986. Computing circular separability. *Disc. Comput. Geom. 1*, 105–113.

VAPNIK, V. 1996. *The Nature of Statistical Learning Theory*. Springer-Verlag, New York.

RECEIVED SEPTEMBER 2004; ACCEPTED NOVEMBER 2005