Space-Time Tradeoffs for Approximate Nearest Neighbor Searching

SUNIL ARYA

Hong Kong University of Science and Technology, Kowloon, Hong Kong, China

THEOCHARIS MALAMATOS

University of Peloponnese, Tripoli, Greece

AND

DAVID M. MOUNT

University of Maryland, College Park, Maryland

Abstract. Nearest neighbor searching is the problem of preprocessing a set of n point points in d-dimensional space so that, given any query point q, it is possible to report the closest point to q rapidly. In approximate nearest neighbor searching, a parameter $\varepsilon > 0$ is given, and a multiplicative error of $(1 + \varepsilon)$ is allowed. We assume that the dimension d is a constant and treat n and ε as asymptotic quantities. Numerous solutions have been proposed, ranging from low-space solutions having space $O(n)$ and query time $O(\log n + 1/\varepsilon^{d-1})$ to high-space solutions having space roughly $O((n \log n)/\varepsilon^d)$ and query time $O(\log(n/\varepsilon))$.

We show that there is a single approach to this fundamental problem, which both improves upon existing results and spans the spectrum of space-time tradeoffs. Given a tradeoff parameter γ, where $2 \leq \gamma \leq 1/\varepsilon$, we show that there exists a data structure of space $O(n\gamma^{d-1} \log(1/\varepsilon))$ that can answer queries in time $O(\log(n\gamma) + 1/(\varepsilon\gamma)^{(d-1)/2})$. When $\gamma = 2$, this yields a data structure of space $O(n \log(1/\varepsilon))$ that can answer queries in time $O(\log n + 1/\varepsilon^{(d-1)/2})$. When

$\gamma = 1/\varepsilon$, it provides a data structure of space $O((n/\varepsilon^{d-1})\log(1/\varepsilon))$ that can answer queries in time $O(\log(n/\varepsilon))$.

Our results are based on a data structure called a (t, ε)-AVD, which is a hierarchical quadtree-based subdivision of space into cells. Each cell stores up to t representative points of the set, such that for any query point q in the cell at least one of these points is an approximate nearest neighbor of q. We provide new algorithms for constructing AVDs and tools for analyzing their total space requirements. We also establish lower bounds on the space complexity of AVDs, and show that, up to a factor of $O(\log(1/\varepsilon))$, our space bounds are asymptotically tight in the two extremes, $\gamma = 2$ and $\gamma = 1/\varepsilon$.

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

General Terms: Algorithms, Theory

Additional Key Words and Phrases: Nearest neighbor searching, space-time tradeoffs

1. *Introduction*

Nearest neighbor searching is a fundamental computational problem, having wide-ranging applications in areas such as knowledge discovery, pattern recognition, machine learning, data compression, and information retrieval. A set S of n points is given in some metric space X, and the task is to preprocess these points so that, given any query point $q \in X$, the point of S nearest to q can be reported quickly. Efficient exact solutions are known for only very limited cases, and this has led to interest in approximation algorithms. Given a point q, let $\mathrm{NN}_q(S)$ be the distance from q to its nearest neighbor in S. For a real parameter $\varepsilon > 0$, we say that a point $p \in S$ is an *ε-nearest neighbor* (*ε-NN*) of q if the distance from p to q is at most $(1 + \varepsilon)\mathrm{NN}_q(S)$.

There are a number of common formulations under which this problem has been considered. In traditional computational geometry, the space is $\mathbb{R}^d$, real d-dimensional space, for some constant d under the Euclidean norm. It is common to treat ε as an asymptotic quantity of secondary importance to n. Other formulations treat the dimension d as an asymptotic quantity [Indyk and Motwani 1998; Kushilevitz et al. 2000; Panigrahy 2006] and seek solutions having no exponential dependence on d. Yet others assume that the points are drawn from a general metric space. The metric space is usually assumed to possess some growth limiting property, such as having constant doubling dimension [Clarkson 1999; Karger and Ruhl 2002; Krauthgamer and Lee 2004, 2005; Cole and Gottlieb 2006; Har-Peled and Mendel 2006; Arya et al. 2008b].

In this article, we focus on the first formulation. While more restrictive than the others, there are nonetheless many applications of nearest neighbor searching in Euclidean spaces of relatively low dimensions (e.g., ranging from 2 to 10) in areas as diverse as computer vision, robotics, solid modeling, and computational physics. In such cases, the geometric structure of the underlying space can be exploited to achieve the best asymptotic performance (subject to the assumption of fixed dimension). The principal complexity issues involve determining the best relationships between query time and the space of the data structure. Since n is the primary asymptotic quantity, the goal is to achieve dependencies on n in the

space and query time that are linear and logarithmic, respectively. Subject to this, the objective is to minimize factors depending on ε. In multidimensional spaces, distances tend to concentrate about their mean value [Beyer et al. 1999; François et al. 2007], and hence accuracy is important, and consequently ε-dependencies are often the dominant terms in query times [Arya et al. 1998].

We build upon many years of research on the design of data structures for ε-NN searching in low-dimensional Euclidean space. Arya et al. [1998] proposed a balanced quadtree-like partition tree, the BBD tree, which achieves $O((1/\varepsilon)^d \log n)$ query time with $O(n)$ space. Various improvements were proposed by others, including Bespamyatnikh [1996], Duncan et al. [2001], and Chan [2002, 2006], the best of which offers a query time of $O(\log n + 1/\varepsilon^{d-1})$ with $O(n)$ space. While these structures are optimal with respect to space, the ε-dependencies in query time are far from optimal. Classical results on approximating convex bodies by polytopes suggest that it should be possible to reduce ε-dependencies to $O(1/\varepsilon^{(d-1)/2})$ [Dudley 1974; Bronshteyn and Ivanov 1976]. Clarkson [1994] first demonstrated the relationship, and subsequently Chan [1998] strengthened Clarkson's results. Har-Peled [2001] proposed a still faster method based on generalizing the *locus method*, in which a geometric query problem is reduced to point location in an appropriate subdivision of space. For example, it is well known that exact nearest neighbor queries can be reduced to point location in the Voronoi diagram of the point set S [de Berg et al. 2000]. Har-Peled's approach involves creating a hierarchical subdivision of space by a balanced quadtree-like structure. Each cell of this subdivision stores a *representative point* of S, which is an ε-nearest neighbor of any query point in the cell. This suggests the name *approximate Voronoi diagram*, or *AVD*. Using an AVD to answer ε-NN queries reduces to point-location in this subdivision, which is solved by a simple descent of the tree. Har-Peled showed that queries can be answered in time $O(\log(n/\varepsilon))$ with space $O((n/\varepsilon^d)(\log n) \log(n/\varepsilon))$. The space bound was improved by Sabharwal et al. [2006] to $O((n/\varepsilon^{d+1}) \log(1/\varepsilon))$. The AVD approach provides much lower query times than the above methods, but with significantly higher ε-dependencies in space.

An interesting commonality among all the above results is that, ignoring log factors, the product of space and query time is roughly the same in each case. That is, if we let $M_\varepsilon(n)$ and $T_\varepsilon(n)$ denote the space and query time, respectively, of these methods, the relationship $M_\varepsilon(n)T_\varepsilon(n) = O(n/\varepsilon^{d-1})$ nearly holds in each case, in spite of the fact that significantly different methods have been used. This suggests that this is the "correct" complexity relationship to expect. This raises the question of whether there is a *single approach* to ε-NN searching that admits a space-time tradeoff throughout the performance spectrum and while achieving this space-time relationship.

In this article, we achieve this objective and more. First, we present a unified approach to ε-NN searching that spans the spectrum of space-time tradeoffs. Surprisingly, we show that the apparently "correct" relationship between space and query time suggested by all the above results is not just wrong, but very wrong. In particular, our approach achieves a relationship that (ignoring log factors) satisfies $M_\varepsilon(n)T_\varepsilon^2(n) = O(n/\varepsilon^{d-1})$. Put differently, for any $c \geq 1$, if we reduce space by a factor of $1/c$, the increase in query time that we suffer is not $O(c)$, but only $O(\sqrt{c})$. Our approach is based on a generalization of Har-Peled's AVD structure. We also present lower bounds on the time and space complexity of ε-NN searching based

on AVDs, and we demonstrate that our constructions are (up to logarithmic factors) asymptotically optimal in the extreme cases of the tradeoff spectrum.

2. *Results*

Recall that we are given an n-element point set $S \in \mathbb{R}^d$. Given a positive real $\varepsilon \leq 1/2$ and an integer $t \geq 1$, we can generalize Har-Peled's structure [Har-Peled 2001] by defining a (t, ε)-*approximate Voronoi diagram* (or (t, ε)-AVD) of S to be a subdivision of space into cells, where each cell w is associated with a subset of at most t *representatives* of S, such that for any point $q \in w$, at least one of these representatives is an ε-nearest neighbor of q with respect to S. As in Har-Peled [2001], the cells of our AVD are the leaves of a balanced quadtree-like data structure (see Section 3.2 for definitions), and so queries can be answered in time $O(\log n' + t)$, where n' is the number of cells of the AVD. The total space of an AVD is proportional to the sum over all the cells of the number of representatives per cell. As we will show, the efficiencies arise because allowing for a small number of additional representatives per cell results in a more than proportional decrease in the total space.

First, we present a basic AVD construction. In addition to S and ε, the construction algorithm is given a tradeoff parameter γ, where $2 \leq \gamma \leq 1/\varepsilon$. We show that there exists a (t, ε)-AVD, where $t = O(1/(\varepsilon\gamma)^{(d-1)/2})$, that can answer ε-NN queries in time $O(\log(n\gamma) + t)$. Letting m denote the quantity $n\gamma^d \log(1/\varepsilon)$, the total space of the AVD is $O(m)$, and (ignoring log factors) it can be constructed in time $O(m/(\varepsilon\gamma)^{(d-1)/2})$ (see Theorem 8.4). At one extreme, when $\gamma = 2$, this yields a data structure of space $O(n \log(1/\varepsilon))$ that can answer queries in time $O(\log n + 1/\varepsilon^{(d-1)/2})$, thus achieving query times as good as those of Clarkson [1994] and Chan [1998] but with nearly linear space. At the other extreme, $\gamma = 1/\varepsilon$, this provides a data structure of space $O((n/\varepsilon^d) \log(1/\varepsilon))$ that can answer queries in time $O(\log(n/\varepsilon))$.

If we ignore logarithmic factors, we have

$$M_\varepsilon(n) = O(n\gamma^d) \qquad \text{and} \qquad T_\varepsilon(n) = O(1/(\varepsilon\gamma)^{(d-1)/2}),$$

and hence, $M_\varepsilon(n)T_\varepsilon^2(n) = O(n\gamma/\varepsilon^{d-1})$. This is close to, but not exactly equal to, the relationship we desire. We present an enhanced AVD construction, called the *bisector-sensitive construction*, which achieves the same query time while improving the space complexity to $O(n\gamma^{d-1} \log(1/\varepsilon))$. In the extreme case of $\gamma = 1/\varepsilon$, the space is reduced to $O((n/\varepsilon^{d-1}) \log(1/\varepsilon))$. (See Theorem 9.9 for details.) This result is superior to the AVD-based solutions of both Har-Peled [2001] and Sabharwal et al. [2006]. Ignoring log factors, the resulting structure satisfies the desired relationship $M_\varepsilon(n)T_\varepsilon^2(n) = O(n/\varepsilon^{d-1})$.

Finally, we establish lower bounds on the space complexity of AVDs. Our lower bound model is both simpler and more general than that of our upper bounds. We assume that cells are fat axis-aligned rectangles (not necessarily based on quadtree boxes), which are allowed to overlap one another (not necessarily disjoint). Also, our lower bound model does not make any assumptions about the existence of a search structure to locate the cell containing the query point. The exact bound is given in Theorem 10.5, but the principal observation is that, up to a factor of $O(\log(1/\varepsilon))$, our space bounds are asymptotically tight in the two extremes, $\gamma = 2$ and $\gamma = 1/\varepsilon$.

Our approach is based on a number of enhancements to the AVD structure as proposed by Har-Peled. As mentioned above, the only fundamental change to the structure is allowing multiple representatives per cell. Query processing is still very simple: descend the tree to locate the query point and return the closest representative. This apparently minor extension leads to a much richer and more complex array of options for the construction and analysis of the AVD: how to select the representatives, how best to decompose space to minimize the total space of the structure, and how to analyze the resulting space. In order to answer these questions, we develop a characterization of the AVD in terms of certain *separation properties* between the cells and the points of S. Intuitively, as the separation increases, more cells are needed but with fewer representatives per cell.

Our preliminary papers [Arya and Malamatos 2002; Arya et al. 2002] explored a number of different approaches to building and analyzing AVDs (including a construction based on deterministic sampling, a cell construction involving quadratic growth rates, and a more complex analysis based on two separation properties). In this article, we have made a number of enhancements and simplifications. We have pared down the relatively complex combination of tools to just two key constructs. The first is a space analysis technique, called *spatial amortization*, which we use to prove that the basic construction leads to nearly optimal space complexity. The second is the aforementioned *bisector-sensitive construction*, which is used to achieve our best space bounds. We have also enhanced the lower bound techniques presented in these earlier papers so that, rather than bounding the number of cells, we bound the more meaningful quantity of total space. This enhancement has necessitated the development of more powerful tools for analyzing packing properties involving fat axis-aligned rectangles with respect to nonorthogonal linear subspaces of various dimensions.

The rest of the article is organized as follows. In Section 3, we begin with a review of two background concepts, the well-separated pair decomposition and BBD trees, which will be used throughout the article. Next, in Section 4 we present a high level overview of the AVD, its construction, and correctness. Section 5 provides the necessary geometric preliminaries to understand separation properties and their role in approximate nearest neighbor searching. Next, Section 6 presents the basic AVD construction. This is followed by Sections 7 and 8, which respectively present the method of spatial amortization and the space analysis of the basic AVD construction. The resulting AVD is slightly suboptimal with respect to space, and in Section 9, we present the bisector-sensitive construction, which produces our best space bounds. Finally, in Section 10, we present our lower-bound results and contrast them with the upper bounds of the preceding sections.

3. *Preliminaries*

As mentioned in the introduction, we assume that the dimension d is a fixed constant. We treat n, ε, and γ as asymptotic quantities. To make this distinction clearer, we use the term *constant* throughout to refer to any fixed quantity (which may depend, even exponentially, on dimension), and we use *parameter* to refer to real-valued quantities that may depend on ε and γ. To avoid specifying the many real-valued constants that arise in our constructions and analyses, we will often hide them using

asymptotic notation. For positive real α, we use the notation $O(\alpha)$ (respectively, $\Omega(\alpha)$) to mean a quantity whose value is at most (respectively, at least) $c\alpha$ for an appropriately chosen constant c. Similarly, $\Theta(\alpha)$ refers to a quantity whose value lies in the interval $[c_1\alpha, c_2\alpha]$ for some appropriately chosen constants c_1 and c_2. To avoid special annotation when α is very small, we take $\log \alpha$ to mean $\max(1, \log \alpha)$ when it is used in asymptotic expressions.

We assume that the set S of points has been scaled and translated to lie within a ball of radius $\varepsilon/11$ placed at the center of the unit hypercube $[0, 1]^d$. This allows us to ignore query points lying outside the unit hypercube, since (as we shall show later in Lemma 5.3) any point of S is an ε-approximate nearest neighbor to such a query point.

Let x and y denote any two points in $\mathbb{R}^d$. We use $\|xy\|$ to denote the Euclidean distance between x and y and $\overline{xy}$ to denote the segment joining x and y and $\overrightarrow{xy}$ to denote the vector $y - x$. Given sets $u, w \subset \mathbb{R}^d$ we let $dist(u, w)$ denote the minimum (or more accurately the infimum) distance between any two points of the sets. If either set is empty, $dist(u, w) = \infty$. We use $dist(x, w)$ to mean $dist(\{x\}, w)$. Note that $\mathrm{NN}_x(w)$ and $dist(x, w)$ have the same formal meanings, but the former is usually used when w is a finite set of points. Given a finite point set S and region w, we define an ε-*representative set* for w with respect to S to be a subset $R \subseteq S$ such that for all $q \in w$, $\mathrm{NN}_q(R) \leq (1 + \varepsilon)\mathrm{NN}_q(S)$.

We denote by $b(x, r)$ a closed ball of radius r centered at x, that is, $b(x, r) = \{y : \|xy\| \leq r\}$. For a ball b and any positive real γ, we use γb to denote the ball with the same center as b and whose radius is γ times the radius of b, and $\overline{b}$ to denote the set of points that are not in b.

Next, we briefly review two concepts that will play an important role in our constructions: well-separated pair decompositions and balanced box-decomposition trees.

3.1. WELL-SEPARATED PAIR DECOMPOSITIONS.   Let S be a set of n points in $\mathbb{R}^d$. We say that two sets of points X and Y are *well separated* if they can be enclosed within two disjoint d-dimensional balls of radius r, such that the distance between the centers of these balls is at least σr, where $\sigma > 2$ is a real parameter called the *separation factor*. If we consider joining the centers of these two balls by a line segment, the resulting geometric shape resembles a *dumbbell*. (See Figure 1.) The balls are the *heads* of the dumbbell. Define the *length* of a dumbbell to be the distance between the centers of the balls, and the *center* of the dumbbell to be the midpoint of these two centers.

A *well-separated pair decomposition* (*WSPD*) of S is a set

$$\mathcal{P}_{S,\sigma} = \{(X_1, Y_1), \dots, (X_m, Y_m)\}$$

of pairs of subsets of S such that (i) for $1 \leq i \leq m$, X_i and Y_i are well separated, and (ii) for any distinct points $x, y \in S$, there exists a unique pair (X_i, Y_i) such that either $x \in X_i$ and $y \in Y_i$ or vice-versa. We say that the pair (X_i, Y_i) *separates* x and y. Callahan and Kosaraju [1995] have shown that we can construct a WSPD containing $O(\sigma^d n)$ pairs in $O(n \log n + \sigma^d n)$ time. For each pair, their construction also provides the corresponding dumbbell. (An example of a well-separated pair decomposition and the associated dumbbells is shown in Figure 1.) In many instances, we will be interested in just the dumbbells themselves, not the points they enclose. For this reason, we will sometimes consider the WSPD to be a set
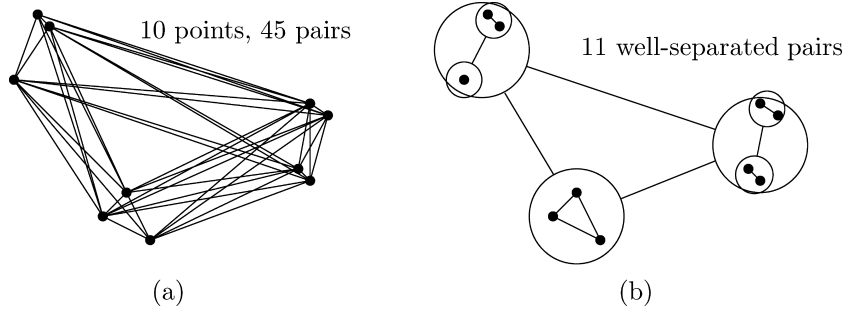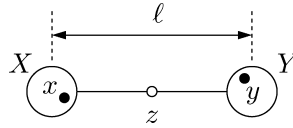
FIG. 1. A well-separated pair decomposition.

FIG. 2. A dumbbell and Lemma 3.1.

of $O(\sigma^d n)$ dumbbells. To avoid adding more notation, we will use the letter P to refer both to a well-separated pair and its associated dumbbell. It will be clear from context which is intended.

In all of our constructions the separation factor σ will be a constant (independent of ε), and we will assume that $\sigma > 4$. Such a separation factor has the useful property that, given any dumbbell, the distance between any two points in different heads is strictly greater than the distance between two points in the same head. The following additional observations are easy consequences of the triangle inequality (see Figure 2).

LEMMA 3.1. *Consider the dumbbell for well-separated pair $P = (X, Y)$ with separation factor $\sigma > 4$. Let ℓ be P's length (the distance between its dumbbell centers), and let z be P's center. Then for any $x \in X$ and $y \in Y$ we have:*

 (i) $\ell/2 < \|xy\| < 3\ell/2$.
(ii) $\ell/4 < \|xz\| < 3\ell/4$ *(and same bounds hold for $\|yz\|$).*

3.2. BOX DECOMPOSITION AND BBD TREES. Let $[0, 1]^d$ denote a unit hypercube in $\mathbb{R}^d$. A *quadtree box* [Samet 1990] is defined recursively to be either the unit hypercube or a hypercube obtained by splitting any quadtree box into 2^d equal parts. We define the *size* of a quadtree box to be its side length. It is easy to see that any two distinct quadtree boxes are either disjoint or one is contained inside the other.

There are a number of data structures derived from a recursive quadtree-based decomposition of space. One problem that arises with quadtrees in their simplest form is the potential for arbitrarily long trivial sequences of splits. This is typically handled by path compression, as is used in the *compressed quadtree* [Har-Peled 2008]. Here we will use a variant of this structure called a *box-decomposition tree* (or *BD tree*) [Arya et al. 1998]. This is a 2^d-ary tree in which each node is associated with a region of space called a *cell*, which is the difference of two quadtree boxes, an *outer box* and an (optional) *inner box*. The root of the tree is associated with a

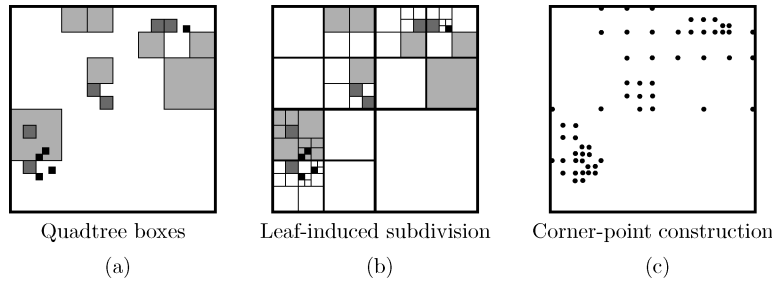| Quadtree boxes	Leaf-induced subdivision	Corner-point construction
(a)	(b)	(c)

FIG. 3. A set of quadtree boxes and the decomposition induced by the structure's leaves. (The outer and inner boxes and tree structure are not illustrated.)

unit hypercube. The cell associated with any node is partitioned into disjoint cells, which are associated with the children of the node. We define the *size* of a cell to be the same as the size of its outer box.

The BD tree need not be balanced, meaning that its depth may be asymptotically as large as the number of nodes. A centroid decomposition [Bent et al. 1985; Frederickson 1997] can be applied to the BD tree in order to produce a balanced structure whose height is logarithmic in the number of leaves. (See, e.g., Arya et al. [1998] and Har-Peled [2008].) One implementation of this idea is based on an alternation of splitting with a centroid-like shrinking operation to generate a balanced tree, called the *balanced box-decomposition tree* (or *BBD tree*) [Arya et al. 1998]. However implemented, such a structure satisfies the following properties, which we refer henceforth as *BBD properties*. (Justifications are given below.)

 (i) (Construction from Point Sets). Given a set S of n points in the d-dimensional unit hypercube, it is possible in $O(n \log n)$ time to construct such a structure having space $O(n)$ such that each leaf node contains at most one point of S. In addition, we may assume that each internal node of the tree stores a single point of S that lies within the corresponding cell (assuming such a point exists).

 (ii) (Construction from Quadtree Boxes). Given a collection $\mathcal{U}$ of n quadtree boxes in the d-dimensional unit hypercube (see Figure 3(a)), it is possible in $O(n \log n)$ time to construct such a structure with $O(n)$ nodes such that the subdivision induced by its leaves is a refinement of the subdivision induced by the quadtree boxes in $\mathcal{U}$ (see Figure 3(b)).

(iii) (Packing Properties). Consider any BBD tree in d-dimensional space. Any subset of its cells that intersect a ball of radius r, such that these cells have pairwise disjoint interiors and are each of size at least s, has cardinality at most $O((1 + r/s)^d)$.

(iv) (Point Location). Given either of the structures of (i) or (ii), it is possible to determine the leaf cell containing an arbitrary query point $q \in \mathbb{R}^d$ in $O(\log n)$ time.

 (v) (Range Sketching). Given an n-element point set S, a quadtree box is said to be *nonempty* if it contains at least one point of S. The construction provided in part (i) has the property that, given a ball b and size parameter s, it is possible to compute the nonempty quadtree boxes of size s that overlap b in time $O(\log n + t)$, where t is the number of nonempty boxes overlapping the
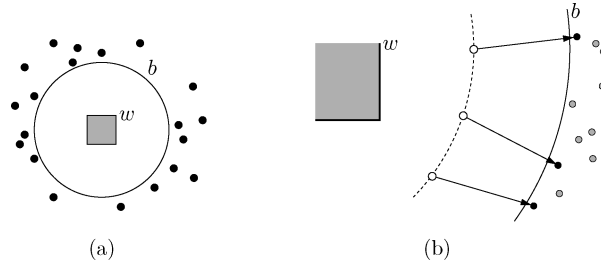
FIG. 4. (a) Separation properties and (b) helpers (white points) and representatives (black).

ball $2b$. At the same time, it is possible to return an arbitrary point of S from within each such box.

(vi) (Approximate Nearest Neighbor Queries). Given the structure of property (i) and given a real parameter $\varepsilon > 0$, it is possible to answer ε-nearest neighbor queries for the set S in time $O((1/\varepsilon)^d \log n)$.

Properties (i), (iii), (iv), and (vi) are proved by Arya et al. [1998]. We can establish (ii) by a simple adaptation of the BBD-tree construction presented there. Let $S_{\mathcal{U}}$ denote the set of at most $2^d n$ points consisting of the vertices of the boxes of $\mathcal{U}$ (see Figure 3(c)). Apply the recursive BBD-tree construction of Arya et al. [1998] to $S_{\mathcal{U}}$ with the modification that the decomposition ceases when there are no points in the *interior* of the current cell. (Thus, the points of $S_{\mathcal{U}}$ reside only along the boundaries of the leaf cells.) Property (v) follows by a modification of the approximate range searching algorithm presented by Arya and Mount [2000] and is proved formally by da Fonseca [2007] and Arya et al. [2008a].

Throughout, we use the following notation. Given a cell w, let s_w denote the size of w, let $r_w = s_w d$, and let b_w be the ball of radius r_w whose center coincides with the center of w's outer box. (Observe that $w \subseteq b_w$.)

4. Overview

Before presenting the details of our AVD structure, we present a high-level overview of the structure and its derivation. Let S be a set of n points in $\mathbb{R}^d$, and let $0 < \varepsilon \leq 1$ be the approximation parameter. Constructing an AVD reduces to the problem of computing a sufficiently refined subdivision of space such that, for any cell of the subdivision, there exists a suitably small ε-representative set. To achieve this, the cells should satisfy certain *separation properties* with respect to S. Consider the subdivision of space induced by the leaves of a BBD-tree decomposition. Let w be an arbitrary cell of the subdivision. Let b be a ball, called the *outer ball*, centered about w that is larger than w's diameter by some appropriate *expansion factor*. We shall show that space-time tradeoffs can be achieved by adjusting this factor, but for now it is convenient to assume it is a constant (see Figure 4(a)). Suppose that all the points of S lie outside of b. We will show that, irrespective of the distribution or number of points of S around b, it is possible to identify a subset of representatives of size $O(1/\varepsilon^{(d-1)/2})$. (This quantity will generally depend on the expansion factor.)

We can compute these representatives as follows. First, we place $O(1/\varepsilon^{(d-1)/2})$ points uniformly on a concentric sphere located roughly midway between the boundaries of w and b, which we call the *helpers* of w. We compute the nearest
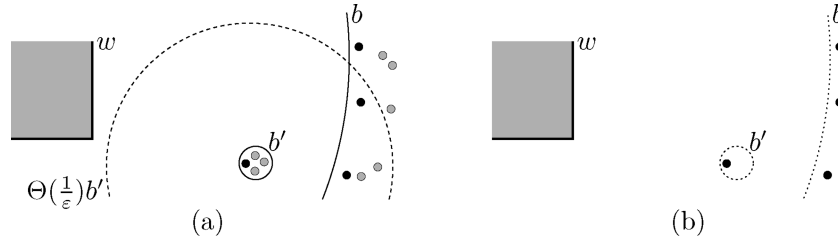
FIG. 5.   (a) Inner-ball separation properties and (b) final representatives.

neighbor of each helper or, in general, a sufficiently close approximate nearest neighbor (see Figure 4(b)). (This idea was introduced earlier by Chan and Snoeyink [1995] in a context unrelated to AVDs.) These neighbors constitute the cell's representatives.

Of course, we cannot produce a subdivision in which *all* the cells have no points within their outer ball. (The points have to be somewhere!) But we can make due with more general separation properties. Clearly we can tolerate a single point of S lying within w, since we can simply add it as a representative. Also, we will allow b to contain an arbitrary number of points of S provided that they can be enclosed within a ball b', called the *inner ball*, such that an expansion of b' does not intersect w (see Figure 5(a)). For our purposes, it will suffice to take the expansion factor for the inner ball to be $\Theta(1/\varepsilon)$. (Note that this is distinct from the outer-ball expansion factor mentioned above.) Because of the high degree of separation between b' and w, it suffices to take just one representative from this cluster in order to determine the approximate nearest neighbor of any query point lying within w. The final set of representatives consists of the single representative from the inner ball and the set of representatives from outside b (see Figure 5(b)).

The question remains of how to generate the cells of the BBD-tree subdivision to achieve these separation properties. Our approach is to construct a well-separated pair decomposition (WSPD) of S with a sufficiently large constant separation factor (which does not depend on ε). The number of pairs in such a decomposition is $O(n)$. Consider any well-separated pair, and let ℓ denote the distance between the centers of the two heads of the associated dumbbell (see Figure 6). We will use this pair to fragment the surrounding region of space into sufficiently small quadtree boxes. This is done in concentric layers. We consider a sequence of balls centered about the associated dumbbell of exponentially increasing radii $\langle \ell, 2\ell, 4\ell, \ldots, \Theta(\frac{1}{\varepsilon})\ell \rangle$ and cover each ball of radius r with a collection of quadtree boxes of side length $\Theta(r)$. Note that boxes generated from different pairs may overlap each other. By a simple packing argument, the number of boxes generated for each well-separated pair is $O(\log(1/\varepsilon))$. We repeat this for all the $O(n)$ pairs of the WSPD.

We compute a BBD tree whose leaf-level subdivision is a refinement of these boxes. By BBD property (ii) this tree has size $O(n\log(1/\varepsilon))$. By the methods mentioned earlier, we compute and store the $O(1/\varepsilon^{(d-1)/2})$ representatives for the leaves of the associated tree. The resulting structure is the desired AVD. In Section 6, we show that the resulting AVD satisfies the aforementioned separation properties. Ignoring the small $O(\log(1/\varepsilon))$ factor for now, the resulting structure has $O(n)$ nodes and total size $O(n/\varepsilon^{(d-1)/2})$. In order to answer ε-approximate nearest neighbor queries, we locate the leaf cell containing the query point in time $O(\log n)$ and
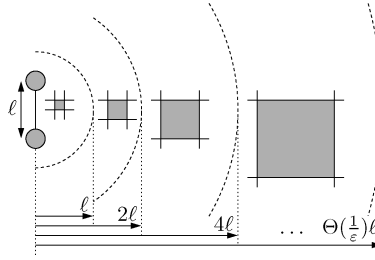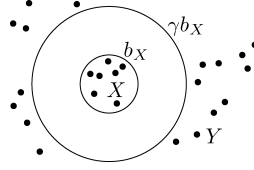
Fig. 6.   Construction of the AVD.

then compute its closest representative in time $O(1/\varepsilon^{(d-1)/2})$, which yields a total query time of $O(\log n + 1/\varepsilon^{(d-1)/2})$.

The $O(n/\varepsilon^{(d-1)/2})$ space bound that this simple analysis provides is quite weak. It can be shown that (if the representatives are chosen with some care) the total space is only $O(n \log(1/\varepsilon))$. In Section 7, we prove this through a charging technique, which we call *spatial amortization*. Here is a short intuitive description. Suppose that a cell requires a large number of representatives. This will be evident by the fact that there are many points at roughly the same distance from the cell, and further these points will be spatially well distributed about the cell. Such a collection of points will give rise to a proportionately large number of well-separated pairs of significant length in the vicinity of the cell. In spatial amortization, each well-separated pair applies a charge to certain cells lying in its vicinity. We show that this can be done so that every cell receives at least as many charges as it has representatives, and furthermore, this can be achieved with only $O(\log(1/\varepsilon))$ charges per well-separated pair. Since there are $O(n)$ pairs in the WSPD, it follows that the total number of charges, and hence the total number of representatives, is $O(n \log(1/\varepsilon))$, which is much smaller than the naive worst-case bound. (See Section 7 for further details.)

For the sake of simplicity, this overview has ignored the issue of space-time tradeoffs. One nice feature of our AVD construction is that it lends itself easily to providing tradeoffs between space and query time. Recall the outer-ball expansion factor introduced earlier. We have assumed it to be a constant thus far, but it can be meaningfully increased up to $O(1/\varepsilon)$. Increasing its value results in more cells, but with fewer representatives per cell. In the extreme case, we need only one representative per cell, but the number of cells grows to $O((n/\varepsilon^d) \log(1/\varepsilon))$. (See Theorem 8.4 for a complete specification of the space-time tradeoffs.) As the outer-ball expansion factor increases, this bound on the number of cells is suboptimal in that it carries an additional factor of $1/\varepsilon$ in the extreme. Our lower bounds of Section 10 indicate that the number of cells in the limiting case should be $O(n/\varepsilon^{d-1})$, rather than the roughly $O(n/\varepsilon^d)$ bound given above. What is the source of this inefficiency? Our construction decomposes the space around each well-separated pair (X, Y) in a uniform manner. Intuitively, the quadtree boxes of the decomposition arise from a need to ascertain whether the approximate nearest neighbor lies in X or in Y. This suggests that we only need to generate those boxes whose nearest neighbor in X is roughly at the same distance as its nearest neighbor in Y, that is, those points lying near the Voronoi bisector of X and Y. Since X and Y are well separated, the bisector is a relatively smooth $(d-1)$-dimensional polyhedral manifold, and hence the number of boxes needed to cover it grows as $1/\varepsilon^{d-1}$ rather than $1/\varepsilon^d$. This suggests a more efficient construction algorithm,

FIG. 7.   Concentric γ-separation.

which is sensitive to the location of this bisector. This construction is presented in Section 9.

In Sections 5 through 9, we present the AVD constructions and analyses, which we have just outlined. Finally, in Section 10, we present lower bounds on the complexity of AVDs.

5. *Separation and Approximate Nearest Neighbors*

An important and recurring theme in the study of AVDs is the notion that in any subdivision of space, say one generated by the leaf cells of a BBD tree, the greater the degree that a cell is separated from the points in its environment the more concisely and/or the more accurately its approximate nearest neighbor information can be encoded. In this section, we quantify these relationships formally. First, we define the principal notion of separation, which will be applied throughout our article in order to bound the number of nearest neighbor representatives. Consider two sets $X, Y \subset \mathbb{R}^d$, and two positive real parameters $\beta, \gamma \geq 1$. Recall that given a ball b, γb is the ball concentric with b whose radius is larger by the factor γ. We say that X is *concentrically γ-separated* from Y if X can be enclosed within a d-dimensional ball b_X such that $\gamma b_X \cap Y = \emptyset$ (see Figure 7). Note that X and Y need not be discrete sets, and may themselves be geometric regions, such as balls or quadtree boxes.

Recall that an *ε-representative set* for a region u with respect to a set of points S is a subset $R \subseteq S$ such that for all $q \in u$, $\mathrm{NN}_q(R) \leq (1+\varepsilon)\mathrm{NN}_q(S)$. Our first result bounds the number of nearest neighbor representatives needed when the query and data points are concentrically separated. The result is symmetrical in the sense that query points and data points may be placed either in the inner ball or outside the outer ball. The proofs of the lemmas of this section are rather technical and are presented later in Appendix A.

LEMMA 5.1 (CONCENTRIC BALL LEMMA). *Let $c > 0$ be a constant, and let $0 < \varepsilon \leq 1$ and $\gamma \geq 1 + c$ be two real parameters. Let S be a set of points in $\mathbb{R}^d$. Let b_1 and b_2 be two balls in $\mathbb{R}^d$ such that b_1 is concentrically γ-separated from $\overline{b}_2$. Then there exist subsets $R_1, R_2 \subseteq S$ each consisting of at most*

$$O\left(\left(1 + \frac{1}{\sqrt{\varepsilon\gamma}}\right)^{d-1}\right)$$

points such that

 (i) *R_1 is an ε-representative set for b_1 with respect to $S \cap \overline{b}_2$, and*
(ii) *R_2 is an ε-representative set for $\overline{b}_2$ with respect to $S \cap b_1$.*

Note that the number of representatives needed to answer any query is a function of only ε, γ, and the dimension, and is independent of the size of S. If γ is $\Omega(1/\varepsilon)$, the above lemma implies that $O(1)$ representatives suffice. The following result shows that if γ is sufficiently large then a single representative suffices.

LEMMA 5.2 (SINGLE-REPRESENTATIVE CONC. BALL LEMMA). *Let $0 < \varepsilon \le 1$ and $\gamma \ge 11/\varepsilon$ be two real parameters. Let S be a set of points in $\mathbb{R}^d$. Let b_1 and b_2 be two balls in $\mathbb{R}^d$ such that b_1 is concentrically γ-separated from $\overline{b}_2$.*

(i) *Let p_x be an $(\varepsilon/2)$-nearest neighbor of any point $x \in b_1$. Then, $\{p_x\}$ is an ε-representative set for b_1 with respect to $S \cap \overline{b}_2$.*

(ii) *Let p be any point of $S \cap b_1$. Then, $\{p\}$ is an ε-representative set for $\overline{b}_2$ with respect to $S \cap b_1$.*

This lemma allows us to dispense with the easy case of a query point that is very far from the point set S. If S lies within a ball b_1 of diameter $\varepsilon/11$ at the center of unit hypercube, then any query point q that lies outside the unit cube lies outside a ball b_2 of unit diameter inscribed within the hypercube. Clearly, b_1 is concentrically $(11/\varepsilon)$-separated from $\overline{b}_2$. By Lemma 5.2(ii), any point of S can be used as an ε-approximate nearest neighbor of q. Thus, we have the following.

LEMMA 5.3. *Let $0 < \varepsilon \le 1$ be a real parameter, let b be a ball of diameter $\varepsilon/11$ located at the center of the unit hypercube $[0, 1]^d$. Given a finite point set $S \subset b$, any single point of S is an ε-representative set for the complement of the unit hypercube with respect to S.*

6. *Basic AVD Construction and Separation Properties*

In this section we describe our basic method for constructing an approximate Voronoi diagram (AVD). Later, in Section 9, we will modify this basic construction to produce the best space and time tradeoffs. Let S be an n-element point set in $\mathbb{R}^d$ that has been scaled to lie within a ball of diameter $\varepsilon/11$ at the center of the unit hypercube. By Lemma 5.3, queries outside the unit hypercube are trivial to answer, and so henceforth it suffices to consider only query points lying within the unit hypercube. Recall from Section 4 that a key idea is to generate a subdivision whose cells satisfy certain separation properties with respect to the points of S. These separation properties will enable us to store a sparse set of nearest neighbor representatives to handle the query points lying within each cell. In this section, we define these separation properties and show that it is possible to construct a BBD tree whose leaf cells satisfy these properties. Because it is often desirable to provide additional control of the degree of separation, we provide two parameters, β and γ. These control the degree of separation with the inner and outer balls, respectively. (In all the constructions of this article $\beta = O(1/\varepsilon)$, but we present our constructions in general because AVDs are useful in other applications of geometric retrieval.)

Recall that, given a leaf cell w of a BBD tree, s_w denotes its size (its side length), and b_w denotes an enclosing ball of radius $r_w = s_w d$ centered about w. Throughout, we let $m(n, d, \gamma, \beta) = n\gamma^d \log \beta$ be the function that provides the asymptotic upper bound on the number of cells of the subdivision. When n, d, γ, and β are clear from context, we will just present this as m.
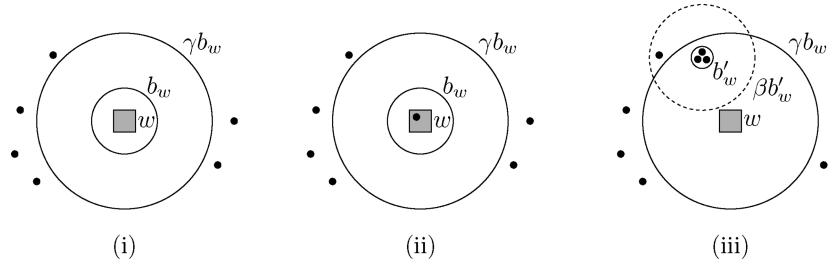
FIG. 8.   Basic AVD separation properties.

LEMMA 6.1 (BASIC SEPARATION PROPERTIES).   *Consider real parameters $\beta \geq$
2 and $\gamma \geq 2$. Let S be a set of n points in $\mathbb{R}^d$, and recall that $m = m(n, d, \gamma, \beta) = n\gamma^d \log \beta$. It is possible to construct a BBD tree T with $O(m)$ nodes, where each leaf cell w satisfies at least one of the following three properties:*

(i) *$S \cap \gamma b_w = \emptyset$, and hence b_w is concentrically γ-separated from S (see Figure 8(i)).*

(ii) *$|S \cap w| = 1$, and b_w is concentrically γ-separated from $S \setminus w$ (see Figure 8(ii)).*

(iii) *There exists a ball b'_w such that $S \cap \gamma b_w \subseteq b'_w$ and b'_w is concentrically β-separated from w (see Figure 8(iii)).*

In time $O(m \log m)$, we can construct T with the following information stored at the nodes. For each leaf cell w, if it satisfies (ii) we store the point $S \cap w$, and if it satisfies (iii) we store the ball b'_w and a single point of S lying within this ball.

Before giving the proof, we present the construction algorithm for the desired BBD tree T. As mentioned above, our construction depends on two parameters β and γ (which generally depend on ε), and there are two constants c_1 and c_2 (which do not depend on ε). The specific values of the constants will become apparent from the proof of Lemma 6.1. To avoid explicit reference to powers of 2 for box sizes, we will use the convenient shorthand "quadtree boxes of size s" to mean "quadtree boxes of size $2^{\lfloor \log s \rfloor}$." Such boxes have side lengths ranging from $s/2$ to s.

We begin by computing a well-separated pair decomposition $\mathcal{P}$ for S using any constant separation factor $\sigma > 4$. For a fixed well-separated pair $P \in \mathcal{P}$, let ℓ and z denote its length and center, respectively. (Recall that these are the respective length and center of the line segment joining the centers of P's dumbbell heads.) Next, compute a set of quadtree boxes $\mathcal{U}(P)$ as follows. For $0 \leq i \leq \lceil \log(c_1 \beta) \rceil$, let $b_i(P)$ denote the ball centered at z of radius $r_i \leftarrow 2^i \ell$. Let $\mathcal{B}(P)$ denote the resulting set of balls. These balls involve radius values ranging from ℓ to $\Theta(\beta \ell)$. For each such ball $b_i(P)$, let $\mathcal{U}_i(P)$ be the set of quadtree boxes of size $r_i/(c_2 \gamma)$ that overlap the ball (see Figure 6 in Section 4). Let $\mathcal{U}(P)$ denote the union of all these boxes over all the $O(\log \beta)$ values of i.

This process is performed for each well-separated pair of $\mathcal{P}$. Let $\mathcal{U} = \bigcup_{P \in \mathcal{P}} \mathcal{U}(P)$ denote the union of all the boxes. To complete the construction, we apply BBD property (ii) to construct a BBD tree T storing all these boxes.

LEMMA 6.2.   *This BBD-tree construction runs in $O(m \log m)$ time and generates $O(m)$ nodes.*
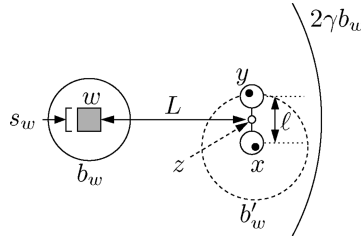
FIG. 9. Proof of AVD separation properties.

PROOF.   For each well-separated pair $P \in \mathcal{P}$ and for each i, we first consider the number of quadtree boxes that overlap the ball $b_i(P)$ of radius r_i in the construction. Since the boxes are of size $r_i/(c_2\gamma)$, by a simple packing argument the number of overlapping quadtree boxes is

$$
O\left(\left(1 + \frac{r_i}{r_i/(c_2\gamma)}\right)^d\right) = O(\gamma^d).
$$

Since the number of balls for each well-separated pair is $O(\log \beta)$, the total number of boxes in $\mathcal{U}(P)$ is $O(\gamma^d \log \beta)$.

The total number of well-separated pairs is $O(n)$, and thus $|\mathcal{U}| = O(n\gamma^d \log \beta)$. Recalling our definition of m above, $|\mathcal{U}| = O(m)$. By BBD property (ii) the number of nodes of T is $O(|\mathcal{U}|) = O(m)$, and it can be constructed in time $O(m \log m)$.   □

In addition to T, we construct a standard BBD tree T_A for S, called the *auxiliary tree*. It will be applied here and later in Section 7 for the purposes of preprocessing. By BBD property (i), T_A has $O(n)$ space and can be constructed in $O(n \log n)$ time. By BBD property (vi), we can use T_A to compute the 1-approximate nearest neighbor of the center of each leaf cell in T. (The distance is within factor 2 of the nearest neighbor.) Each such computation takes $O(\log n)$ time, so the total time for this step is $O(m \log n) = O(m \log m)$. We can now prove Lemma 6.1.

PROOF OF LEMMA 6.1.   We have just shown that the BBD tree T resulting from the above construction satisfies the size and construction time bounds. It suffices to show that for all suitably large constants c_1 and c_2, T possesses the stated separation properties. Let w be any leaf cell of T, and let $x \in S$ be a 1-approximate nearest neighbor of the center of w. If $x \notin 2\gamma b_w$, then since x is a 1-approximate nearest neighbor, it follows that there is no point of S in γb_w, and so (i) holds. Thus, we assume henceforth that $x \in 2\gamma b_w$.

If x is the only point of S in $2\gamma b_w$, then either $x \in w$, implying that (ii) holds, or $x \notin w$, implying that (iii) holds for any sufficiently small ball centered at x. (This ball need not intersect γb_w.) Let us assume therefore that x is not the only point of S in $2\gamma b_w$. Let y be the point of $S \cap 2\gamma b_w$ that is farthest from x. Let $r'_w = \|xy\|$, and let b'_w be the ball of radius r'_w centered at x (see Figure 9). Clearly, any point of $S \cap \gamma b_w$ is contained within b'_w.

It suffices to show that b'_w satisfies separation property (iii). To this end, consider the dumbbell $P \in \mathcal{P}$ that separates points x and y. Let ℓ denote the length of this dumbbell, let z denote its center, and let $L = dist(z, w)$. Throughout, we will make use of the following inequalities, which follow from Lemma 3.1 and the fact that

$\sigma > 4$:

$$\|xz\| \;<\; \ell, \qquad \ell \;<\; 2\|xy\|, \qquad \text{and} \qquad \ell \;>\; \frac{\|xy\|}{2}.$$

We distinguish two cases, based on the relationship between L and ℓ:

Case 1. ($L > c_1\beta\ell$) We will show that separation property (iii) holds. By the triangle inequality

$$dist(x, w) \;\geq\; L - \|xz\| \;>\; c_1\beta\ell - \ell \;>\; (c_1\beta - 1)\frac{\|xy\|}{2} \;=\; \frac{c_1\beta - 1}{2}r'_w.$$

Since $\beta \geq 2$, for all sufficiently large constants c_1 this exceeds $\beta r'_w$, which implies that b'_w is concentrically β-separated from w.

Case 2. ($L \leq c_1\beta\ell$) We will show that this case cannot occur, since otherwise the dumbbell P would have caused w to be split, contradicting the assumption that it is a leaf cell. Since x, y, and w are all contained in the ball $2\gamma b_w$ whose center lies within w, we have both $dist(x, w) \leq 2\gamma r_w$, and $\ell < 2\|xy\| \leq 2(4\gamma r_w) = 8\gamma r_w$. Thus, by the triangle inequality, we have

$$L \;\leq\; \|xz\| + dist(x, w) \;<\; \ell + 2\gamma r_w \;<\; 8\gamma r_w + 2\gamma r_w \;=\; 10\gamma r_w.$$

Because $L \leq c_1\beta\ell$, it follows from our construction that there is a ball of $\mathcal{B}(P)$ that overlaps w. Let b denote the smallest such ball, and let r denote its radius. By the construction we have $r \leq \max(\ell, 2L)$. Since our construction generates all quadtree boxes of size $r/(c_2\gamma)$ that overlap b, it follows that $s_w \leq r/(c_2\gamma)$. Thus, we have

$$r_w \;=\; s_w d \;\leq\; \frac{rd}{c_2\gamma} \;\leq\; \frac{\max(\ell, 2L)d}{c_2\gamma} \;<\; \frac{(20\gamma r_w)d}{c_2\gamma} \;=\; \frac{20 r_w d}{c_2}.$$

Choosing $c_2 \geq 20d$ yields the desired contradiction.

Thus, we have shown that each leaf cell w of T satisfies either separation property (i), (ii), or (iii). If (ii) applies we store the point x with w. If (iii) applies, we have shown that there exists a ball b'_w centered at x that satisfies (iii). For computational purposes (since we do not know the exact locations of the points of $S \cap \gamma b_w$) it suffices to let b'_w be the largest ball centered at x that is concentrically β-separated from w. If this ball lies entirely outside γb_w, we degenerate to satisfying separation property (i). Otherwise, we store the resulting ball and the point x with w. Given w and x this can be done in $O(1)$ time, which completes the proof. □

Now that we have shown that the leaf cells of T satisfy the basic separation properties, it is also useful to observe that each such cell is not too far away from its closest point of S. This fact will be useful in bounding the distance to the nearest neighbor of any point of the cell. Intuitively this follows because for each leaf cell that is created there is a well-separated pair in the vicinity that induced its creation.

LEMMA 6.3. *The BBD tree of Lemma* 6.1 *satisfies the property that there exists a constant $c > 1$ such that for any leaf cell w of this tree the ball $c\gamma b_w$ contains at least one point of S.*

PROOF.   Let v be the quadtree box that has twice the size of w and contains w. Note that some dumbbell $P \in \mathcal{P}$ must generate a quadtree box u that is smaller and contained within v, because otherwise there would be no need to refine v further. Let $b \in \mathcal{B}(P)$ be a ball responsible for generating u (see Figure 10). Let z and r
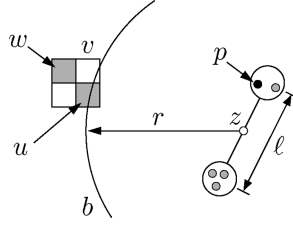
FIG. 10.   Proof of Lemma 6.3.

denote the center and radius of b, respectively. Recall from our construction that b overlaps u and $s_u \geq r/(2c_2\gamma)$. Thus, $dist(z, u) \leq r \leq 2c_2\gamma s_u$.

Let ℓ denote the length of dumbbell P. Since each head of P must contain at least one point of S, it follows from Lemma 3.1 that there is a point $p \in S$ such that $\|pz\| < 3\ell/4 < \ell$. Recalling that the radius of the smallest ball in $\mathcal{B}(P)$ is ℓ, we have $\|pz\| < r$. Thus, by the triangle inequality,

$$dist(p, u) \leq \|pz\| + dist(z, u) < r + r = 2r.$$

Since both w and u are contained in v, the distance between the center of w and any point of u is at most the diameter of v, which is bounded above by $s_v d = 2s_w d = 2r_w$. Thus, the distance between the center of w and p is at most

$$2r_w + dist(p, u) < 2r_w + 2r \leq 2r_w + 4c_2\gamma s_u.$$

Since $s_u \leq s_v/2 = s_w = r_w/d$ and $\gamma \geq 2$, this quantity is at most $(1 + 4c_2/d)\gamma r_w$. It follows that the ball $c\gamma b_w$ contains p for any constant $c \geq 1 + 4c_2/d$. $\square$

In order to apply the basic AVD for the purposes of answering ε-approximate nearest neighbor queries, it suffices to compute an ε-representative set for each leaf cell. For now we present a simple, albeit space-inefficient, construction.

Given the point set S and parameters $0 < \varepsilon \leq 1/2$ and $2 \leq \gamma \leq 1/\varepsilon$, we construct the BBD tree T described in Lemma 6.1, for $\beta = 1/\varepsilon$. The number of nodes in this tree is $O(n\gamma^d \log(1/\varepsilon))$. This tree and the subdivision induced by its leaves form the basis of the AVD. For each leaf cell w, let b_w and b'_w be the balls defined in Lemma 6.1. Since w is contained within the ball b_w, by Lemma 5.1(i) there exists an ε-representative set O_w of size $O(1/(\varepsilon\gamma)^{(d-1)/2})$ for w with respect to $S \cap \overline{\gamma b_w}$.

If either case (i) or (ii) of Lemma 6.1 holds, the desired ε-representative set R_w is defined to be $O_w \cup (S \cap w)$. (In case (i), $S \cap w$ is empty, and in case (ii), it consists of a single point.) Finally, if case (iii) holds, there is a ball b'_w that contains all the points of $S \cap \gamma b_w$, and b'_w is concentrically $(1/\varepsilon)$-separated from w. By Lemma 5.1(ii) there exists an ε-representative set I_w of size $O(1)$ for w with respect to $S \cap b'_w$. The representative set in this case is the union of the "outer" and "inner" representatives, that is, $R_w = O_w \cup I_w$. In any case, $|R_w| = O(1/(\varepsilon\gamma)^{(d-1)/2})$.

To answer ε-nearest neighbor queries, we first locate the leaf cell of the AVD that contains q. By BBD property (iv), this takes time $O(\log(n\gamma^d \log(1/\varepsilon))) = O(\log(n\gamma) + \log\log(1/\varepsilon))$. Later in Section 8, we will see that the $\log\log(1/\varepsilon)$ term may be ignored. We then compute the distance from q to each point of R_w and return the closest among them. Thus, the query time is $O(\log(n\gamma) + 1/(\varepsilon\gamma)^{(d-1)/2})$. A naive analysis of the space bound is provided by taking the product of the number of cells and the maximum number of representatives per cell, which yields

$O(m/(\varepsilon\gamma)^{(d-1)/2})$, but in the next two sections we will show how to improve the space requirements to $O(m)$.

7. *Spatial Amortization and Representatives*

As mentioned above, a naive analysis of the basic AVD construction yields a rather weak space bound. To obtain more accurate bounds, we will need to apply a somewhat more sophisticated analysis (along with a more careful way of selecting representatives). In this section we present such a tool, called *spatial amortization*, which is based on a more careful counting of the total number of representatives. This concept has been shown to be useful in other applications of AVDs [Arya et al. 2005, 2006, 2009], and we believe that it may be of independent interest in similar applications of geometric approximation.

Recall from the description of Section 4 that spatial amortization is a combinatorial analysis tool based on a charging scheme. We begin by describing the method in a generic setting, and in Section 8 we specialize the analysis to bound the space and time of the basic AVD construction. Given our n-element point set S, let $\mathcal{W}$ denote any collection of cells of a BBD-tree decomposition having pairwise disjoint interiors. As we shall see later, spatial amortization can be applied generally to any set of geometric objects that satisfies a packing property as given in BBD property (iii).

Let w be an arbitrary cell of $\mathcal{W}$. Recall that s_w denotes the size (side length) of w and that b_w is an enclosing ball centered at w of radius $r_w = s_w d$. Due to the nature of nearest neighbor approximation, it is reasonable to assume that w's representatives satisfy a couple of basic properties. First, the fact that w exists implies that there must be a well-separated pair nearby (relative to r_w) that caused w's parent to split (recall Lemma 6.3). Thus, there exists a parameter $\rho \geq 1$ (which will depend on the parameter γ used in the construction) such that all nearest neighbor representatives of w lie within the expanded ball ρb_w. Second, since queries need only be answered approximately, we may generally assume that the representatives are sparse, in the sense that they may be selected so that no more than a constant number lie within any region of diameter $O(\delta \rho r_w)$, for some small $\delta > 0$ (which will depend on the approximation factor ε). To model these constraints, we assume that we are given two parameters $\rho \geq 1$ and $\delta > 0$, and each cell w is associated with a collection of quadtree boxes $\mathcal{U}_w$ that satisfy the following conditions:

—*Nonempty*: Each box of $\mathcal{U}_w$ contains at least one point of S.
—*Local*: The boxes of $\mathcal{U}_w$ overlap the ball ρb_w.
—*Sparse*: The boxes of $\mathcal{U}_w$ are of size $s'_w = \delta \rho r_w$. (More accurately the boxes of $\mathcal{U}_w$ are of size $2^{\lfloor \log s'_w \rfloor}$.)

These boxes of $\mathcal{U}_w$ should not to be confused with the cells of $\mathcal{W}$. Instead, think of them as some subset of the boxes of a rectangular grid that lie within ρb_w and that contain at least one point of S (see Figure 11(a)). We will show that, for a suitable choice of ρ and δ, the total number of representatives can be bounded asymptotically by $\sum_{w \in \mathcal{W}} |\mathcal{U}_w|$. We will then apply spatial amortization to bound this sum.

The naive approach to bounding this sum would be to first bound the worst-case size of $\mathcal{U}_w$ (say, by a packing argument) and then multiply by the number of cells
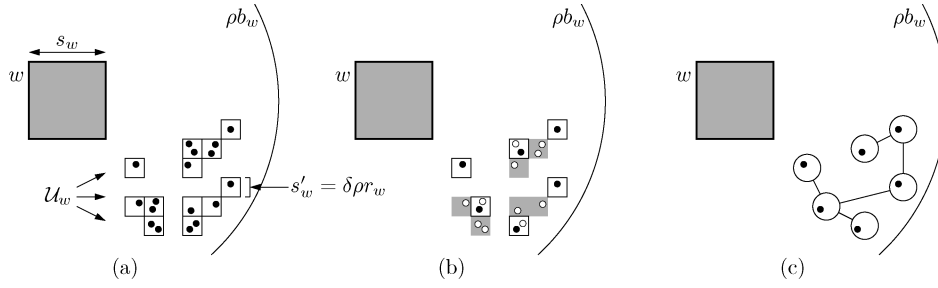
FIG. 11. Spatial amortization and Lemma 7.1.

in $\mathcal{W}$. Unfortunately, this produces a bound that is far from optimal, since most cells cannot achieve the worst case. This is where spatial amortization comes to the rescue. First, we show that the set $\mathcal{U}_w$ can be identified with a roughly equal number of dumbbells of a well-separated pair decomposition (WSPD) of S (denoted by $\mathcal{P}_w$ below), where each pair separates two points whose distance is at least as large as the size of the boxes of $\mathcal{U}_w$. Next, we develop a charging scheme in which each dumbbell of the WSPD charges a subset of cells of $\mathcal{W}$ so that the number of charges received by each cell is proportional to $|\mathcal{U}_w|$. Finally, we bound the total number of such charges over all the dumbbells.

We start by showing that the process of counting quadtree boxes of $\mathcal{U}_w$ can be reduced to counting an appropriate subset of dumbbells of a WSPD. Henceforth, let $\mathcal{P}$ denote the set of dumbbells of a WSPD of S for some constant separation factor $\sigma > 4$. Let $S_w \subseteq S$ be a set of *points of interest* contained in the union of the boxes of $\mathcal{U}_w$ such that each box of $\mathcal{U}_w$ contains at least one point of S_w. We say that a dumbbell $P \in \mathcal{P}$ is *useful* for w if it separates some pair of points $x, y \in S_w$ such that $\|xy\| \geq s'_w$. Let $\mathcal{P}_w$ denote the set of dumbbells that are useful for w.

LEMMA 7.1. *Let $\mathcal{P}$ be the set of dumbbells of a WSPD for a point set S for any constant separation factor $\sigma > 4$. Given a cell w, let $\mathcal{U}_w$, S_w, and $\mathcal{P}_w$ be as defined above. Then $|\mathcal{U}_w| = O(|\mathcal{P}_w| + 1)$.*

PROOF. We first identify a subset of points $S'_w \subseteq S_w$ such that $|\mathcal{U}_w| = O(|S'_w|)$, and the distance between any pair of points of S'_w is at least s'_w. We start with S'_w being empty and consider the quadtree boxes of $\mathcal{U}_w$ one by one. For each quadtree box u examined, we add any one point of $S_w \cap u$ to S'_w and then eliminate the at most 3^d quadtree boxes in $\mathcal{U}_w$ that share a common boundary with u from future consideration. We continue in this manner until all the quadtree boxes are eliminated. It is clear that this process yields a subset S'_w with the properties mentioned above. (These are the black points of Figure 11(b).)

Consider the following process for identifying a set $\mathcal{P}'_w$ of useful dumbbells. At each step, we find the dumbbell of $\mathcal{P}$ that separates the closest pair of points among the remaining points of S'_w and add it to $\mathcal{P}'_w$. We then eliminate one of these two points. We repeat this process with the remaining points until only one point of S'_w remains. Let $\mathcal{P}'_w$ denote the resulting set of dumbbells (see Figure 11(c)).

Because the closest pair is chosen at each stage, and the separation factor is greater than 4, the dumbbell head containing the eliminated point at each stage contains no other point from among the uneliminated points of S'_w. It follows therefore that all

the dumbbells obtained are distinct. Thus, $|\mathcal{P}'_w| = |S'_w| - 1$. Since $|\mathcal{U}_w| = O(|S'_w|)$, we have $|\mathcal{U}_w| = O(|\mathcal{P}'_w| + 1) = O(|\mathcal{P}_w| + 1)$, as desired. □

We define a *charging scheme* to be a process in which each dumbbell of our WSPD $\mathcal{P}$ allocates a *unit charge* to a subset of cells of $\mathcal{W}$. Recall that $\mathcal{P}_w \subseteq \mathcal{P}$ is the set of dumbbells that are useful for w. A charging scheme is *valid* if, for any $w \in \mathcal{W}$, all the dumbbells of $\mathcal{P}_w$ allocate a unit charge to w. (Note that a cell may receive charges from multiple dumbbells and may receive charges from dumbbells that are not in $\mathcal{P}_w$.) It follows easily that a valid charging scheme can be used to achieve our goal of bounding the sum of the cardinalities of $\mathcal{U}_w$ over all cells w, as shown next.

LEMMA 7.2. *Let Ψ denote the total number of charges allocated to the cells of $\mathcal{W}$ by a valid charging scheme. Then*

$$\sum_{w \in \mathcal{W}} |\mathcal{U}_w| \; = \; O(\Psi + |\mathcal{W}|).$$

PROOF. By Lemma 7.1, we have $|\mathcal{U}_w| = O(|\mathcal{P}_w| + 1)$. Since the charging scheme is valid, each dumbbell of $\mathcal{P}_w$ allocates a unit charge to cell w and so $\sum_w |\mathcal{P}_w| \leq \Psi$. Therefore, $\sum_w |\mathcal{U}_w| = O(\sum_w |\mathcal{P}_w| + \sum_w 1) = O(\Psi + |\mathcal{W}|)$. □

In order to carry out the charging analysis suggested by Lemma 7.2, a charging scheme should be set up so that it is easy to calculate the number of cells charged by each dumbbell. This will usually be done through a straightforward packing argument. The following lemma provides a generic analysis of spatial amortization, which will be applied later in a number of specific instances.

LEMMA 7.3 (BASIC SPATIAL AMORTIZATION). *Let S be a set of n points in $\mathbb{R}^d$, and let $\mathcal{W}$ be the leaf cells of a BBD-tree decomposition. Let $\rho \geq 1$ and $0 < \delta \leq 1$ be two real parameters. For each $w \in \mathcal{W}$, let $\mathcal{U}_w$ be the set of nonempty quadtree boxes of size $\delta \rho r_w$ that overlap ρb_w. Then*

$$\sum_{w \in \mathcal{W}} |\mathcal{U}_w| \; = \; O\left(n\rho^d \log \frac{1}{\delta} \; + \; |\mathcal{W}|\right).$$

Note that the bound is relatively sensitive to locality, as evidenced by the ρ^d term, but the dependence on sparseness is quite mild, growing only logarithmically with δ. To better appreciate the power of spatial amortization, let us consider what bound would be produced by a naive analysis. Since the sizes of the quadtree boxes of $\mathcal{U}_w$ are smaller than the radius of ρb_w by a factor of δ, it follows from a simple packing argument that $|\mathcal{U}_w| = O(1/\delta^d)$. Summing this over all the cells of $\mathcal{W}$ yields $\sum_{w \in \mathcal{W}} |\mathcal{U}_w| = O(|\mathcal{W}|/\delta^d)$. If we were to apply this to our AVD construction (as we will do in the next section) we would obtain a bound on $|\mathcal{W}|$ that is roughly $O(n\rho^d)$. The resulting bound of $O(n(\rho/\delta)^d)$ obtained by this naive analysis would be bigger by a factor of roughly $\Omega(1/\delta^d)$. When we apply this to our AVD construction, δ can be as small as ε, and so this additional factor would be quite significant.

PROOF OF LEMMA 7.3. Let c_1 and c_2 be two constants, both assumed to be suitably large. Let $\mathcal{P}$ be the set of dumbbells corresponding to the WSPD for S, using any constant separation factor greater than 4. Each dumbbell $P \in \mathcal{P}$ allocates a unit charge to some of the cells of $\mathcal{W}$ according to the following charging scheme,
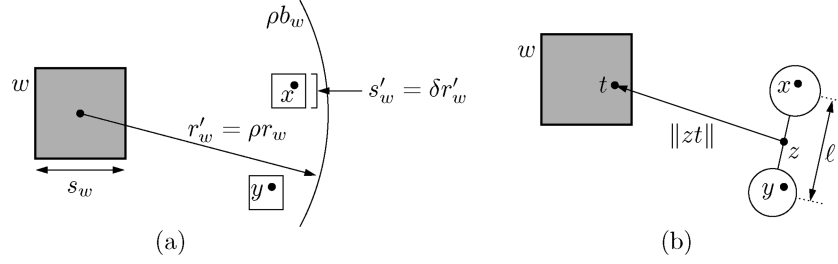
FIG. 12. Establishing the validity of the charging scheme of Lemma 7.3.

which we will show below to be valid. Let ℓ denote P's length, and let z denote its center. Let $\mathcal{B}(P)$ denote the set of balls of radii $r_i = 2^i\ell$ centered at z, for $0 \le i \le \lceil \log(c_1/\delta) \rceil$. Let $b_i(P)$ denote the ball of $\mathcal{B}(P)$ of radius r_i, and let $\mathcal{W}_i(P) \subseteq \mathcal{W}$ be the set of cells of the BBD-tree decomposition overlapping $b_i(P)$ that have size at least $r_i/(c_2\rho)$. (The process is very reminiscent of the AVD construction, illustrated in Figure 6 of Section 4.) Dumbbell P allocates a unit charge to each of the cells of the set $\bigcup_{b_i(P)\in\mathcal{B}(P)} \mathcal{W}_i(P)$.

By BBD property (iii), it follows that $|\mathcal{W}_i(P)| = O(\rho^d)$. Thus, the number of cells charged by P is $\sum_i |\mathcal{W}_i(P)| = O(\rho^d \log(1/\delta))$. Since $|\mathcal{P}| = O(n)$, the total number of charges allocated to all the cells of $\mathcal{W}$ is $\Psi = O(n\rho^d \log(1/\delta))$. Thus, once we have established that this charging scheme is valid, it will then follow from Lemma 7.2 that $\sum_w |\mathcal{U}_w| = O(n\rho^d \log(1/\delta) + |\mathcal{W}|)$, as desired.

To establish validity, let S_w be the subset of points of S that lie within some box of $\mathcal{U}_w$, and recall that $\mathcal{P}_w$ denotes the associated set of useful dumbbells for w. We will show that each of these dumbbells charges w. Let $r'_w = \rho r_w$ denote the radius of ρb_w, and let $s'_w = \delta r'_w$ be the size of the quadtree boxes of $\mathcal{U}_w$ (see Figure 12(a)). By definition, each dumbbell $P \in \mathcal{P}_w$ separates some pair of points $x, y \in S_w$ such that $\|xy\| \ge s'_w$. As before, let ℓ denote P's length, and let z be its center (see Figure 12(b)).

By Lemma 3.1, it follows that ℓ, $\|xy\|$, and $\|xz\|$ are all within a constant factor of each other. Let t be an arbitrary point in w. Since both x and y lie within quadtree boxes of size $s'_w \le r'_w$ that overlap the ball ρb_w, it follows that $\|xt\|$, $\|yt\|$, and $\|xy\|$ are all $O(r'_w)$. Therefore we have

$$\|zt\| \le \|xz\| + \|xt\| = O(r'_w) = O\left(\frac{s'_w}{\delta}\right).$$

By hypothesis, $\|xy\| \ge s'_w$, and since ℓ is within a constant factor of $\|xy\|$ we have $\|zt\| = O(\ell/\delta)$.

Therefore, for any suitably large constant c_1, the largest ball of $\mathcal{B}(P)$ contains t, and so w is eligible to be considered for charging. To see that it will be charged, let r denote the radius of the smallest ball $b \in \mathcal{B}(P)$ that contains t. If b is the smallest ball of $\mathcal{B}(P)$, then its radius is ℓ, which we have shown to be $O(r'_w)$. Otherwise, because the ball radii grow by factors of 2 we have $r \le 2\|zt\| = O(r'_w)$. Recalling that $r'_w = \rho r_w = \rho s_w d$ and that P allocates a charge to all cells of $\mathcal{W}$ overlapping b that have size at least $r/(c_2\rho)$, it is clear that, for sufficiently large c_2, w must receive a charge from P. This completes the proof. □

We observe as well that with the aid of the auxiliary BBD tree T_A for S (recall the comments made just after the proof of Lemma 6.2) we obtain the following computational result.

LEMMA 7.4.   *Given the same conditions as in Lemma 7.3 and the auxiliary BBD tree T_A for the points of S, we can compute the nonempty quadtree boxes $\mathcal{U}_w$ for all $w \in \mathcal{W}$ and an arbitrary point of S from each such box in time*

$$O\left(n\rho^d \log \frac{1}{\delta} + |\mathcal{W}| \log n\right).$$

PROOF.   By BBD property (v), for each $w \in \mathcal{W}$ we can compute $\mathcal{U}_w$ and an arbitrary point within each box of $\mathcal{U}_w$ in time $O(\log n + t_w)$, where t_w is the number of nonempty quadtree boxes of size $\delta\rho r_w$ that overlap the factor-2 expansion of ρb_w. We can bound $\sum_{w \in \mathcal{W}} t_w$ in exactly the same way we bounded $\sum_{w \in \mathcal{W}} |\mathcal{U}_w|$ in Lemma 7.3 (since all that has changed is the doubling of the radius of ρb_w, which only alters the constant factors). The time bound given in the statement of the lemma follows directly.   □

## 8. *Bounding the Total Space for the Basic AVD*

We are now in a position to apply spatial amortization to bound the total number of nearest-neighbor representatives as well as the preprocessing time for computing them. To begin, assume that we have applied Lemma 6.1 to construct a BBD tree T, where the parameter β has been chosen to be $16/\varepsilon$. (We will justify this choice below.) Let $\mathcal{W}$ denote the subdivision induced by the leaf cells of T. Lemma 6.1 implies that the total number of cells of $\mathcal{W}$ is $O(n\gamma^d \log(1/\varepsilon))$. As we did in this lemma, let $m = m(n, d, \gamma, \beta) = n\gamma^d \log(1/\varepsilon)$ denote the asymptotic bound on the number of cells. (Note that we have omitted the factor of 16 to simplify the expression, but there is no harm in doing so since it will only be used in asymptotic expressions.)

Our goal is to compute an ε-representative set for each cell $w \in \mathcal{W}$. Recall that this is a subset $R_w \subseteq S$ such that for any query point $q \in w$, its nearest neighbor in R_w is an ε-approximation to its nearest neighbor in S. The following lemma provides the main technical result of this section. It bounds the maximum and total sizes of these representative sets and their total computation time.

LEMMA 8.1.   *Let $0 < \varepsilon \leq 1/2$ and $2 \leq \gamma \leq 1/\varepsilon$ be two real parameters. Let S be a set of n points in $\mathbb{R}^d$, and let T be the BBD tree described above. Let $\mathcal{W}$ denote T's leaf cells. Then, for each cell $w \in \mathcal{W}$, there exists an ε-representative set R_w for w, such that $|R_w| = O(1/(\varepsilon\gamma)^{(d-1)/2})$. The total number of representatives over all the cells is $O(m) = O(n\gamma^d \log(1/\varepsilon))$. Moreover, it is possible to compute the sets R_w for all cells $w \in \mathcal{W}$ in total time*

$$O\left(m\left(\left(\frac{1}{\varepsilon\gamma}\right)^{\frac{d-1}{2}} + \log m\right)\right).$$

The remainder of this section is devoted to proving this lemma. Throughout, let $\varepsilon, \gamma, S, n, \mathcal{W}$, and m be as specified in the statement of the above lemma, and let w be any cell of $\mathcal{W}$. We begin by showing that, given our choice of $\beta = 16/\varepsilon$ in the application of Lemma 6.1, all but one of the representatives may be assumed
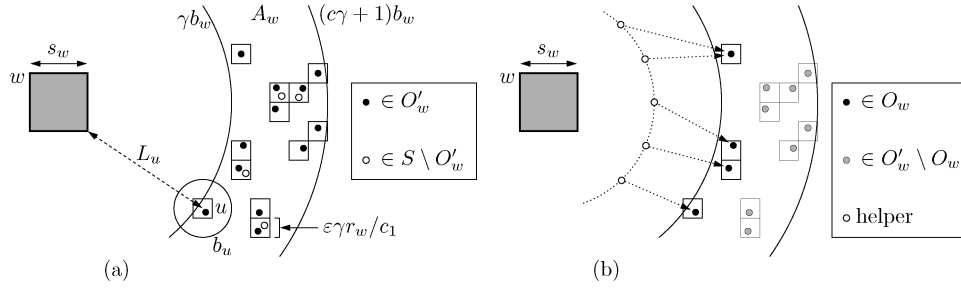
FIG. 13. Computing the outer representatives for the cell w.

to be taken from outside w's outer ball. Based on the separation properties given in Lemma 6.1, for a query point lying within w there are three possible sources of ε-nearest neighbor representatives: (a) from within w, (b) within the inner ball b'_w, and (c) from outside the outer ball γb_w. (Recall Figure 8 of Section 6.) The representatives of type (c) are called the ε-*outer representatives*, which we will denote by O_w. Formally, we define O_w to be an ε-representative set for w with respect to the points of S lying outside γb_w that are nearest neighbors of some query point in w. If w satisfies condition (i) of Lemma 6.1, the outer ball contains no points of S, and so $R_w \leftarrow O_w$. If (ii) holds, then there is only one point p within the outer ball, and so we may take the final representatives to be $R_w \leftarrow O_w \cup \{p\}$. (In fact, it can be shown that since $\gamma \geq 2$, the nearest neighbor of any point in w will be p, so we do not need O_w at all.) Finally, if (iii) holds, then by our choice of β, all the points lying within the inner ball b'_w are concentrically $(16/\varepsilon)$-separated from w. By Lemma 5.2, it follows therefore that it suffices to take an arbitrary point p as the representative from the inner ball. (The construction provides such a point.) The final set of representatives will be $R_w \leftarrow O_w \cup \{p\}$.

Thus, given the sets O_w for all $w \in \mathcal{W}$ we can compute the complete representative sets R_w in $O(1)$ additional time per cell, that is, $O(m)$ additional time overall. Hence, for the rest of this section, it suffices to consider the problem of computing just the outer representatives O_w for each of the cells $w \in \mathcal{W}$. One way to do this would be to implement the construction implied by the proof of Lemma 5.1. We will describe a significantly more efficient method here. For the sake of clarity, we first ignore issues related to preprocessing time, which we will address a bit later.

By Lemma 6.3, there is a constant $c > 1$ such that the ball $c\gamma b_w$ contains at least one point of S. It follows from the triangle inequality that our search for the outer representatives can be restricted to the annulus $A_w = (c\gamma + 1)b_w \setminus \gamma b_w$. Let $\mathcal{U}_w$ denote the set of nonempty quadtree boxes of size $\varepsilon \gamma r_w / c_1$ overlapping A_w, where c_1 is a suitably large constant (see Figure 13(a)).

Our approach will be to construct an initial set O'_w of representatives, which will be larger than needed, and then prune this set to the desired set O_w. For each $u \in \mathcal{U}_w$, let p_u be an arbitrary point of $S \cap u$, and let $O'_w = \bigcup_{u \in \mathcal{U}_w} \{p_u\}$. Since the cells of $\mathcal{U}_w$ are at distance $\Theta(\gamma r_w)$ from w, and their size is smaller by a factor of $O(\varepsilon)$, we would expect that the points of O'_w form an $O(\varepsilon)$-outer representative set. To compensate for later additional approximations, it will be useful to engineer the approximation factor to be a bit smaller than ε.

LEMMA 8.2. *For all sufficiently large constants c_1 (in the definition of $\mathcal{U}_w$) O'_w is an $(\varepsilon/4)$-outer representative set for w.*

PROOF.   For any box $u \in \mathcal{U}_w$, let $s_u = \varepsilon \gamma r_w / c_1$ be its size, and let b_u denote the enclosing ball of radius $r_u = s_u d$ centered about u. We claim that b_u is concentrically $(64/\varepsilon)$-separated from w. Assuming this claim for now, since $64/\varepsilon = (16/(\varepsilon/4))$, it follows from Lemma 5.2 that for any query point $q \in w$, $\|q p_u\| \leq (1 + \varepsilon/4) \mathrm{NN}_q(S \cap u)$. That is, $\{p_u\}$ is an $(\varepsilon/4)$-representative set for w with respect to $S \cap u$. We have seen that each point of $S \cap \overline{\gamma b_w}$ that is the nearest neighbor of some point in w lies within A_w and therefore lies within some box of $\mathcal{U}_w$. It follows directly that O'_w is an $(\varepsilon/4)$-outer representative set for w.

To complete the proof, we establish the above claim. Let L_u denote the distance between the center of ball b_u and w (see Figure 13(a)). It suffices to show that $L_u / r_u \geq 64/\varepsilon$. Recall that $r_w = s_w d$ and $\varepsilon \leq 1/2$. Setting $c_1 \geq 4d$, we obtain

$$r_u = s_u d = \frac{\varepsilon \gamma r_w d}{c_1} \leq \frac{\gamma r_w}{8}.$$

Since $\gamma \geq 2$, and b_u overlaps $\overline{\gamma b_w}$, by applying the triangle inequality we have

$$L_u \geq \gamma r_w - r_u - r_w \geq \gamma r_w - \frac{\gamma r_w}{8} - \frac{\gamma r_w}{2} = \frac{3 \gamma r_w}{8}.$$

Thus, setting $c_1 \geq 64 \cdot 8d/3$ implies that

$$\frac{L_u}{r_u} \geq \frac{(3 \gamma r_w / 8)}{(\varepsilon \gamma r_w d / c_1)} = \frac{3 c_1}{8 \varepsilon d} \geq \frac{64}{\varepsilon}.$$

as desired.   □

In view of this lemma, we could let O'_w be the set of outer representatives for w, but their number would be too large to satisfy our desired bounds on query times. (A simple packing argument shows that in the worst case, $|O'_w|$ could be as large as $\Theta(1/\varepsilon^d)$.) We will discuss how to reduce the number later, but first we show that, through the use of spatial amortization, we can bound the total number of representatives over all the cells of $\mathcal{W}$ to be $O(m)$. This is in contrast to the much higher bound of $O(m/(\varepsilon \gamma)^{(d-1)/2})$ arising from the naive analysis.

LEMMA 8.3.   *For each cell $w \in \mathcal{W}$, let O'_w be the set of representatives defined above. Then $\sum_w |O'_w| = O(m)$. Furthermore, we can compute O'_w for all the cells of $\mathcal{W}$ in time $O(m \log n)$.*

PROOF.   Since $|O'_w| = |\mathcal{U}_w|$ it suffices to bound $\sum_w |\mathcal{U}_w|$. As observed before, all the boxes of $\mathcal{U}_w$ overlap the ball $(c\gamma + 1) b_w$. Since $\gamma \geq 2$, this ball is contained in $c_2 \gamma b_w$ for a suitable constant c_2. To bound $\sum_w |\mathcal{U}_w|$, we apply Lemma 7.3 with $\rho = c_2 \gamma$ and $\delta = \varepsilon/(c_1 c_2)$. It is easy to see that all the conditions of Lemma 7.3 are met, and so we have

$$\sum_{w \in \mathcal{W}} |\mathcal{U}_w| = O\left(n \gamma^d \log \frac{1}{\varepsilon} + m\right) = O(m).$$

We can compute the points of O'_w by sampling an arbitrary point from each of the boxes of $\mathcal{U}_w$. By Lemma 7.4, this can be done in time $O(n \gamma^d \log(1/\varepsilon) + |\mathcal{W}| \log n) = O(m \log n)$.   □

To produce the final outer representative set, recall from the proof of Lemma 8.2 that, for any $u \in \mathcal{U}_w$, $r_u \leq \gamma r_w / 8$. Thus, by the triangle inequality, all the points

of O'_w are at distance at least $\gamma r_w - 2r_u \geq (3\gamma/4)r_w$ from the center of w, that is, they lie outside $(3\gamma/4)b_w$. Thus, by Lemma 5.1 there exists a subset $O_w \subseteq O'_w$ of size $O(1/(\varepsilon\gamma)^{(d-1)/2})$ such that $\mathrm{NN}_q(O_w) \leq (1+\varepsilon/4)\mathrm{NN}_q(O'_w)$ (see Figure 13(b)). Combining this with Lemma 8.2 and our assumption that $\varepsilon \leq 1/2$, it follows that O_w is the desired outer ε-representative set for w. Since $O_w \subseteq O'_w$, it follows that $\sum_w |O_w| = O(m)$.

Now let us consider how to compute these representative sets for all $w \in \mathcal{W}$. Recall from Lemma 6.1 the BBD tree T can be constructed in time $O(m \log m)$. We have just seen that the sets O'_w can be computed for all the cells of $\mathcal{W}$ in time $O(m \log n)$. It remains only to consider the time it takes to prune O'_w to O_w. The proof of Lemma 5.1, which was used to establish the existence of O_w, shows that there exists a set of $O(1/(\varepsilon\gamma)^{(d-1)/2})$ points, called *helpers*, and the points of O_w can be taken to be the $(\varepsilon/2)$-approximate nearest neighbors of these helpers. Thus, for each helper it suffices to compute its exact nearest neighbor from O'_w, which is done by a simple brute-force scan of the points of O'_w. Summing over all the cells yields a total asymptotic time of

$$\left(\frac{1}{\varepsilon\gamma}\right)^{\frac{d-1}{2}} \sum_w |O'_w| \;=\; O\left(\left(\frac{1}{\varepsilon\gamma}\right)^{\frac{d-1}{2}} m\right).$$

Recall that once the sets O_w have been computed, the final representative sets R_w can be computed in additional $O(m)$ time. This establishes the computation time for Lemma 8.1, and completes its proof.

To answer an ε-NN query for a point q, we first determine the leaf cell that contains q. This takes time $O(\log m) = O(\log(n\gamma) + \log\log(1/\varepsilon))$ time by a simple descent in the BBD tree. We then compute the distance from q of each point in R_w and return the closest among them, which can be done in time $O(1/(\varepsilon\gamma)^{(d-1)/2})$. The total query time is the sum of these two quantities. Note that $\log\log(1/\varepsilon)$ is never dominant and so may be ignored. This follows because

$$\log\log\frac{1}{\varepsilon} \;\leq\; \log\frac{1}{\varepsilon} \;=\; \log\gamma \;+\; \log\frac{1}{\varepsilon\gamma} \;\leq\; \log(n\gamma) \;+\; \left(\frac{1}{\varepsilon\gamma}\right)^{\frac{d-1}{2}}.$$

Therefore the overall query time is $O(\log(n\gamma) + 1/(\varepsilon\gamma)^{(d-1)/2})$.

We can now provide the complete analysis of the basic AVD construction of Section 6.

THEOREM 8.4 (BASIC AVD THEOREM). *Let S be a set of n points in $\mathbb{R}^d$, and let $0 < \varepsilon \leq 1/2$ and $2 \leq \gamma \leq 1/\varepsilon$ be two real parameters. Let $m = n\gamma^d \log(1/\varepsilon)$. We can construct a (t, ε)-AVD, where $t = O(1/(\varepsilon\gamma)^{(d-1)/2})$, of space $O(m)$ that can answer ε-NN queries in time $O(\log(n\gamma) + t)$. It can be constructed in time $O(m(t + \log m))$.*

By considering the two extreme cases, $\gamma = 2$ and $\gamma = 1/\varepsilon$, we have:

COROLLARY 8.5. *Let S be a set of n points in $\mathbb{R}^d$, and let $0 < \varepsilon \leq 1/2$. Then:*

(i) *Let $m' = n \log(1/\varepsilon)$. There exists a (t, ε)-AVD, where $t = O(1/\varepsilon^{(d-1)/2})$, of space $O(m')$ that can answer ε-NN queries in time $O(\log n + t)$. It can be constructed in time $O(m'(t + \log m'))$.*

(ii) *Let $m'' = (n/\varepsilon^d)\log(1/\varepsilon)$. There exists a (t, ε)-AVD, where $t = O(1)$, of space $O(m'')$ that can answer ε-NN queries in time $O(\log(n/\varepsilon))$. It can be constructed in time $O(m'' \log m'')$.*

When $\gamma = 1/\varepsilon$, we need only $O(1)$ representatives per cell. It is natural to wonder whether this number can be reduced to exactly one representative per cell, while retaining the asymptotic space and query bounds of Corollary 8.5(ii). The answer is yes. By setting $\gamma = 32/\varepsilon$ and $\beta = 16/\varepsilon$ in the construction of the BBD tree of Lemma 6.1, it can be shown that a single representative suffices for each node. We will spare the reader the straightforward technical details, but here is a sketch. The factor γ is sufficiently large that Lemma 5.2 implies that a single representative suffices from beyond the outer ball γb_w. Similarly, β is sufficiently large that, if the inner ball b'_w exists, a single representative suffices from within it. If there is an inner ball, we use its representative, and otherwise, we use the representative from the outer ball.

Readers familiar with the earlier version of this article [Arya et al. 2002] may have noted that the bounds on the number of cells m in the above theorem and corollary carry an additional factor of $\log(1/\varepsilon)$. In the earlier paper we showed that this additional term could be eliminated through a combination of elements involving deterministic sampling, a more complex construction involving a quadratic growth rate, and a second separation condition (analogous to Lemma 5.1) based on disjoint balls. We chose to simplify the presentation by eliminating these elements at the price of this small additional factor.

9. *Bisector-Sensitive AVDs*

Our bounds on the number of cells in our various constructions so far have been suboptimal since they carry additional factors of γ and/or ε. For example, ignoring logarithmic factors our bound on the number of cells is $O(n\gamma^d)$, while $O(n\gamma^{d-1})$ would have been expected. To gain some intuition as to the source of this inefficiency, observe that our constructions process the space around each well-separated pair in a uniform manner. For example, given a well-separated pair $P = (X, Y)$, the basic construction of Section 6 generates quadtree boxes uniformly throughout a ball centered about the associated dumbbell. Intuitively, one may view these boxes as arising from a need to ascertain whether the approximate nearest neighbor lies in X or Y. This point of view suggests that we only need to generate those boxes that contain a query point q such that the respective nearest-neighbor distances, $\mathrm{NN}_q(X)$ and $\mathrm{NN}_q(Y)$, are approximately equal. That is, the boxes of interest are those that overlap the bisector between the sets X and Y. Since X and Y are well separated, the bisector is a $(d - 1)$-dimensional polyhedral manifold, and hence the number of boxes needed to cover it should grow as γ^{d-1} rather than γ^d. In this section we shall see that, by exploiting this observation, it is possible to construct an AVD having only $O(n\gamma^{d-1})$ cells. An efficient implementation of this construction would involve a number of technical geometric details. To simplify the presentation, we will not discuss preprocessing time, and instead we will only establish the existence of an AVD of the desired size.

In the next section, we will derive the relevant geometric properties of bisectors of well-separated sets. Then, in Section 9.2, we present our bisector-sensitive AVD construction and establish its separation properties. Finally, in Section 9.3, we
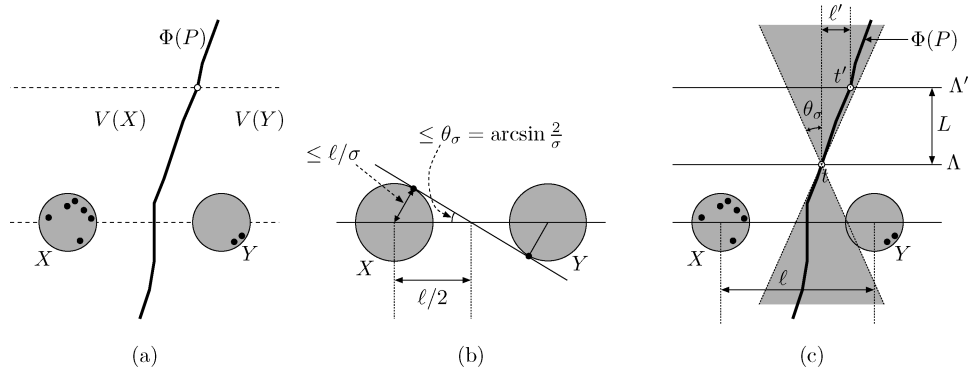
FIG. 14. The Voronoi bisector of a well-separated pair and Lemma 9.1.

analyze the total number of representatives needed and present the total space requirements.

9.1. BISECTOR PROPERTIES. Consider a WSPD with constant separation factor $\sigma > 4$. Throughout this section we will focus on a single well-separated pair $P = (X, Y)$ of the decomposition. Let z denote their midpoint, or the dumbbell *center*. Recall that the line passing through the centers of the dumbbell heads is called P's *axis*, and the distance between the centers of these heads, denoted ℓ, is P's *length*. Recall also that the distance between a point and a set is the distance to its closest point within the set.

Define the *Voronoi regions* $V(X) = \{q : dist(q, X) \leq dist(q, Y)\}$ and $V(Y) = \{q : dist(q, Y) \leq dist(q, X)\}$ (see Figure 14(a)). Define P's *bisector*, denoted $\Phi(P)$ to be the points that are equidistant to X and Y, that is, $\Phi(P) = V(X) \cap V(Y)$. Because the sets X and Y are linearly separated (since $\sigma > 2$), it follows from standard results on Voronoi diagrams [Preparata and Shamos 1990] that these Voronoi regions are each connected sets and the bisector $\Phi(P)$ is a connected $(d-1)$-dimensional polyhedral manifold, each of whose facets lies on the bisecting hyperplane between some pair of points $x \in X$ and $y \in Y$. It follows that any line that is parallel to P's axis intersects $\Phi(P)$ in a single point. (As the separation factor σ increases $\Phi(P)$ approaches the perpendicular bisector of the dumbbell centers.) The following lemma presents a bound on the degree of deviation from the limiting case as a function of σ.

LEMMA 9.1. *Let $P = (X, Y)$ be a well-separated pair of length ℓ with separation factor $\sigma > 4$. Let Λ and Λ' be two lines that are parallel to P's axis, and let L denote the distance between these lines. Let t and t' denote the respective intersection points of Λ and Λ' with $\Phi(P)$. Then the component of the distance between t and t' that is parallel to P's axis (shown as ℓ' in Figure 14(c)) is at most $3L/\sigma$.*

PROOF. P's dumbbell heads are of radius at most ℓ/σ. By considering the cross tangents between these heads (see Figure 14(b)) it is easy to see that the line joining any two points $x' \in X$ and $y' \in Y$ forms an angle with respect to P's axis of at

most $\theta_\sigma = \arcsin(2/\sigma)$. Using our hypothesis that $\sigma > 4$, we have

$$\tan^2 \theta_\sigma = \frac{\sin^2 \theta_\sigma}{1 - \sin^2 \theta_\sigma} = \frac{4/\sigma^2}{1 - (4/\sigma^2)} = \frac{4}{\sigma^2 - 4} < \frac{9}{\sigma^2}.$$

Thus, $\tan \theta_\sigma < 3/\sigma$.

As mentioned above, the facets that make up the bisector $\Phi(P)$ each lie on the $(d-1)$-dimensional bisecting flat between a pair of points, one in X and the other in Y. By the symmetry of the spherical dumbbell heads, it follows easily that each such bisecting flat forms an angle of at least $(\pi/2) - \theta_\sigma$ with respect to any line that is parallel to P's axis.

Consider the 2-dimensional flat passing through Λ and Λ'. For the sake of illustration, take these lines to be horizontal on this flat. (See Figure 14(c). Note that this 2-dimensional drawing is a bit misleading since P's axis need not lie on this flat.) The intersection of $\Phi(P)$ with this flat is a vertically monotone polyline in which each edge forms an absolute angle of at most θ_σ with respect to vertical. It follows easily that if we travel a vertical distance of L along the intersection of $\Phi(P)$ with the flat, the horizontal component of the distance (shown as ℓ' in the figure) cannot exceed $L \tan \theta_\sigma \leq 3L/\sigma$, as desired. □

Using this lemma, we will prove a packing lemma analogous to BBD property (iii), but restricted now just to objects that intersect the bisector of a well-separated pair.

LEMMA 9.2.   *Consider a well-separated pair $P = (X, Y)$ with separation factor $\sigma > 4$. Let b be a ball of radius r centered at P's dumbbell center. Then, for any positive s, the following hold.*

  (i) *It is possible to cover $b \cap \Phi(P)$ with $O((1 + r/s)^{d-1})$ balls of radius s.*
 (ii) *The number of BBD-tree cells of size s intersecting $b \cap \Phi(P)$ is $O((1+r/s)^{d-1})$.*
(iii) *The number of quadtree boxes of size s intersecting $b \cap \Phi(P)$ is $O((1+r/s)^{d-1})$.*

PROOF.   We begin by proving assertion (i). Let H be the $(d-1)$-flat that passes through P's dumbbell center and is orthogonal to P's axis. Let $b' = b \cap H$. Clearly, b' is a $(d-1)$-dimensional ball of radius r, and by a simple packing argument we can cover b' by a collection $\mathcal{B}'$ of balls centered on H each of radius $s/2$ (see Figure 15(a)), such that $|\mathcal{B}'| = O((1 + r/s)^{d-1})$. For each ball of $\mathcal{B}'$, consider the line Λ parallel to P's axis that passes through the center of this ball, and consider the cylinder C_Λ of radius $s/2$ centered about this line. Since the balls of $\mathcal{B}'$ cover b', the union of these cylinders covers b.

To establish (i), it suffices to show that for each Λ, it is possible to cover $C_\Lambda \cap \Phi(P)$ with $O(1)$ balls of radius s. To see this, consider a set of points placed along Λ spaced uniformly at intervals of length $s\sqrt{3}$, and place a ball of radius s centered at each point (see Figure 15(b)). It is easy to verify that these balls cover C_Λ. Let t denote the point of intersection between $\Phi(P)$ and Λ. Consider any other line Λ' that is parallel to Λ and lies within C_Λ, and let t' denote its intersection with $\Phi(P)$. These two lines are separated by a distance of at most $s/2$. By Lemma 9.1, it follows that, the component of the distance between t and t' along the direction of Λ is at most $3(s/2)/\sigma < 3s/8$. Therefore, a constant number of balls along Λ suffice to cover $C_\Lambda \cap \Phi(P)$, and this completes assertion (i). Let $\mathcal{B}$ denote the resulting set of size $O(|\mathcal{B}'|) = O((1 + r/s)^{d-1})$ covering balls.
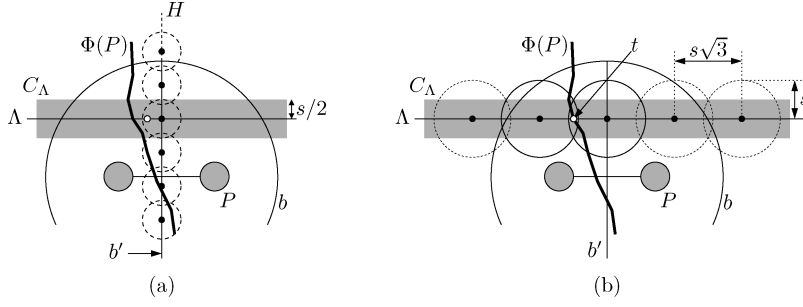
FIG. 15. Proof of Lemma 9.2.

To establish (ii), observe that BBD property (iii) implies that the number of BBD-tree cells of size s that intersect each ball of $\mathcal{B}$ is $O(1 + s/s)^d) = O(1)$. The number of BBD-tree cells overlapping $b \cap \Phi(P)$ cannot be any larger. Finally, to establish (iii), observe that quadtree boxes are a special case of BBD-tree cells (without an inner box), and so BBD property (iii) can be applied here in exactly the same way. □

9.2. BISECTOR-SENSITIVE CONSTRUCTION. In this section, we modify the basic AVD construction given in Section 6 to be sensitive to the bisector defined by each well-separated pair. As before, the approach is to generate a subdivision whose cells satisfy certain separation properties with respect to the points of S. The separation properties are the same as those of Lemma 6.1 with only a couple of significant exceptions. In particular, because the construction only focuses on cells that intersect the bisector of each well-separated pair, the separation properties will not apply to all the points of S. We will show, however, that they do apply to the subset of S that are the nearest neighbors of some point within the cell. This follows because the bisector between any pair of such points must intersect the cell, and it is therefore easy to show the bisector of any well-separated pair that separates these points must also intersect the cell. By focusing consideration only on cells that overlap this manifold of dimension $d - 1$, we will see that the dependency on γ in the number of cells decreases from γ^d to γ^{d-1}. We let $m^-(n, d, \gamma, \beta) = n\gamma^{d-1} \log \beta$ denote the asymptotic bound on the number of nodes in the bisector sensitive case. As before, when n, d, γ, and β are clear from context, we will refer to this simply as m^-. Here are our bisector-sensitive separation properties.

LEMMA 9.3. *Consider real parameters $\beta \geq 2$ and $\gamma \geq 2$. Let S be a set of n points in $\mathbb{R}^d$, and recall that $m^- = m^-(n, d, \gamma, \beta) = n\gamma^{d-1} \log \beta$. It is possible to construct a BBD tree T with $O(m^-) = O(n\gamma^{d-1} \log \beta)$ nodes, where each leaf cell w satisfies at least one of the following three properties. Let S_w denote the subset of points of S that are the nearest neighbor of some point in w.*

(i) *$S_w \cap \gamma b_w = \emptyset$, and hence b_w is concentrically γ-separated from S_w.*
(ii) *$|S_w \cap w| = 1$ and b_w is concentrically γ-separated from $S_w \setminus w$.*
(iii) *There exists a ball b'_w such that $S_w \cap \gamma b_w \subseteq b'_w$ and b'_w is concentrically β-separated from w.*

The remainder of this section is devoted to giving the construction and proving these separation properties. In contrast to the basic construction given in Section 6,
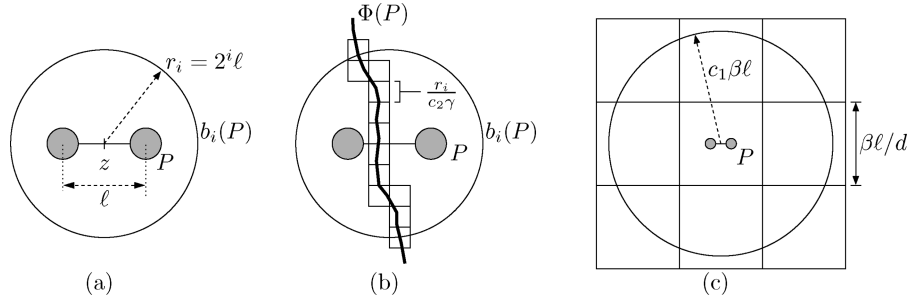
FIG. 16.   Bisector-sensitive construction.

we will generate an asymptotically smaller set of boxes by concentrating only on boxes that overlap the bisector of the well-separated pair, $\Phi(P)$. We will apply Lemma 9.2(iii) from the previous section to bound the number of boxes generated.

The bisector-sensitive construction has the same general structure as the basic construction of Section 6. It is parameterized in terms of the given quantities β and γ and constants c_1 and c_2, whose values are determined in the proof of Lemma 9.3. As in the basic construction, we will use the convenient shorthand "quadtree boxes of size s" to mean "quadtree boxes of size $2^{\lfloor \log s \rfloor}$."

We begin by computing a well-separated pair decomposition $\mathcal{P}$ for S using any constant separation factor $\sigma > 4$. For a fixed well-separated pair $P \in \mathcal{P}$, let ℓ and z denote its length and center, respectively. We compute a set of quadtree boxes $\mathcal{U}(P)$ as follows. For $0 \le i \le \lceil \log(c_1 \beta) \rceil$, let $b_i(P)$ denote the ball centered at z of radius $r_i \leftarrow 2^i \ell$ (see Figure 16(a)). Let $\mathcal{B}(P)$ denote the resulting set of balls. For each such ball $b_i(P)$, let $\mathcal{U}_i(P)$ be the set of quadtree boxes[1] of size $r_i/(c_2 \gamma)$ that overlap $b_i(P) \cap \Phi(P)$ (see Figure 16(b)). Let $\mathcal{U}^+(P)$ denote the set of quadtree boxes of size $\beta\ell/d$ that overlap the ball of radius $c_1 \beta \ell$ centered at z (see Figure 16(c)). The boxes of $\mathcal{U}^+(P)$ are not generated in a bisector-sensitive manner. They are needed for technical reasons, but their number will not dominate the asymptotic bounds. Let $\mathcal{U}(P)$ denote the union of all these boxes over all the $O(\log \beta)$ values of i together with $\mathcal{U}^+(P)$.

This process is performed for each well-separated pair of $\mathcal{P}$. Let $\mathcal{U} = \bigcup_{P \in \mathcal{P}} \mathcal{U}(P)$ denote the union of all the boxes. To complete the construction, we apply BBD property (ii) to construct a BBD tree T storing all these boxes.

This bisector-sensitive construction is essentially the same as the basic construction with two notable differences. First, only the boxes that overlap the bisector $\Phi(P)$ are generated. Second, the boxes of $\mathcal{U}^+(P)$, which are not bisector-sensitive, have been included. The following lemma bounds the size of the resulting BBD tree. It is analogous to Lemma 6.2 but with the use of $m^- = n\gamma^{d-1} \log \beta$.

---

[1] Since we will not discuss preprocessing time, we will simply assume the existence of an oracle that computes these covering boxes. It can be shown that, if we have access to an oracle that answers approximate nearest neighbor queries for the points lying within each of the dumbbell heads, it is possible to efficiently construct a slightly larger superset of the quadtree boxes that cover the bisector $\Phi(P)$. Once these covering quadtree boxes have been determined, the construction follows in a manner analogous to that of Section 6.

LEMMA 9.4. *This BBD-tree construction runs in $O(m^- \log m^-)$ time and generates $O(m^-)$ nodes.*

PROOF. Since $\mathcal{U}_i(P)$ is the set of quadtree boxes of size $r_i/(c_2\gamma)$ that overlap $b_i(P) \cap \Phi(P)$ and since the radius of $b_i(P)$ is r_i, by Lemma 9.2(iii), we have $|\mathcal{U}_i(P)| = O((1+\gamma)^{d-1})$. The size of $\mathcal{U}^+(P)$ is bounded by the number of quadtree boxes of size $\beta\ell/d$ that overlap the ball of radius $c_1\beta\ell$. By a standard packing argument, this is $O((1 + (c_1\beta\ell)/(\beta\ell/d))^d) = O(1)$. Since the number of balls for each well-separated pair is $O(\log\beta)$, the total number of boxes in $\mathcal{U}(P)$ is $O(\gamma^{d-1}\log\beta)$.

The total number of well-separated pairs is $O(n)$, and thus $|\mathcal{U}| = O(n\gamma^{d-1}\log\beta)$. Recalling our definition of m^- above, $|\mathcal{U}| = O(m^-)$. By BBD property (ii), the number of nodes of T is $O(|\mathcal{U}|) = O(m^-)$, and it can be constructed in time $O(m^- \log m^-)$. □

We now present the proof of Lemma 9.3 by showing that the BBD tree T produced by this construction satisfies the desired separation properties.

PROOF OF LEMMA 9.3. We have just shown that the BBD tree T resulting from the above construction satisfies the size and construction time bounds. It suffices to show that for all suitably large constants c_1 and c_2, T possesses the stated separation properties.

The proof follows the same structure as that of Lemma 6.1, and so we focus principally on the differences between the two. In particular, we use the prior observation that any cell generated by the old construction for a given well-separated pair P will also be generated by this construction if it overlaps $\Phi(P)$.

Let w be any leaf cell of T. Following the proof of Lemma 6.1, we consider cases based on the relationship between the points of S lying within $2\gamma b_w$, but here we focus attention not on S but on the points of S_w, which are nearest neighbors of some point in w. (In fact, this proof works for γb_w, but we continue to use $2\gamma b_w$ for the sake of symmetry.)

First, if $S_w \cap 2\gamma b_w = \emptyset$, then (i) trivially holds. Thus, for the remainder of the proof we assume that $S_w \cap 2\gamma b_w$ is nonempty, and we let x be any point of $S_w \cap 2\gamma b_w$. If x is the only point of S_w in $2\gamma b_w$, as in the proof of Lemma 6.1, either $x \in w$, implying that (ii) holds, or $x \notin w$, implying that (iii) holds for any sufficiently small ball centered at x.

Henceforth, we may assume that x is not the only point of S_w in $2\gamma b_w$. Let y be the point of $S_w \cap 2\gamma b_w$ that is farthest from x. Let $r'_w = \|xy\|$, and let b'_w be the ball of radius r'_w centered at x (see Figure 17). Clearly, any point of $S_w \cap \gamma b_w$ is contained within b'_w.

It suffices to show that b'_w satisfies separation property (iii). To this end, consider the dumbbell $P = (X, Y) \in \mathcal{P}$ that separates points x and y. Let ℓ denote the length of this dumbbell, and let z denote its center. Throughout, we will make use of the following inequalities, which follow from Lemma 3.1 and the fact that $\sigma > 4$: $\|xz\| < \ell, \ell < 2\|xy\|$, and $\ell > \|xy\|/2$.

First, we assert that the bisector $\Phi(P)$ intersects w. Since $x, y \in S_w$, they are the nearest neighbors of some points q_x and q_y, respectively, in w (see Figure 17). BBD-tree cells are connected (although they are not necessarily convex), which implies that there is a path from q_x to q_y lying entirely within w. It follows from a
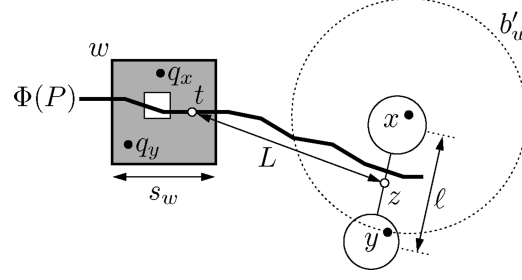
FIG. 17.   Proof of AVD separation properties for the bisector-sensitive construction.

simple continuity argument that on any such path there is at least one point t that is equidistant from X and Y, that is, $t \in w \cap \Phi(P)$. Let $L = \|zt\|$.

We distinguish cases based on the relationship between L and ℓ.

Case 1 ($L > c_1\beta\ell$). We will show that separation property (iii) holds. We first assert that $dist(z, w) > c_1\beta\ell/2$. To see this, observe that if w does not overlap the ball of radius $c_1\beta\ell$ centered at z, then clearly $dist(z, w) > c_1\beta\ell > c_1\beta\ell/2$. Otherwise, since our construction generates all quadtree boxes of size $\beta\ell/d$ that overlap this ball (from $\mathcal{U}^+(P)$), it follows that $s_w \leq \beta\ell/d$. Thus, by the triangle inequality we have

$$dist(z, w) \geq \|zt\| - diam(w) \geq L - s_w d > c_1\beta\ell - \beta\ell.$$

Choosing $c_1 \geq 2$ implies that $dist(z, w) > c_1\beta\ell/2$, as desired. Given this assertion, and arguing as in Case 1 of Lemma 6.1, we obtain

$$\begin{aligned} dist(x, w) &\geq dist(z, w) - \|xz\| > \frac{c_1\beta\ell}{2} - \ell \\ &> \left(\frac{c_1\beta}{2} - 1\right)\frac{\|xy\|}{2} = \left(\frac{c_1\beta}{4} - \frac{1}{2}\right)r'_w. \end{aligned}$$

Since $\beta \geq 2$, for all sufficiently large constants c_1 this exceeds $\beta r'_w$, which implies that b'_w is concentrically β-separated from w, as desired.

Case 2 ($L \leq c_1\beta\ell$). We adapt the argument presented in Case 2 of Lemma 6.1 to show that this case cannot occur, since otherwise the dumbbell P would have caused w to be split. Since x, y, and t are all contained in the ball $2\gamma b_w$, we have both $\|xt\| \leq 4\gamma r_w$ and $\ell < 2\|xy\| \leq 2(4\gamma r_w) = 8\gamma r_w$. Thus, by the triangle inequality, we have

$$L = \|zt\| \leq \|xz\| + \|xt\| < \ell + 4\gamma r_w < 8\gamma r_w + 4\gamma r_w = 12\gamma r_w.$$

Because $L \leq c_1\beta\ell$, it follows from our construction that there is a ball of $B(P)$ that overlaps t. Let b denote the smallest such ball, and let r denote its radius. By our construction we have $r \leq \max(\ell, 2L)$. Since our construction generates all quadtree boxes of size $r/(c_2\gamma)$ that overlap $b \cap \Phi(P)$, it follows that $s_w \leq r/(c_2\gamma)$, and so

$$r_w = s_w d \leq \frac{rd}{c_2\gamma} \leq \frac{\max(\ell, 2L)d}{c_2\gamma} < \frac{(24\gamma r_w)d}{c_2\gamma} = \frac{24 r_w d}{c_2}.$$

Choosing $c_2 \geq 24d$ yields the desired contradiction.   □

Now that we have shown that the leaf cells of T satisfy the basic separation properties, it is also useful to observe that each such cell is not too far away from its closest point of S. This fact, which is analogous to Lemma 6.3 for the basic construction, will be useful in bounding the distance to the nearest neighbor of any point of the cell.

LEMMA 9.5. *The BBD tree of Lemma 9.3 satisfies the property that there exists a constant $c > 1$ such that for any leaf cell w of this tree the ball $c\gamma b_w$ contains at least one point of S.*

PROOF. As in Lemma 6.3, let v be the quadtree box that has twice the size of w and contains w. There must be a dumbbell $P \in \mathcal{P}$ that generated a quadtree box u that is smaller and contained within v. Let b be the ball responsible for generating u in the construction, and let z and r denote the center and radius of b, respectively. (It may be helpful to recall Figure 10 from Section 6.)

Recall from our construction that b overlaps u. If $b \in \mathcal{B}(P)$, then as argued in Lemma 6.3 we have $dist(z, u) \le r \le 2c_2\gamma s_u$. Otherwise, $u \in \mathcal{U}^+(P)$. For this to occur, b has radius $r = c_1\beta\ell$ and $s_u \ge \beta\ell/(2d)$, where ℓ denotes the length of P. Since $\gamma \ge 2$ we have $dist(z, u) \le r \le c_1(2d)s_u \le c_1 d\gamma s_u$. Thus, in either case, we have $dist(z, u) \le r \le \max(2c_2, c_1 d)\gamma s_u$. Let p denote any point of S in a head of P. Arguing exactly as in Lemma 6.3, but with this new bound on r, we can easily show that the ball $c\gamma b_w$ contains p for any constant $c \ge 1 + \max(4c_2/d, 2c_1)$. □

9.3. BOUNDING TOTAL SPACE IN THE BISECTOR-SENSITIVE CASE. In this section we describe how to apply spatial amortization to bound the total number of nearest-neighbor representatives for our bisector-sensitive construction, and hence to bound the total space of the AVD. This section follows the same general structure as Section 8, but now in the context of the bisector-sensitive construction. In the analysis of the basic construction, for each cell w, we considered all the nonempty quadtree boxes of a certain size that were sufficiently close to w as being eligible for providing a representative. Here, we will be more economical and instead consider only quadtree boxes that contain a point of S that is the nearest neighbor of some query point lying within w. We shall see that if a well-separated pair separates two such points, then its bisector intersects the cell. As in the previous section, we will use the fact that the bisector is a $(d-1)$-dimensional manifold in order to produce a tighter bound on the number of representatives. We begin with a bisector-sensitive variant of Lemma 7.3.

LEMMA 9.6 (BISECTOR-SENSITIVE SPATIAL AMORTIZATION). *Let S be a set of n points in $\mathbb{R}^d$, and let $\mathcal{W}$ be the leaf cells of a BBD-tree decomposition. Let $\rho \ge 1$ and $0 < \delta \le 1$ be two real parameters. For each $w \in \mathcal{W}$, let $\mathcal{U}_w$ be the set of quadtree boxes of size $\delta\rho r_w$ that overlap ρb_w and contain at least one point of S that is the nearest neighbor of some query point in w. Then*

$$\sum_{w \in \mathcal{W}} |\mathcal{U}_w| = O\left(n\rho^{d-1}\log\frac{1}{\delta} + |\mathcal{W}|\right).$$

PROOF. The proof follows the general framework of Lemma 7.3. Recall from Section 7 that a charging scheme is a process in which each dumbbell of a WSPD allocates a unit charge to a subset of cells of $\mathcal{W}$. For each $w \in \mathcal{W}$, some subset of dumbbells are declared to be useful for w. A charging scheme is said to be valid
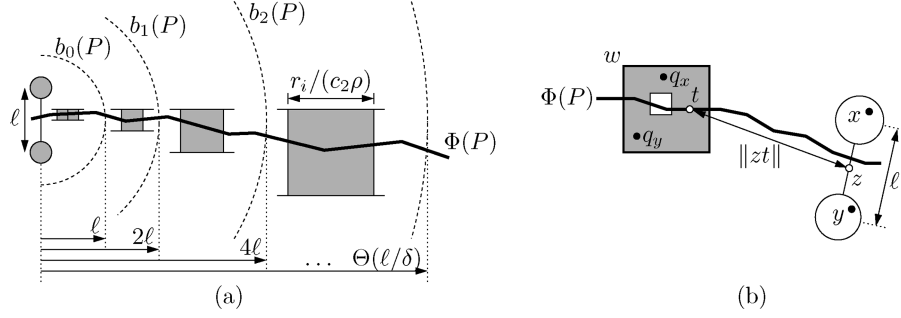
FIG. 18.   Proof of bisector-sensitive spatial amortization.

if, for any $w \in \mathcal{W}$, all w's useful dumbbells allocate a unit charge to w. We will present a charging scheme in this context and establish its validity.

Let c_1 and c_2 be two constants, both assumed to be suitably large. Let $\mathcal{P}$ be the set of dumbbells corresponding to the WSPD for S, assuming any constant separation factor greater than 4. Each dumbbell $P \in \mathcal{P}$ allocates a unit charge to some of the cells of $\mathcal{W}$ according to the following charging scheme. Let ℓ denote P's length, and let z denote its center. Let $\mathcal{B}(P)$ denote the set of balls of radii $r_i = 2^i \ell$ centered at z, for $0 \leq i \leq \lceil \log(c_1/\delta) \rceil$. Let $b_i(P)$ denote the ball of $\mathcal{B}(P)$ of radius r_i, and let $\mathcal{W}_i(P) \subseteq \mathcal{W}$ be the set of cells overlapping $b_i(P) \cap \Phi(P)$ that have size at least $r_i/(c_2 \rho)$ (see Figure 18(a)). Dumbbell P allocates a unit charge to all the cells of the set $\bigcup_{b_i(P) \in \mathcal{B}(P)} \mathcal{W}_i(P)$.

By Lemma 9.2(ii), it follows that $|\mathcal{W}_i(P)| = O((1 + r_i/(r_i/c_2\rho))^{d-1}) = O(\rho^{d-1})$. Thus, the number of cells charged by P is

$$\sum_{i=0}^{\lceil \log(c_1/\delta) \rceil} |\mathcal{W}_i(P)| = O\left(\rho^{d-1} \log \frac{1}{\delta} \right).$$

Since $|\mathcal{P}| = O(n)$, the total number of charges allocated to all the cells of $\mathcal{W}$ is $\Psi = O(n\rho^{d-1} \log(1/\delta))$. Below, we will show that this charging scheme is valid, and it will then follow from Lemma 7.2 that $\sum_w |\mathcal{U}_w| = O(n\rho^{d-1} \log(1/\delta) + |\mathcal{W}|)$, as desired.

To establish the validity of this charging scheme, let S_w be the subset of points of S that lie within some box of $\mathcal{U}_w$ and that are the nearest neighbor of some point in w. Let $s'_w = \delta \rho r_w$ be the size of the quadtree boxes of $\mathcal{U}_w$. Let $\mathcal{P}_w$ denote the subset of dumbbells that separate some pair of points $x, y \in S_w$ such that $\|xy\| \geq s'_w$. These are the useful dumbbells for w.

It suffices to show that each dumbbell $P \in \mathcal{P}_w$ charges w. By definition of S_w, each $P \in \mathcal{P}_w$ separates a pair $x, y \in S$, such that these two points are the nearest neighbors of two respective points $q_x, q_y \in w$ (see Figure 18(b)). By a simple continuity argument, $\Phi(P)$ intersects any path from q_x and q_y, and since BBD-tree cells are connected, it follows that $\Phi(P)$ intersects w. Let t be any point in $w \cap \Phi(P)$. Let ℓ denote P's length, and let z be its center. The argument given in the last two paragraphs of the proof of Lemma 7.3 shows that there is a ball $b \in \mathcal{B}(P)$ of radius r such that b overlaps t, and the size of w is at least $r/(c_2\rho)$. (The only additional element here is that t lies on $\Phi(P)$.) Clearly, w overlaps $b \cap \Phi(P)$ since
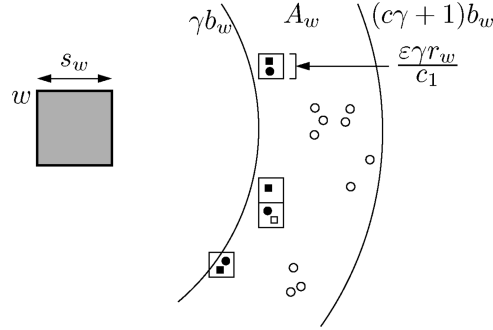
FIG. 19. Computing the outer representatives for the cell w in the bisector-sensitive case. Filled black points are the nearest neighbor of some point in w, whereas hollow points are not. Square points, either filled or hollow, are the points of O'_w.

$t \in w \cap \Phi(P)$. It follows that w receives a charge from P, which establishes the validity of the charging scheme and completes the proof. $\square$

We are now in a position to apply spatial amortization to bound the total number of nearest-neighbor representatives. To begin, we apply the bisector-sensitive construction of Lemma 9.3 to build a BBD tree T, where the parameter β has been chosen to be $16/\varepsilon$. As usual, let $\mathcal{W}$ denote the subdivision induced by the leaf cells of T. By this lemma we know that $|\mathcal{W}| = O(n\gamma^{d-1} \log(1/\varepsilon))$. As we did in this lemma, let $m^- = m^-(n, d, \gamma, \beta) = n\gamma^{d-1} \log(1/\varepsilon)$ denote this asymptotic bound. (Just as in Section 8, we may ignore the factor of 16 to simplify the expression, since m^- will only be used in asymptotic expressions.)

As before, our goal is to compute an ε-representative set for each cell $w \in \mathcal{W}$, that is, a subset $R_w \subseteq S$ such that for any query point $q \in w$, its nearest neighbor in R_w is an ε-approximation to its nearest neighbor in S. The principal lemma is given below.

LEMMA 9.7. *Let* $0 < \varepsilon \leq 1/2$ *and* $2 \leq \gamma \leq 1/\varepsilon$ *be two real parameters. Let S be a set of n points in* $\mathbb{R}^d$, *and let T be the BBD tree described above. Let* $\mathcal{W}$ *denote T's leaf cells. Then, for each cell* $w \in \mathcal{W}$, *there exists an* ε-*representative set* R_w *for w, such that* $|R_w| = O(1/(\varepsilon\gamma)^{(d-1)/2})$. *The total number of representatives over all the cells is* $O(m^-) = O(n\gamma^{d-1} \log(1/\varepsilon))$.

The remainder of this section is devoted to proving this lemma. Throughout, let ε, γ, S, n, $\mathcal{W}$, and m^- be as specified in the statement of the above lemma, and let w be any cell of $\mathcal{W}$. As mentioned in the comments following the statement of Lemma 8.1 in Section 8, there are three possible sources for the ε-nearest neighbor representatives, but the only nontrival case are the representatives lying outside the ball γb_w, called the *outer representatives* and denoted by O_w.

By Lemma 9.5, there is a constant $c > 1$ such that the ball $c\gamma b_w$ contains at least one point of S, and therefore the outer representatives may be assumed to lie within the annulus $A_w = (c\gamma + 1)b_w \setminus \gamma b_w$. For some suitably large constant c_1 (to be defined later), let $\mathcal{U}_w$ denote the set of nonempty quadtree boxes of size $\varepsilon\gamma r_w/c_1$ overlapping A_w that contain at least one point of S such that this point is the nearest neighbor of some query point in w (see Figure 19).

As in Section 8, our approach will be to construct an initial set O'_w of representatives, which will be larger than needed, and then prune this set to the desired set O_w. For each $u \in \mathcal{U}_w$, let p_u be any point of $S \cap u$. (Note that p_u need not be in S_w.) Let $O'_w = \bigcup_{u \in \mathcal{U}_w} \{p_u\}$ (see Figure 19). The boxes of $\mathcal{U}_w$ and the points p_u satisfy the conditions essential to the proof of Lemma 8.2. (In particular, they are of the same size, overlap the same annulus, and each point of $S \cap \overline{\gamma b_w}$ that is the nearest neighbor of some point in w lies within one of these boxes.) Therefore, the proof of Lemma 8.2 can be applied here as well to show that O'_w is an $(\varepsilon/4)$-outer representative set for w. We can now apply the bisector-sensitive version of spatial amortization to produce the following analog to Lemma 8.3, which bounds the total number of representatives of O'_w over all the cells of the BBD tree with respect to the tighter m^- bound.

LEMMA 9.8.   *For each cell $w \in \mathcal{W}$, let O'_w be the set of representatives defined above. Then $\sum_w |O'_w| = O(m^-)$.*

PROOF.   The proof is identical to that of Lemma 8.3 except that we invoke Lemma 9.6 in place of Lemma 7.3. □

The rest of the construction, which involves pruning the set O'_w down to O_w, follows from exactly the same construction given in Section 8, just after the proof of Lemma 8.3, but using m^- in place of m. This completes the proof of Lemma 9.7.

Summarizing, we have the following (existential) bisector-sensitive improvement to the AVD Theorem (Theorem 8.4).

THEOREM 9.9.   *Let S be a set of n points in $\mathbb{R}^d$, and let $0 < \varepsilon \le 1/2$ and $2 \le \gamma \le 1/\varepsilon$ be two real parameters. Let $m^- = n\gamma^{d-1} \log(1/\varepsilon)$. There exists a (t, ε)-AVD, where $t = O(1/(\varepsilon\gamma)^{(d-1)/2})$, of space $O(m^-)$ that can answer ε-NN queries in time $O(\log(n\gamma) + t)$.*

The space bounds of Corollary 8.5 can be improved as well. The difference is only reflected in the high-space case of $\gamma = 1/\varepsilon$, which we present below.

COROLLARY 9.10.   *Let S be a set of n points in $\mathbb{R}^d$, and let $0 < \varepsilon \le 1/2$. Let $m'' = (n/\varepsilon^{d-1}) \log(1/\varepsilon)$. There exists a $(O(1), \varepsilon)$-AVD of space $O(m'')$ that can answer ε-NN queries in time $O(\log(n/\varepsilon))$.*

By the same reasoning that we presented following Corollary 8.5, we can reduce the number of representatives per cell from $O(1)$ to 1. Owing to the additional complexities of the construction, we do not know whether it is possible to eliminate the logarithmic factor in the space bounds entirely. However, by applying the same methods mentioned at the end of Section 8 it is possible to reduce this factor in Theorem 9.9 from $\log(1/\varepsilon)$ to $\log \min(1/(\varepsilon\gamma), \gamma)$ [Arya et al. 2002]. Thus, in Corollary 9.10, the log factor can be eliminated altogether from the space bound.

10. *Lower Bounds*

In this section, we present a lower bound on the space requirements of an AVD. Our main result, presented in Theorem 10.5 in Section 10.3, provides a lower bound on the total space of any (t, ε)-AVD for a set of n sites for a given approximation bound ε and desired number of representatives t. We establish our lower bounds in a simpler and more general model, which we call *covering AVDs*, in which cells

are allowed to overlap each other and may generally be fat axis-aligned rectangles (rather than quadtree boxes). We will show that our lower bounds are nearly tight with respect to our upper bounds of Sections 8 and 9. Further, in Corollary 10.6, we will contrast the ratio between our upper and lower bounds throughout the range of space-time tradeoffs.

This section is organized as follows. In Section 10.1, we introduce the covering AVD model; in Section 10.2, we present a couple of technical results; and finally in Section 10.3, we present and analyze the lower bound construction.

10.1. COVERING AVDs. Throughout, we use the term *rectangle* to denote an axis-aligned d-dimensional hyperrectangle, that is, the Cartesian product of d closed intervals on the coordinate axes. We assume that the *side lengths* of such a rectangle, that is, the lengths of these coordinate intervals, are strictly positive. A rectangle's *aspect ratio* is the ratio between its longest and shortest side lengths. Given $\varphi \geq 1$, a rectangle is φ-*fat* if its aspect ratio is at most φ. When the value of φ is clear from context we will simply use the term *fat*.

The AVD that has been presented in our earlier constructions is a subdivision of space into BBD-tree cells, each of which is a quadtree box or the set-theoretic difference of two quadtree boxes. One of the advantages of such a subdivision is the ability to rapidly determine the leaf cell containing any query point. However, the nonconvexity of the cells resulting from inner boxes is a messy element, which we would prefer to avoid. Thus, it will be convenient to establish our lower bounds in a somewhat simpler model. Given a finite set of sites S in $\mathbb{R}^d$, a positive integer parameter $t \geq 1$, a parameter $\varepsilon > 0$, and constant $\varphi \geq 1$, we define a $(t, \varepsilon, \varphi)$-*covering AVD* to be a collection of rectangles, again called *cells*, that are φ-fat and cover the convex hull of S. Each such cell w is associated with an ε-representative set for w with respect to S, whose cardinality is at most t. (It follows that if multiple rectangles intersect, they must all store representatives for query points lying in the region of intersection.) This model is both cleaner (since inner boxes are eliminated) and more general (since we allow cells to overlap and do not demand that they be quadtree boxes). As with standard AVDs, the *space* of a covering AVD is defined to be the total number of representatives summed over all its cells. Unlike the AVDs of our upper bounds, we do not assume the existence of a structure for determining a rectangle containing the query point. Also, we require that the cells cover just the convex hull of S, whereas our upper bound AVDs cover a region whose size is larger by a factor of $\Omega(1/\varepsilon)$.

The AVDs that we have presented in our earlier BBD-tree constructions are not of this type, but we show in the lemma below that it is possible to cover any BBD-tree cell by a constant number of 3-fat rectangles. This implies that we can convert any (t, ε)-AVD into a $(t, \varepsilon, 3)$-covering AVD whose space is larger by at most a constant factor. Thus the asymptotic lower bounds on space that we will establish for covering AVDs will apply to standard AVDs as well. The proof appears in Appendix B.

LEMMA 10.1. *It is possible to cover any BBD-tree cell in $\mathbb{R}^d$ by a set of at most $2d$ closed axis-aligned rectangles, each of aspect ratio at most 3.*

For the rest of the lower bound analysis, we will consider covering AVDs involving cells that are fat rectangles. Since the aspect ratio bound φ will only affect the constant factors, we will henceforth use the terminology (t, ε)-*covering*

AVD to denote any $(t, \varepsilon, \varphi)$-covering AVD, where φ is bounded above by some constant.

10.2. TECHNICAL PRELIMINARIES. Our lower bound construction is parameterized by the number of sites n, the dimension d, the approximation factor $\varepsilon > 0$, and the maximum number t of representatives per cell. Intuitively, in order to force the space to be large we should create regions of space in which there are more than t contenders for the nearest neighbor, thus forcing us to create many AVD cells to cover this region. Consider a linear subspace K of some dimension k, and let K' denote its $(d - k)$-dimensional orthogonal complement. If we place $t + 1$ sites on K' at unit distance from the origin, then every point of K is equidistant to all these sites. Since we are limited to using t representatives per cell, we shall argue that this will force us to use many cells in order to process query points that are sufficiently close to both K and the origin. We will then use this to drive a packing argument, by showing that no cell of the AVD that is close to the origin can have a large intersection with K. Given such a configuration of sites, the final construction (which is described in Section 10.3) involves generating roughly $n/(t + 1)$ widely distributed copies of this configuration. Our construction will take the dimension k as a parameter, and later we will determine the value of k that produces the best lower bound as a function of d, ε and t.

In order to carry out this plan, it will be desirable that any cell of the AVD that has a large intersection with K contains a large ball centered on K, and hence requires many representatives to handle query points in this ball. The following lemma formalizes this observation by showing that for any k, there exists a k-dimensional linear subspace with this property. It is proved in the more general context of fat rectangles. Let $diam(U)$ denote the diameter of a given geometric set U. The proof appears in Appendix B.

LEMMA 10.2. *For any integer k and real φ, where $1 \leq k \leq d - 1$ and $\varphi \geq 1$, there exists a k-dimensional linear subspace $K \subset \mathbb{R}^d$ and a positive real $c < 1$ (depending on d and φ), such that for any φ-fat (axis-aligned) rectangle R that has a nonempty intersection with K, there is a ball of radius at least $c \cdot diam(K \cap R)$ contained in R that is centered on K.*

The next step in our plan is to show that any AVD cell that has a large diameter of intersection with K within the unit ball requires many nearest-neighbor representatives. The previous lemma implies that such a cell must contain a large ball centered on K. The following lemma shows that any sufficiently large ball that is sufficiently close to the origin requires many representatives. Given any $\beta > 0$ a set is said to be β-*sparse* if any pair of points of the set are separated by a distance of at least β.

LEMMA 10.3. *Given $0 \leq k \leq d - 1$, $0 < \varepsilon \leq 1$, and $0 < \beta \leq 2$, let K be a linear k-dimensional subspace, and let K' be its $(d - k)$-dimensional orthogonal complement. Let S be a β-sparse set of sites on K' all at unit distance from the origin. Let b be a ball centered on K whose center is within unit distance of the origin, and whose radius is at least $7\varepsilon/\beta^2$. Then, any ε-nearest neighbor representative set for b with respect to S must contain all the sites of S.*

PROOF. We first clarify that when $k = 0$, K consists of the origin, and the ball b is necessarily centered at the origin. Also, when $k = d - 1$, K' degenerates to a
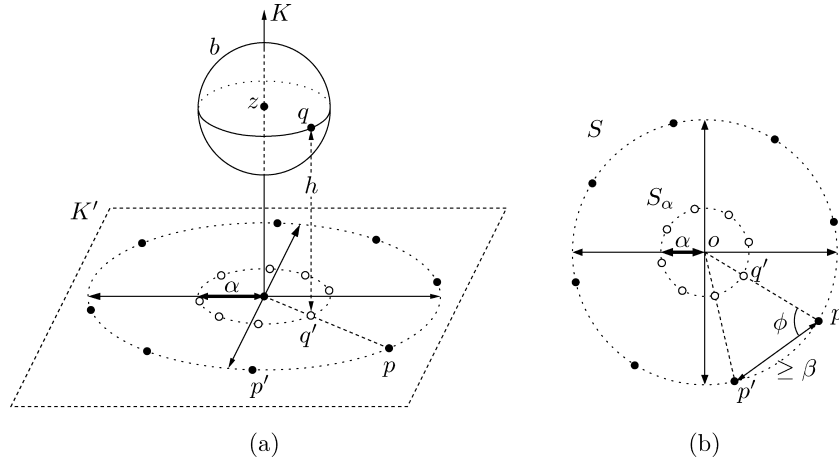
FIG. 20. Proof of Lemma 10.3.

line, and the only nontrivial β-sparse set consists of two sites, each at unit distance from the origin along K'.

To show that every point of S must be included in any ε-representative set, it suffices to show that:

$$\forall p \in S, \exists q \in b, \text{ such that } \forall p' \in S \setminus \{p\}, \ \|qp'\| > \|qp\|(1 + \varepsilon). \tag{1}$$

Fix any $p \in S$, and let p' be any point of $S \setminus \{p\}$. Let α denote b's radius, let z denote the center of b, and let $h \leq 1$ denote the distance from z to the origin (see Figure 20(a)). The distance of each point of S to z is at most $1 + h \leq 2$. If $\alpha > 2$, every point of S (and p in particular) lies within b. By setting $q = p$, Eq. (1) holds irrespective of ε. Thus, we may assume henceforth that $\alpha \leq 2$.

Let S_α be the point set resulting by scaling the coordinates of each point of S by α. Thus, the points of S_α lie on a $(d - k - 1)$-sphere of radius α centered at the origin. Let $S_\alpha + z$ denote the translate of S_α by z. Clearly, the points of $S_\alpha + z$ lie within the ball b. Given any $p \in S$, let $q' = \alpha p$ be the corresponding point of S_α and let $q = q' + z$ be the corresponding point of $S_\alpha + z$. Clearly, $\overrightarrow{qq'} \in K$ and $\overrightarrow{q'p} \in K'$, and so $\overrightarrow{qp}$ can be expressed as the sum of these orthogonal vectors. Thus,

$$\|qp\|^2 = \|qq'\|^2 + \|q'p\|^2 = h^2 + (1 - \alpha)^2 = h^2 + 1 + \alpha^2 - 2\alpha. \tag{2}$$

Because $h \leq 1$ and $\alpha \leq 2$, the above quantity is at most 2.

Recall that p' is any point of $S \setminus \{p\}$. Given vectors $\overrightarrow{u}$ and $\overrightarrow{v}$, let $\|\overrightarrow{u}\|$ denote the Euclidean length of $\overrightarrow{u}$, let $(\overrightarrow{u} \cdot \overrightarrow{v})$ denote their dot product, and let o denote the origin. By basic vector analysis (see Figures 20(a) and 20(b)) and the fact that $\|\overrightarrow{u} + \overrightarrow{v}\|^2 = \|\overrightarrow{u}\|^2 + 2(\overrightarrow{u} \cdot \overrightarrow{v}) + \|\overrightarrow{v}\|^2$, we have

$$
\begin{aligned}
\|qp'\|^2 - \|qp\|^2 &= \|\overrightarrow{qo} + \overrightarrow{op'}\|^2 - \|\overrightarrow{qo} + \overrightarrow{op}\|^2 \\
&= 2\left(\overrightarrow{qo} \cdot \left(\overrightarrow{op'} - \overrightarrow{op}\right)\right) \qquad (\text{since } \|op\|^2 = \|op'\|^2) \\
&= 2\left(\overrightarrow{qo} \cdot \overrightarrow{pp'}\right) = 2\left(\overrightarrow{q'o} \cdot \overrightarrow{pp'}\right) (\text{since } \overrightarrow{qq'} \text{ is orthogonal to } \overrightarrow{pp'}) \\
&= 2 \cdot \|q'o\| \cdot \|pp'\| \cdot \cos\phi,
\end{aligned}
$$

where ϕ is the angle between $\overrightarrow{q'o}$ and $\overrightarrow{pp'}$, which is equal to $\angle opp'$. Since $\|op\| = \|op'\| = 1$, it follows that $\cos\phi = \|pp'\|/2$. Also $\|oq'\| = \alpha$ and $\|pp'\| \geq \beta$, and so we have

$$\|qp'\|^2 - \|qp\|^2 \ = \ \frac{2 \cdot \|oq'\| \cdot \|pp'\|^2}{2} \ \geq \ \alpha\beta^2 \ \geq \ \left(\frac{7\varepsilon}{\beta^2}\right)\beta^2 \ > \ 6\varepsilon.$$

Since $\varepsilon \leq 1$ and $\|qp\|^2 \leq 2$, we have $6 \geq (2 + \varepsilon)\|qp\|^2$. This yields $\|qp'\|^2 - \|qp\|^2 > (2 + \varepsilon)\|qp\|^2\varepsilon$, which implies that

$$\|qp'\|^2 \ > \ \|qp\|^2 + \|qp\|^2(2\varepsilon + \varepsilon^2) \ = \ \|qp\|^2(1 + \varepsilon)^2.$$

By taking square roots of both sides we see that Eq. (1) holds, and this completes the proof. □

10.3. LOWER-BOUND CONSTRUCTION.   We are now ready to present the lower-bound construction. Recall that $0 < \varepsilon \leq 1$ is the approximation factor, and t is the maximum number of representatives per cell. Let $0 \leq k \leq d - 1$ be an integer parameter whose value will be fixed later. If $k = 0$, let K be the origin and let $K' = \mathbb{R}^d$. Otherwise, let K be the k-dimensional linear subspace given by Lemma 10.2, and let K' be its $(d - k)$-dimensional orthogonal complement. Consider the $(d - k - 1)$-dimensional sphere consisting of the points of K' that are at unit distance from the origin. If $0 \leq k \leq d - 2$, then for all sufficiently small β (depending on k and t) there exists a β-sparse set on this sphere consisting of $t + 1$ sites. By a straightforward packing argument such a set exists for $\beta = \Omega\left(1/t^{1/(d-k-1)}\right)$. If $k = d - 1$, K' degenerates to a line, and the only nontrivial β-sparse set consists of two sites, each at unit distance from the origin along K'. In this case, we require that $t = 1$, and observe that for $\beta = 2$ there is a β-sparse set of size $t + 1$. In either case, let S' be a set of sites consisting of such a β-sparse set.

We create $\lfloor(n - (d + 1))/(t + 1)\rfloor = \Omega(n/t)$ translated copies of S', where the translation vectors are drawn from K, and the minimum distance between any two copies is any sufficiently large constant, which we denote by L. (See Figure 21(a).) Finally, we add $(d + 1)$ points sufficiently far away so that the convex hull of these points contains a unit ball centered about each translate. (Recall that a covering AVD is required to contain the convex hull of the set of sites. These $d + 1$ points have been added to ensure that the covering AVD will contain not only the translates, but a local region around each one in order for us to apply Lemma 10.3.) Let S denote the resulting set of at most n sites. Except for the final $d + 1$ points, all of the sites of S are equidistant to any point of K. We will exploit this fact to establish a lower bound on the total space of any covering AVD for S.

LEMMA 10.4.   *Given integers k and t such that either (a) $0 \leq k \leq d - 2$ and $t \geq 1$ or (b) $k = d - 1$ and $t = 1$, let S be the point set given in the above construction. Then, for $0 < \varepsilon \leq 1$, any (t, ε)-covering AVD for S requires space*

$$\Omega\left(n\left(\frac{1}{\varepsilon\,t^{2/(d-k-1)}}\right)^k\right),$$

in case (a) and space $\Omega(n/\varepsilon^{d-1})$ in case (b).

PROOF.   Irrespective of ε, k, and t, any AVD for n sites naively requires $\Omega(n)$ space. If either $k = 0$ or $t = \Omega\left(1/\varepsilon^{(d-k-1)/2}\right)$, the stated formula provides a bound
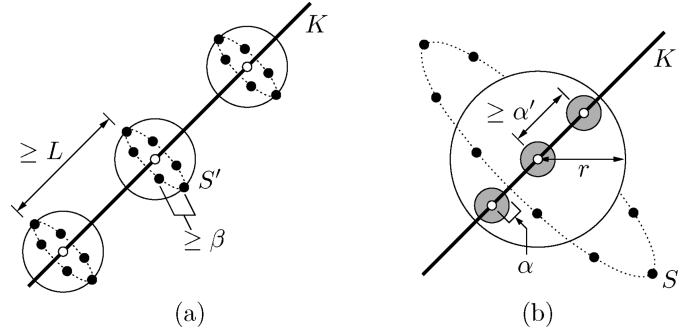
FIG. 21.   Lower bound construction (not to scale).

that is no better. Thus, we may assume that $k \geq 1$ and t is asymptotically strictly smaller than $1/\varepsilon^{(d-k-1)/2}$. Recall the constant $0 < c < 1$ from Lemma 10.2 and the sparseness parameter β in the definition of S. In addition, we define the following three quantities:

$$\alpha \ = \ \frac{7\varepsilon}{\beta^2}, \qquad \alpha' \ = \ \frac{2\alpha}{c}, \qquad \text{and} \qquad r \ = \ \frac{1}{3\sqrt{d}}.$$

Clearly, $2\alpha < \alpha'$ and $r < 1$. We may assume that $\alpha' < 2r$, since otherwise it would follow that $\beta \leq (21\varepsilon\sqrt{d}/c)^{1/2} = O(\varepsilon^{1/2})$, which in turn would imply that $t = \Omega\big(1/\varepsilon^{(d-k-1)/2}\big)$, thus leading to the trivial $\Omega(n)$ lower bound.

Our proof is based on generating a collection of sparsely placed balls, showing that no AVD cell can overlap two or more of these balls, and then bounding the space of the AVD cells that overlap each of these balls. Let Z' be a maximal α'-sparse set of points that lie within the intersection of a ball of radius r centered at the origin and the k-dimensional linear subspace K (shown as hollow points in Figure 21(b)). By a standard packing argument the number of points of Z' is $\Omega((r/\alpha')^k) = \Omega(1/\alpha^k)$. Let Z be the set that results by making translated copies of the points of Z' in exactly the same manner that was used to generate S from each copy S'. Let $\mathcal{B}_{Z'}$ be a collection of balls, each of radius α, centered at the points of Z' (shown as shaded balls in Figure 21(b)), and let $\mathcal{B}_Z$ be the union of these balls over all the translated copies. Because $2\alpha < \alpha'$ the balls of $\mathcal{B}_{Z'}$ are pairwise disjoint. Also, by making the separation distance L between translates a sufficiently large constant, the balls of any two translates are disjoint. It is easy to verify that the balls of $\mathcal{B}_{Z'}$ are contained within a unit ball centered at the origin, from which it follows that they are contained within the convex hull of S. Thus, the cells of any covering AVD of S must cover all the balls of $\mathcal{B}_Z$.

It is easy to see that for any $\varepsilon \leq 1$ and L sufficiently large, the approximate nearest neighbor of any query point lying in a ball of $\mathcal{B}_{Z'}$ must lie in the corresponding copy S'. Because the points of Z' are within distance $r \leq 1$ of the origin and of radius $\alpha = 7\epsilon/\beta^2$, by Lemma 10.3, it follows that each ball of $\mathcal{B}_{Z'}$ requires all $t+1$ points of S' to form its ε-representative set. Thus, the union of cells of the AVD that intersect any ball $b \in \mathcal{B}_Z$ must together have at least $t+1$ representatives, and so the total space for these cells is $\Omega(t)$. Assuming for now that no cell of the AVD can intersect two or more balls of $\mathcal{B}_Z$, it follows that the total space required is at
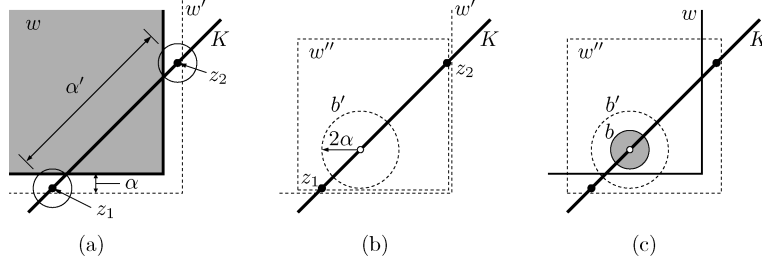
FIG. 22.   Proof of Lemma 10.4.

least

$$\Omega(t \cdot |Z|) = \Omega\!\left(t\left(\frac{n}{t}\right)|Z'|\right) = \Omega\!\left(\frac{n}{\alpha^k}\right) = \Omega\!\left(n\left(\frac{\beta^2}{\varepsilon}\right)^k\right).$$

In case (b) ($k = d - 1$), we have $\beta = 2$, and this is $\Omega(n/\varepsilon^{d-1})$. Otherwise, in case (a), ($0 \le k \le d - 2$), this is $\Omega\!\left(n(1/(\varepsilon\, t^{2/(d-k-1)}))^k\right)$, as desired.

To complete the proof, it suffices to establish the assertion that no cell of a (t, ε)-covering AVD can intersect two or more balls of $\mathcal{B}_Z$. Suppose to the contrary that there was such a cell w. Let $z_1, z_2' \in Z$ denote the centers of these balls. Irrespective of whether z_1 and z_2' arise from the same or different translated copies of Z', we have $\|z_1 z_2'\| \ge \alpha'$. Consider the line segment $\overline{z_1 z_2'}$. Let z_2 be the point along this segment at distance α' from z_1. Clearly, both z_1 and z_2 lie on K, they are separated by distance $\alpha' \le 2r$, and (since cells are convex) the balls of radius α centered at these points both intersect w (see Figure 22(a)).

Since all translated copies are identical, we may assume without loss of generality that $z_1 \in Z'$. Let w' denote the rectangle that results by expanding w by translating each of bounding facets outwards by distance α (see Figure 22(a)). It is easy to see that w' has an aspect ratio no greater than w, and it contains both z_1 and z_2. Every coordinate of the vector $\overrightarrow{z_1 z_2}$ is of absolute value at most $\alpha' \le 2r$. It follows that there exists a fat rectangle $w'' \subseteq w'$ of diameter at most $2r\sqrt{d}$ that contains both z_1 and z_2 (see Figure 22(b)). (To see this, observe that there exists a rectangle nested within w' containing z_1 and z_2 whose jth side length is the minimum of $2r$ and the corresponding side length of w'. Clearly, the aspect ratio of this rectangle is not greater than that of w', and its sides are all of length at most $2r$.) Since z_1 lies within distance r of the origin, it follows that every point of this rectangle lies within distance $r + 2r\sqrt{d} \le 3r\sqrt{d} = 1$ of the origin. By our hypothesis that $1 \le k \le d - 1$ and applying Lemma 10.2, there is a ball b' of radius $c\,\alpha' = 2\alpha$ that lies within w'' and is centered on K (see Figure 22(b)). By translating the faces of w'' inwards by α, it follows that w contains a ball b of radius $2\alpha - \alpha = \alpha$ that is centered on K at a point within unit distance of the origin (see Figure 22(c)). By Lemma 10.3, such a ball would require all $t + 1$ points of S' as representatives, contradicting the hypothesis that w is a cell of a (t, ε)-covering AVD. $\square$

Considering the case $0 \le k \le d - 2$ for now, let $L_{d,n,t,\varepsilon}(k) = n(1/(\varepsilon t^{2/(d-k-1)}))^k$ denote the asymptotic lower bound of the previous lemma as a function of k. For a given number of per-cell representatives t, we wish to determine the value of k that produces the best lower bound. Since our interest is in representatives ranging

from $1 = 1/\varepsilon^0$ up to $1/\varepsilon^{(d-1)/2}$, it will be convenient to substitute the variable t with a variable τ, where $0 \le \tau \le 1$, so that the number of representatives per cell is $t(\tau) = 1/\varepsilon^{(1-\tau)^2(d-1)/2}$. (That is, $\tau = 1 - \sqrt{-(2\ln t)/((d-1)\ln \varepsilon)}$.) With this substitution, we have

$$L_{d,n,\tau,\varepsilon}(k) = n\left(\frac{1}{\varepsilon}\right)^{k\left(1 - \frac{(d-1)(1-\tau)^2}{d-k-1}\right)}.$$

To derive the best lower bound for a fixed value of τ, we select k to maximize the exponent. Setting the derivative to zero yields $k = (d-1)\tau$. The value of τ must be chosen so that k is an integer in the range $0 \le k \le d-2$, which holds if τ is of the form $j/(d-1)$ for $0 \le j \le d-2$. Using this value of k, the exponent in the lower bound is

$$k\left(1 - \frac{(d-1)(1-\tau)^2}{d-k-1}\right) = (d-1)\tau\left(1 - \frac{(d-1)(1-\tau)^2}{(d-1)-(d-1)\tau}\right) = (d-1)\tau^2.$$

Substituting this exponent into $L_{d,n,\tau,\varepsilon}(k)$, we obtain a lower bound of

$$\Omega\left(n\left(\frac{1}{\varepsilon}\right)^{(d-1)\tau^2}\right)$$

on the space of the AVD.

On the other hand, if $k = d-1$, we have assumed that $t = 1$, and the previous lemma shows that there is a lower bound of $\Omega(n(1/\varepsilon)^{d-1})$. Note that, by setting $\tau = 1$, this arises nicely as a special case of the previous bound.

THEOREM 10.5. *Given a number of sites n, approximation bound $0 < \varepsilon \le 1$, and parameter τ of the form $j/(d-1)$, for $0 \le j \le d-1$, there exists a set of n sites S in $\mathbb{R}^d$ such that any (t, ε)-covering AVD for S requires space at least*

$$\Omega\left(n\left(\frac{1}{\varepsilon}\right)^{(d-1)\tau^2}\right), \quad \text{where} \quad t = \left(\frac{1}{\varepsilon}\right)^{\frac{(1-\tau)^2(d-1)}{2}}.$$

At one extreme, $\tau = 0$, the number of representatives per cell is $t = 1/\varepsilon^{(d-1)/2}$, and the total space is $\Omega(n)$. At the other extreme, $\tau = 1$, the number of representatives per cell is $t = 1$, and the total space is $\Omega(n/\varepsilon^{(d-1)})$. Both of these bounds match the upper bounds given in Corollaries 8.5(i) and 9.10, respectively, up to a factor of $O(\log(1/\varepsilon))$. (Recall that lower bounds on covering AVDs hold for standard AVDs as well.) Because of the quadratic dependence on τ, the lower bound differs from the upper bound between these extremes, and their ratio is bounded in the following lemma.

COROLLARY 10.6. *Given n, ε, and τ as in Theorem 10.5, the ratio between the upper bound provided in Theorem 9.9 and the lower bound of Theorem 10.5 is*

$$O\left(\left(\frac{1}{\varepsilon}\right)^{2(d-1)\tau(1-\tau)}\log\frac{1}{\varepsilon}\right).$$

PROOF. Recall that for any γ, $2 \le \gamma \le 1/\varepsilon$, the construction of Theorem 9.9 generates an AVD of space $O(n\gamma^{d-1}\log(1/\varepsilon))$ with $O(1/(\varepsilon\gamma)^{(d-1)/2})$ representatives per cell. To achieve the desired number $t = (1/\varepsilon)^{(1-\tau)^2(d-1)/2}$ representatives

per cell, we set $\gamma = c/\varepsilon^{\tau(2-\tau)}$, for a suitable constant c. (When $\tau = 0$, we have $\gamma = c$, which might violate the constraint that $\gamma \geq 2$. However, a constant factor change in γ will not affect the asymptotics.) Expressed in these terms, the upper bound on the space of the resulting AVD given by Theorem 9.9 is

$$U_d(n, \tau, \varepsilon) = O\left(n\gamma^{d-1}\log\frac{1}{\varepsilon}\right) = O\left(n\left(\frac{1}{\varepsilon}\right)^{(d-1)\tau(2-\tau)}\log\frac{1}{\varepsilon}\right).$$

Taking the ratio with our lower bound yields

$$\frac{U_d(n, \tau, \varepsilon)}{L_d(n, \tau, \varepsilon)} = O\left(\left(\frac{1}{\varepsilon}\right)^{(d-1)(\tau(2-\tau)-\tau^2)}\log\frac{1}{\varepsilon}\right) = O\left(\left(\frac{1}{\varepsilon}\right)^{2(d-1)\tau(1-\tau)}\log\frac{1}{\varepsilon}\right),$$

as desired. □

As mentioned above, this is nearly tight in the extremes. The ratio is maximized when $\tau = 1/2$, which implies that the ratio between the upper and lower bounds is $O((1/\varepsilon)^{(d-1)/2}\log(1/\varepsilon))$.

11. Conclusions

We have shown that the (t, ε)-AVD data structure provides a simple, general, and efficient solution to ε-NN searching in Euclidean spaces of constant dimension. Our formulation of the AVD structure has been in terms of a separation parameter γ, and we have seen that by increasing the degree of separation between each cell and the surrounding set of points, we can naturally induce a tradeoff between the number of cells and the number of representatives needed per cell. Stated in more traditional terms as a tradeoff between space $M_\varepsilon(n)$ and query time $T_\varepsilon(n)$, our structure achieves the tradeoff relationship $M_\varepsilon(n)T_\varepsilon^2(n) = O(n/\varepsilon^{d-1})$ (up to log factors). In terms of the relationship between asymptotic space and query time, our structure performs better (and sometimes significantly better) than existing data structures for ε-NN searching.

We have also presented lower bounds showing that these bounds are essentially tight in the extremes, assuming an AVD-like approach. If one considers the entire continuum, there is a gap of roughly $(1/\varepsilon)^{(d-1)/2}$ between our lower bounds and upper bounds. The most obvious open problem remaining is to close this gap.

In order to analyze the space complexity of the AVD, we have introduced a technique called spatial amortization. A (t, ε)-AVD allows t representatives per cell, and thus a naive analysis would suggest that the total space would be larger by a factor of t. However, spatial amortization shows that the upper bounds on total space and numbers of cells differ only by a constant. This insight was used to establish the space efficiency of our AVD constructions. We believe that this technique may find applications elsewhere, and it has already proved to be useful in the design of AVDs for range searching [Arya et al. 2005, 2006, 2009].

Although the AVD dramatically outperforms partition tree-based methods (such as the BBD tree [Arya et al. 1998], the BAR tree [Duncan et al. 2001], and the linearized quadtree [Chan 2002, 2006]) in terms of query times, methods based on partition trees still have a number of advantages. For one, partition-tree space and constructions are independent of ε, and hence ε need only be provided at query

time. Also, depending on the variant, partition trees also have much lower constant factors in space (not growing exponentially with dimension as for the AVD) and are amenable to efficient point insertion and deletion. Chan's in-place quadtree structure [Chan 2006] and the skip quadtree [Eppstein et al. 2008] are notable examples.

Viewed generically, an AVD is a decomposition of space into regions, each of which is sufficiently well separated from its surroundings so that all the information needed for answering queries can be encoded concisely, typically in $O(1)$ space or as a flat data structure such as an array. (This representation implies that query processing is very simple, consisting of a tree descent followed by a small number of distance computations. However, it is interesting to consider whether the space or query time could be improved through the use of a more sophisticated storage scheme for the leaf data.) The AVD concept was perhaps first used by Vleugels and Overmars [1998] as a means to approximate the Voronoi diagram of a set of convex obstacles in the context of motion planning. The use of AVDs in approximate nearest neighbor searching has been established here and in earlier work [Har-Peled 2001; Sabharwal et al. 2006]. AVDs have also been adapted to achieve many of the best known space and query times for approximate range searching in spaces of constant dimension [Arya et al. 2005, 2006, 2009]. In the same way that the locus method can be used for solving exact geometric retrieval problems through point location, we believe that AVDs will continue to find new applications in approximate geometric retrieval problems. We hope that this article will provide the fundamental groundwork upon which these future structures will be based.

Appendix

A. *Proofs of Separation Theorems*

In this section, we provide proofs of our theorems involving separation and numbers of representatives, which were stated in Section 5.

We begin with a couple of technical lemmas regarding triangles. The first lemma is an elegant result, due to Chan and Snoeyink. Given an obtuse triangle, it bounds the sum of lengths of the short edges in terms of the length of the long edge. We provide a proof for the sake of completeness.

LEMMA A.1 (CHAN AND SNOEYINK [1995]). *Let* $\triangle xyz$ *be a triangle with* $\angle xzy = \theta$, $\angle yxz = \phi$, *and* $\angle xyz \geq \pi/2$. *Then*

$$\|xy\| + \|yz\| \leq (1 + \sin\theta \sin\phi)\|xz\|.$$

PROOF. By the Law of Sines,

$$\frac{\|xy\|}{\sin\theta} = \frac{\|yz\|}{\sin\phi} = \frac{\|xz\|}{\sin(\theta+\phi)}.$$

This implies that

$$\frac{\|xy\| + \|yz\|}{\|xz\|} - 1 = \frac{\sin\theta + \sin\phi - \sin(\theta + \phi)}{\sin(\theta + \phi)}$$

$$= \sin\theta \, \sin\phi \left(\frac{\tan(\phi/2) + \tan(\theta/2)}{\sin(\theta + \phi)} \right).$$

Since $\theta + \phi \leq \pi/2$, we have

$$\tan\frac{\theta}{2} + \tan\frac{\phi}{2} \leq \frac{2}{\pi}(\theta + \phi) \leq \sin(\theta + \phi),$$

and the proposition follows directly. □

LEMMA A.2. *Given a triangle $\triangle xyz$:*

(i) $\max(\|xy\|, \|xz\|) \geq \|yz\|$ *implies that $\angle yxz \leq \pi/2$.*
(ii) $\sin(\angle yxz) \leq \|yz\|/\max(\|xy\|, \|xz\|)$.

PROOF. Part (i) follows from the fact that x must lie outside the circle with $\|yz\|$ as diameter.

For part (ii) let $\theta = \angle yxz$. It follows from the law of sines that $\sin\theta/\|yz\| \leq 1/\|xy\|$ and $\sin\theta/\|yz\| \leq 1/\|xz\|$. Thus, $\sin\theta \leq \|yz\|/\max(\|xy\|, \|xz\|)$. □

Using the above results, we now restate and prove Lemma 5.1.

LEMMA 5.1 (CONCENTRIC BALL LEMMA). *Let $c > 0$ be a constant, and let $0 < \varepsilon \leq 1$ and $\gamma \geq 1 + c$ be two real parameters. Let S be a set of points in $\mathbb{R}^d$. Let b_1 and b_2 be two balls in $\mathbb{R}^d$ such that b_1 is concentrically γ-separated from $\overline{b}_2$. Then there exist subsets $R_1, R_2 \subseteq S$ each consisting of at most*

$$O\left(\left(1 + \frac{1}{\sqrt{\varepsilon\gamma}}\right)^{d-1}\right)$$

points such that

(i) *R_1 is an ε-representative set for b_1 with respect to $S \cap \overline{b}_2$, and*
(ii) *R_2 is an ε-representative set for $\overline{b}_2$ with respect to $S \cap b_1$.*

PROOF. We only prove part (i), since the proof of part (ii) is entirely symmetrical. Clearly, if the lemma holds when c equals some positive constant, it also holds if we set c to a larger constant. Thus, it suffices to prove the lemma for $0 < c \leq 1$. First, if $\gamma \geq 16/\varepsilon$, then the lemma follows from Lemma 5.2, and so we may assume that $\gamma < 16/\varepsilon$. Let b_3 be the ball $(1 + (c/2))b_1$. Let $\delta = cr\sqrt{\varepsilon\gamma}/16$, where r is the radius of b_1. Let R' be a set of points on the boundary of b_3 that is δ-dense for the boundary. (That is, for any point y on the boundary of b_3, there is a point $x \in R'$ such that $\|yx\| \leq \delta$.) By standard results [Yao 1982], we can find such a set R' of size $O(1/(\varepsilon\gamma)^{(d-1)/2})$. For each point $x \in R'$, we let p_x denote any point of S that is its $(\varepsilon/2)$-NN. We define $R = \{p_x : x \in R'\}$. We now show that R satisfies the property given in part (i) of the lemma.

Let q be a point in b_1. Let p_q denote the nearest neighbor of q among the points of $S \cap \overline{b}_2$. Let y denote the point of intersection of $\overline{qp_q}$ with the boundary of b_3.
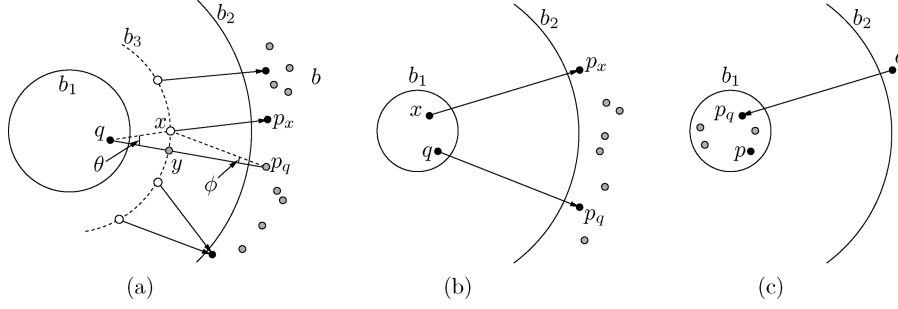
FIG. 23.   Proofs of Lemmas 5.1 and 5.2.

Let x be the point of R' that is closest to y (see Figure 23(a)). We will show that $\|qp_x\| \leq (1+\varepsilon)\|qp_q\|$, which will imply (i).

By the triangle inequality, we have $\|qp_x\| \leq \|qx\| + \|xp_x\|$. Since p_x is an $(\varepsilon/2)$-NN of x, we have $\|xp_x\| \leq (1+\varepsilon/2)\|p_qx\|$. Thus,

$$\|qp_x\| \;\leq\; \left(1+\frac{\varepsilon}{2}\right)(\|qx\| + \|p_qx\|). \tag{3}$$

In the triangle $\triangle qp_qx$, let θ denote $\angle p_qqx$ and ϕ denote $\angle qp_qx$. We will make use of the following, which will be proved below:

   (a): $\angle qxp_q > \pi/2,$ (b): $\sin\theta \leq \dfrac{\sqrt{\varepsilon\gamma}}{8},$ (c): $\sin\phi \leq \dfrac{1}{4}\sqrt{\dfrac{\varepsilon}{\gamma}}.$

Assuming these hold for now and applying Lemma A.1, we obtain $\|qx\|+\|p_qx\| \leq (1+\varepsilon/32)\|qp_q\|$. Substituting this in Eq. (3), and noting that $\varepsilon \leq 1$, we obtain

$$\|qp_x\| \;\leq\; \left(1+\frac{\varepsilon}{2}\right)\left(1+\frac{\varepsilon}{32}\right)\|qp_q\| \;\leq\; (1+\varepsilon)\|qp_q\|,$$

which is the desired result.

To prove inequalities (a)–(c) above consider $\triangle yqx$. Since R' is δ-dense for the boundary of b_3, we have $\|xy\| \leq \delta = cr\sqrt{\varepsilon\gamma}/16$. Also, $\max(\|qx\|, \|qy\|) \geq cr/2$. Thus $\|xy\|/\max(\|qx\|, \|qy\|) \leq \sqrt{\varepsilon\gamma}/8$. By Lemma A.2(ii), it follows that $\sin\theta \leq \sqrt{\varepsilon\gamma}/8$, which establishes (b). Now, using Lemma A.2(i) and the fact that $\gamma < 16/\varepsilon$, it is easy to see that $\theta \leq \pi/6$. We next consider $\triangle yp_qx$. Since $\gamma \geq 1+c$ and $c \leq 1$ we see that

$$\max(\|p_qx\|, \|p_qy\|) \geq \gamma r - \left(1+\frac{c}{2}\right)r \geq \gamma r - \frac{\left(1+\frac{c}{2}\right)}{1+c}\gamma r \geq \frac{c/2}{1+c}\gamma r \geq \frac{c\gamma r}{4}.$$

Thus $\|xy\|/\max(\|p_qx\|, \|p_qy\|) \leq \frac{1}{4}\sqrt{\varepsilon/\gamma}$. By Lemma A.2(ii), $\sin\phi \leq \frac{1}{4}\sqrt{\varepsilon/\gamma}$, which establishes (c). Now, applying Lemma A.2(i) and noting that $\varepsilon \leq 1$ and $\gamma \geq 1+c > 1$, it follows that $\phi \leq \pi/6$. Finally, since $\theta \leq \pi/6$ and $\phi \leq \pi/6$, we have $\angle qxp_q > \pi/2$, which establishes (a). $\square$

Finally, we restate and prove Lemma 5.2.

LEMMA 5.2 (SINGLE-REPRESENTATIVE CONC. BALL LEMMA).   *Let $0 < \varepsilon \leq 1$ and $\gamma \geq 11/\varepsilon$ be two real parameters. Let S be a set of points in $\mathbb{R}^d$. Let b_1 and b_2 be two balls in $\mathbb{R}^d$ such that b_1 is concentrically γ-separated from $\overline{b}_2$.*

(i) *Let p_x be an $(\varepsilon/2)$-nearest neighbor of any point $x \in b_1$. Then $\{p_x\}$ is an ε-representative set for b_1 with respect to $S \cap \overline{b}_2$.*

(ii) *Let p be any point of $S \cap b_1$. Then $\{p\}$ is an ε-representative set for $\overline{b}_2$ with respect to $S \cap b_1$.*

PROOF. We first prove (i). It suffices to show that, for any point $q \in b_1$, p_x is an ε-NN of q with respect to $S \cap \overline{b}_2$. Let p_q denote the nearest neighbor of q among the points of $S \cap \overline{b}_2$ (see Figure 23(b)). By the triangle inequality, we have $\|qp_x\| \leq \|qx\| + \|xp_x\|$. Since p_x is an $(\varepsilon/2)$-NN of x, it follows that $\|xp_x\| \leq (1 + \varepsilon/2)\|xp_q\|$. Therefore,

$$\|qp_x\| \leq \|qx\| + \left(1 + \frac{\varepsilon}{2}\right)\|xp_q\|.$$

Applying the triangle inequality again, we obtain $\|xp_q\| \leq \|qx\| + \|qp_q\|$. Using the fact that $\varepsilon \leq 1$, we see that

$$\|qp_x\| \leq \|qx\| + \left(1 + \frac{\varepsilon}{2}\right)(\|qx\| + \|qp_q\|) = \left(2 + \frac{\varepsilon}{2}\right)\|qx\| + \left(1 + \frac{\varepsilon}{2}\right)\|qp_q\|$$

$$\leq \frac{5}{2}\|qx\| + \left(1 + \frac{\varepsilon}{2}\right)\|qp_q\|. \tag{4}$$

Letting r denote the radius of ball b_1, we have $\|qx\| \leq 2r$. Since $\gamma \geq 11/\varepsilon$ and $\varepsilon \leq 1$ it follows that

$$\|qp_q\| \geq (\gamma - 1)r \geq \left(\frac{11}{\varepsilon} - \frac{1}{\varepsilon}\right)r \geq \frac{10}{\varepsilon}r. \tag{5}$$

Therefore, $\|qx\| \leq (\varepsilon/5)\|qp_q\|$. Substituting this into Eq. (4), we obtain $\|qp_x\| \leq (1 + \varepsilon)\|qp_q\|$, as desired.

Next we prove (ii). It suffices to show that, for any point $q \in \overline{b}_2$, p is an ε-NN of q with respect to $S \cap b_1$. Let p_q denote the nearest neighbor of q among the points of $S \cap b_1$ (see Figure 23(c)). By the triangle inequality, we have $\|qp\| \leq \|qp_q\| + \|pp_q\|$. Let r denote the radius of the ball b_1. We have $\|pp_q\| \leq 2r$, and since $\gamma \geq 11/\varepsilon$ and $\varepsilon \leq 1$, as in Eq. (5), we have $\|qp_q\| \geq (10/\varepsilon)r$. Therefore, $\|pp_q\| \leq (\varepsilon/5)\|qp_q\|$. Putting this all together, we have

$$\|qp\| \leq \|qp_q\| + \|pp_q\| \leq \|qp_q\| + \frac{\varepsilon}{5}\|qp_q\| < (1 + \varepsilon)\|qp_q\|,$$

as desired. □

B. *Proofs of Technical Lemmas on Fat Rectangles*

First, we restate and prove Lemma 10.1.

LEMMA 10.1. *It is possible to cover any BBD-tree cell in $\mathbb{R}^d$ by a set of at most $2d$ closed axis-aligned rectangles, each of aspect ratio at most 3.*

PROOF. Let w be the BBD-tree cell in question. If w has no inner box, then the theorem is trivially satisfied by taking w itself as the covering rectangle (of aspect ratio 1). Otherwise, let u_i and u_o denote w's inner and outer boxes, respectively (see Figure 24(a)). Let s_i and s_o denote their respective side lengths. Let $\mathcal{F}$ denote
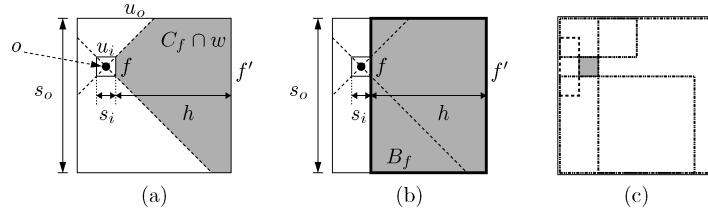
FIG. 24. Proof of Lemma 10.1, with the final four covering rectangles shown in part (c).

set of the up to $2d$ facets (faces of dimension $d - 1$) of u_i that do not intersect a facet of u_o.

We begin by subdividing w into a collection of at most $2d$ polygonal regions. For each facet $f \in \mathcal{F}$, consider the infinite convex polygonal cone C_f formed by the union of rays emanating from u_i's center point and passing through all points of f. Let B_f denote the smallest axis-aligned rectangle enclosing $C_f \cap w$ (see Figure 24(b)). Clearly the cones cover space, and thus the associated enclosing rectangles B_f form a cover of w of cardinality at most $2d$.

To complete the proof, we show that for any $f \in \mathcal{F}$, B_f has aspect ratio at most 3. Let f' denote the facet of u_o corresponding to f, that is, the one sharing the same outward normal vector, and let $h = dist(f, f')$ (see Figure 24(a)). Since $f \in \mathcal{F}$, we have $h > 0$, and by basic properties of quadtree alignment, $h \geq s_i$. Since the slopes of the sides of C_f form an angle of $45°$ with respect to its central axis, the intersection of C_f with the $(d - 1)$-dimensional hyperplane containing f' is a $(d - 1)$-dimensional hypercube, denoted B', of side length $2(h + s_i/2) = 2h + s_i$. It is easy to see that the side lengths of B' are at least $h + s_i$ (in Figure 24(a) consider when u_i touches the top edge of u_o) and at most $2h + s_i$. Since the side of B_f that is orthogonal to f is of length h, B_f's aspect ratio is at most

$$\max\left(\frac{\max(h, 2h + s_i)}{\min(h, h + s_i)}\right) \leq \max\left(\frac{3h}{h}\right) \leq 3,$$

as desired. □

Next, we restate and prove Lemma 10.2. This lemma asserts that, given any $1 \leq k \leq d$, there exists a linear subspace K of any dimension k such that any fat rectangle that has a large intersection with K contains a large ball centered on K.

LEMMA 10.2. *For any integer k and real φ, where $1 \leq k \leq d - 1$ and $\varphi \geq 1$, there exists a k-dimensional linear subspace $K \subset \mathbb{R}^d$ and a positive real $c < 1$ (depending on d and φ), such that for any φ-fat (axis-aligned) rectangle R that has a nonempty intersection with K, there is a ball of radius at least $c \cdot diam(K \cap R)$ contained in R that is centered on K.*

Intuitively, a subspace may fail to achieve this property because it "scrapes" along near the boundary of the rectangle for a long distance. Our choice of K is motivated by the property that no vector of K is too nearly orthogonal with respect to any k-dimensional axis-aligned hyperplane. This will imply that the linear subspace is in some sense well "slanted" with respect to the coordinate hyperplanes. Making this intuition formal will involve some matrix analysis.

As with many constructions in computational geometry, our construction is based on the moment curve. Consider the $(d - k) \times d$ matrix $U = [u_{i,j}]$, where $u_{i,j} = i^j$.

Let K be the null space of U, that is, $K = \{x \in \mathbb{R}^d : Ux = 0\}$. The rows of U are points on the d-dimensional *moment curve* [Edelsbrunner 1987], and it is well known that they are linearly independent. Thus the dimension of K is k.

Let us first explore some properties of U. Given a pair of increasing sequences $0 < a_1 < a_2 < \cdots < a_m$ and $0 < b_1 < b_2 < \cdots < b_m$, the $m \times m$ matrix $V = [v_{i,j}]$, where $v_{i,j} = a_i^{b_j}$, is called a *generalized Vandermonde matrix* [Gantmacher 1959]. It is known that the determinant of any generalized Vandermonde matrix is strictly positive [Gantmacher 1959]. Clearly, any $(d - k) \times (d - k)$ submatrix of U is of this form, and so the corresponding minor of U is nonzero. In order to define such a submatrix, consider any k-element integer sequence $I : 1 \leq j_1 < j_2 < \cdots < j_k \leq d$, and let U' be the $(d - k) \times k$ submatrix of U formed by concatenating the columns of U corresponding to the indices of I. Let U'' be the $(d - k) \times (d - k)$ submatrix formed by concatenating the remaining columns of U. In an analogous way, we can decompose any vector $x \in \mathbb{R}^d$ into $x' \in \mathbb{R}^k$ and $x'' \in \mathbb{R}^{d-k}$. Clearly, $x \in K$ if and only if $U'x' + U''x'' = 0$. When I is understood from context, we express these decompositions as $U = (U'; U'')$ and $x = (x'; x'')$.

Any k-dimensional axis-aligned hyperplane is, up to translation, the span of k coordinate unit vectors, and so can be identified with some choice of I. In order to show that a vector $x \in K$ is not too nearly orthogonal to such a coordinate hyperplane it suffices to show that the length of x'' (which is orthogonal to the hyperplane) is not significantly greater than the length of x' (which is parallel to the hyperplane). We prove a slightly stronger version of this next. Let $\|\cdot\|_\infty$ and $\|\cdot\|_2$ denote the L_∞ and L_2 norms, respectively, for both vectors and matrices [Golub and Loan 1996].

LEMMA B.1. *Let $1 \leq k \leq d$, and consider the $(d - k) \times d$ matrix U defined above and any $x \in \mathbb{R}^d$. For any k-element index sequence I, let $U = (U'; U'')$ and $x = (x'; x'')$ denote the corresponding decompositions. Given a fixed $x' \in \mathbb{R}^k$, there is a unique $x'' \in \mathbb{R}^{d-k}$ such that $(x'; x'') \in K$. Furthermore, there exists a constant $c' > 1$ (depending on dimension) such that $\|x''\|_\infty \leq c'\|x'\|_\infty$.*

PROOF. It follows from our earlier remarks that $\det(U'') \neq 0$, and so its inverse, $(U'')^{-1}$, exists and is unique. If $x \in K$, then $0 = Ux = U'x' + U''x''$. Therefore x'' is uniquely determined by $x'' = -(U'')^{-1}U'x'$. Each entry of $(U'')^{-1}$, as computed by Cramer's rule, is the ratio of nonzero determinants involving integers of magnitude at most d^d and hence are themselves of magnitude at most $(d^d)^{O(d)} \leq d^{O(d^2)}$. Thus, $\|(U'')^{-1}\|_\infty = d^{O(d^2)}$. Since the entries of U' are of magnitude at most d^d, it follows that $\|U'x'\|_\infty \leq d^{O(d)}\|x'\|_\infty$. Thus, by standard linear algebra (see, e.g., Golub and Loan [1996]) we have

$$\|x''\|_\infty \leq \|(U'')^{-1}\|_\infty \|U'x'\|_\infty \leq d^{O(d^2)} \cdot d^{O(d)}\|x'\|_\infty \leq c'\|x'\|_\infty,$$

where $c' = d^{O(d^2)}$. □

We are now in a position to prove Lemma 10.2. Our approach is to show that if $diam(K \cap R)$ is large for some fat rectangle R, then given any point $x \in K \cap R$, we can slide along K in a direction such that the distance from the boundary of R increases rapidly. To determine this direction, we first classify which faces of R are close to x. We decompose R into a collection of subrectangles according to which face of R is close. The direction is a function of the subrectangle that contains x.
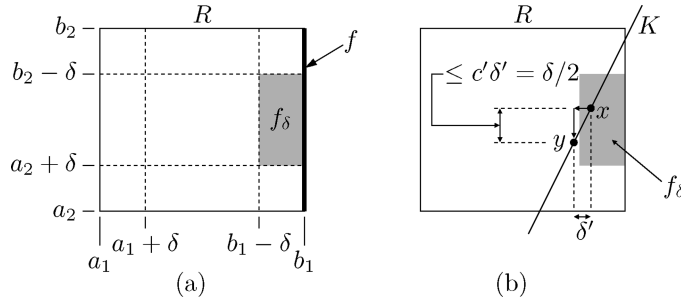
FIG. 25. The subrectangles Lemma 10.2. For example, for face f we have $\pi = (+1, 0)$, and so $f = \{b_1\} \times [a_2, b_2]$ and $f_\delta = [b_1 - \delta, b_1] \times [a_2 + \delta, b_2 - \delta]$. (Since the figure is drawn in the plane, we have violated the constraint that $j < k$ in (b). Both are equal to 1 here.)

PROOF OF LEMMA 10.2. Let $R = \prod_{1 \le i \le d}[a_i, b_i]$, and let $\Delta = diam(K \cap R)$. Let $\mathcal{F}$ denote the set of faces (of all dimensions) of R. Each face $f \in \mathcal{F}$ can be represented by a d-vector $\pi \in \{-1, 0, +1\}^d$, so that $f = \prod_{1 \le i \le d} \Psi_i$, where

$$\Psi_i = \begin{cases} \{a_i\} & \text{if } \pi_i = -1 \\ [a_i, b_i] & \text{if } \pi_i = 0 \\ \{b_i\} & \text{if } \pi_i = +1. \end{cases}$$

Let us next decompose R into a collection of subrectangles by "fattening" each face of R into a rectangle as follows. Recalling that R is φ-fat, we define $\delta = \Delta/(c''\varphi)$, where c'' is a sufficiently large constant (to be specified below). Given any face $f \in \mathcal{F}$, let f_δ be the rectangle $f = \prod_{1 \le i \le d} \Psi_i(\delta)$, where

$$\Psi_i(\delta) = \begin{cases} [a_i, a_i + \delta] & \text{if } \pi_i = -1 \\ [a_i + \delta, b_i - \delta] & \text{if } \pi_i = 0 \\ [b_i - \delta, b_i] & \text{if } \pi_i = +1 \end{cases}$$

(see Figure 25(a)).

By convexity of $K \cap R$, Δ is the length of some line segment contained within R. It follows that the longest side of R is of length at least $\Delta/\sqrt{d}$. Since R is φ-fat, the shortest side of R is of length at least $\Delta/(\varphi\sqrt{d})$. By selecting $c'' > 2\sqrt{d}$, we see that δ is less than one half the length of the shortest side of R. Thus, the rectangles f_δ defined above are all contained within R, they cover R, and have pairwise disjoint interiors. In addition, for any $x, y \in f_\delta$ we see that coordinates similar along skinny dimensions and we are far from the boundary of R along fat dimensions. That is,

$$\pi_i \in \{-1, +1\} \Rightarrow |x_i - y_i| \le \delta \tag{6}$$
$$\pi_i = 0 \Rightarrow \min(|x_i - a_i|, |b_i - x_i|) \ge \delta \tag{7}$$

As mentioned above, there is a line segment contained within R whose length is Δ. Because the fattened faces cover R, it follows that there is a face $f \in \mathcal{F}$ such that $diam(K \cap f_\delta) \ge \Delta/|\mathcal{F}| = \Delta/3^d$. Let π denote the vector that defines this face f, and let j denote the number of nonzero entries of π. It will simplify the presentation to assume that the axes have been relabeled so that the first j entries of π are nonzero, and hence the last $d - j$ are zero. Henceforth, for any $x \in \mathbb{R}^d$, let $(x'; x'')$ denote the decomposition induced by the first k and last $d - k$ components of x.

We begin by showing that $j < k$. Suppose to the contrary that $j \geq k$. For $i \leq j$, we have $\pi_i \in \{-1, +1\}$. Consider any pair of points $x, y \in K \cap f_\delta$. By Eq. (6) it follows that $|x_i - y_i| \leq \delta$. Since $k \leq j$, this is true for all $i \leq k$, and therefore $\|x' - y'\|_\infty \leq \delta$. Since $x, y \in K$ we have $x - y \in K$, and so by Lemma B.1, we have

$$\|x'' - y''\|_\infty \leq c'\|x' - y'\|_\infty \leq c'\delta,$$

for $c' > 1$. Thus, $\|x - y\|_\infty \leq c'\delta$, and so $\|x - y\|_2 \leq c'\delta\sqrt{d}$. By setting $c'' > c'3^d\sqrt{d}$ in the definition of δ and using the fact that $\varphi \geq 1$, we see that this is less than $\Delta/3^d$. Since x and y are arbitrary, this implies that $diam(K \cap f_\delta) \leq 3^d\sqrt{d}$, which contradicts the above bound on $diam(K \cap f_\delta)$.

Given that $j < k$, we will show that there exists a point $y \in K \cap R$ (but not necessarily in f_δ) whose distance from each of the sides of R is at least $\delta' = \delta/(2c')$, where $c' > 1$ is the constant of Lemma B.1 (see Figure 25(b)). Let x be any point of $K \cap f_\delta$. First, for $i \leq j$, we have $\pi_i \in \{-1, +1\}$. If $\pi_i = -1$, let $y_i = x_i + \delta'$, and if $\pi_i = +1$, let $y_i = x_i - \delta'$. The distance from y to the closest bounding side of R along dimension i is at least $\min(\delta', \delta - \delta') \geq \delta'$. Next, for i in the range from $j + 1$ to k, we let $y_i = x_i$. For these points, $\pi_i = 0$, and so by Eq. (7) it follows that each of these coordinates is at distance at least $\delta \geq \delta'$ from the closest bounding side of R.

Having defined all the components of y', we apply Lemma B.1 to conclude that there is a unique y'' so that $y = (y'; y'') \in K$. Furthermore, since both $x, y \in K$, we have $x - y \in K$, and so $\|x'' - y''\|_\infty \leq c'\|x' - y'\|_\infty$. Clearly, $\|x' - y'\|_\infty \leq \delta'$, and so $\|x'' - y''\|_\infty \leq c'\delta' = \delta/2$. For all the components of x'', we have $\pi_i = 0$, and so from Eq. (7) it follows that the distance from x_i to the closest bounding side of R is at least δ. Therefore, the distance from y_i to the closest bounding side of R is at least $\delta - (\delta/2) \geq \delta'$. The point y is the desired point in $K \cap R$. The ball of radius δ' centered at y lies entirely within R, and by our definition of δ it has radius

$$\delta' = \frac{\delta}{2c'} = \frac{\Delta}{2c'c''\varphi}.$$

Let $c = 1/(2c'c''\varphi)$. Since c', c'' and φ are all at least 1, we have $c < 1$, as desired. $\square$

REFERENCES

ARYA, S., DA FONSECA, G. D., AND MOUNT, D. M. 2008a. Tradeoffs in approximate range searching made simpler. In *SIBGRAPI '08: Proceedings of the 2008 XXI Brazilian Symposium on Computer Graphics and Image Processing*. IEEE Computer Society, Los Alamitos, CA, 237–244.

ARYA, S., AND MALAMATOS, T. 2002. Linear-size approximate Voronoi diagrams. In *Proceedings of the 13th Annual ACM-SIAM Symposium on Discrete Algorithms*. ACM, New York, 147–155.

ARYA, S., MALAMATOS, T., AND MOUNT, D. M. 2002. Space-efficient approximate Voronoi diagrams. In *Proceedings of the 34th Annual ACM Symposium on Theory of Computing*. ACM, New York, 721–730.

ARYA, S., MALAMATOS, T., AND MOUNT, D. M. 2005. Space-time tradeoffs for approximate spherical range counting. In *Proceedings of the 16th Annual ACM-SIAM Symposium on Discrete Algorithms*. ACM, New York, 535–544.

ARYA, S., MALAMATOS, T., AND MOUNT, D. M. 2006. On the importance of idempotence. In *Proceedings of the 38th Annual ACM Symposium on Theory Computing*. ACM, New York, 564–573.

ARYA, S., MALAMATOS, T., AND MOUNT, D. M. 2009. The effect of corners on the complexity of approximate range searching. *Disc. Comput. Geom. 41*, 398–443.

ARYA, S., AND MOUNT, D. M. 2000. Approximate range searching. *Comput. Geom. Theory Appl. 17*, 135–152.

ARYA, S., MOUNT, D. M., NETANYAHU, N., SILVERMAN, R., AND WU, A. Y. 1998. An optimal algorithm for approximate nearest neighbor searching in fixed dimensions. *J. ACM 45*, 891–923.

ARYA, S., MOUNT, D. M., VIGNERON, A., AND XIA, J. 2008b. Space-time tradeoffs for proximity searching in doubling spaces. In *Proceedings of the 16th Annual European Symposium on Algorithms*. Lecture Notes Computer Science, vol. LNCS 5193/2008. Springer-Verlag, Berlin, Germany, 112–123.

BENT, S. W., SLEATOR, D. D., AND TARJAN, R. E. 1985. Biased search trees. *SIAM J. Comput. 14*, 545–568.

BESPAMYATNIKH, S. N. 1996. Dynamic algorithms for approximate neighbor searching. In *Proceedings of the 8th Canadian Conference on Computer Geometry*. 252–257.

BEYER, K., GOLDSTEIN, J., RAMAKRISHNAN, R., AND SHAFT, U. 1999. When is "nearest neighbor" meaningful. In *Proceedings of the International Conference on Database Theory*. 217–235.

BRONSHTEYN, E. M. AND IVANOV, L. D. 1976. The approximation of convex sets by polyhedra. *Siber. Math. J. 16*, 852–853.

CALLAHAN, P. B., AND KOSARAJU, S. R. 1995. A decomposition of multidimensional point sets with applications to *k*-nearest-neighbors and *n*-body potential fields. *J. ACM 42*, 67–90.

CHAN, T. M. 1998. Approximate nearest neighbor queries revisited. *Disc. Comput. Geom. 20*, 359–373.

CHAN, T. M. 2002. Closest-point problems simplified on the RAM. In *Proceedings of the 13th Annual ACM-SIAM Symposium on Discrete Algorithms*. ACM, New York, 472–473.

CHAN, T. M. 2006. A minimalist's implementation of an approximate nearest neighbor algorithm in fixed dimensions. Manuscript. (www.cs.vwaterloo.ca/~tmchan/sss.ps).

CHAN, T. M., AND SNOEYINK, J. 1995. Algorithms for approximate nearest-neighbor queries. Manuscript.

CLARKSON, K. L. 1994. An algorithm for approximate closest-point queries. In *Proceedings of the 10th Annual ACM Symposium on Computer Geometry*. ACM, New York, 160–164.

CLARKSON, K. L. 1999. Nearest neighbor queries in metric spaces. *Disc. Comput. Geom. 22*, 63–93.

COLE, R., AND GOTTLIEB, L. 2006. Searching dynamic point sets in spaces with bounded doubling dimension. In *Proceedings of the 38th Annual ACM Symposium on Theory of Computing*. ACM, New York, 574–583.

DA FONSECA, G. D. 2007. Approximate range searching in the absolute error model. Ph.D. dissertation, Department of Computer Science, Univ. Maryland.

DE BERG, M., VAN KREVELD, M., OVERMARS, M., AND SCHWARZKOPF, O. 2000. *Computational Geometry: Algorithms and Applications*, 2nd ed. Springer-Verlag, Berlin, Germany.

DUDLEY, R. M. 1974. Metric entropy of some classes of sets with differentiable boundaries. *J. Approx. Theory 10*, 227–236.

DUNCAN, C. A., GOODRICH, M. T., AND KOBOUROV, S. G. 2001. Balanced aspect ratio trees: Combining the advantages of k-d trees and octrees. *J. Algorithms 33*, 303–333.

EDELSBRUNNER, H. 1987. *Algorithms in Combinatorial Geometry*. EATCS Monographs on Theoretical Computer Science, vol. 10. Springer-Verlag, Heidelberg, West Germany.

EPPSTEIN, D., GOODRICH, M. T., AND SUN, J. Z. 2008. The skip quadtree: A simple dynamic data structure for multidimensional data. *Int. J. Comput. Geom. Appl. 18*, 131–160.

FRANÇOIS, D., WERTZ, V., AND VERLEYSEN, M. 2007. The concentration of fractional distances. *IEEE Trans. Knowl. Data Eng. 19*, 873–886.

FREDERICKSON, G. N. 1997. A data structure for dynamically maintaining rooted trees. *J. Algorithms 24*, 37–65.

GANTMACHER, F. R. 1959. *Matrix Theory, Volume 2*. Chelsea, New York.

GOLUB, G. H., AND LOAN, C. F. V. 1996. *Matrix Computations*, 3rd ed. Johns Hopkins University Press, Baltimore, MD.

HAR-PELED, S. 2001. A replacement for Voronoi diagrams of near linear size. In *Proceedings of the 42nd Annual IEEE Symposium on Foundations of Computer Science*, IEEE Computer Society Press, Los Alamitos, CA, 94–103.

HAR-PELED, S. 2008. Geometric approximation algorithms. (Lecture notes available from `http://valis.cs.uiuc.edu/~sariel/teach/notes/aprx/`).

HAR-PELED, S., AND MENDEL, M. 2006. Fast construction of nets in low dimensional metrics, and their applications. *SIAM J. Comput. 35*, 1148–1184.

INDYK, P., AND MOTWANI, R. 1998. Approximate nearest neighbors: Towards removing the curse of dimensionality. In *Proceedings of the 30th Annual ACM Symposium on Theory of Computing*. ACM, New York, 604–613.

KARGER, D. R., AND RUHL, M. 2002. Finding nearest neighbors in growth-restricted metrics. In *Proceedings of the 34th Annual ACM Symposium on Theory of Computing*. ACM, New York, 741–750.

KRAUTHGAMER, R., AND LEE, J. R. 2004. Navigating nets: Simple algorithms for proximity search. In *Proceedings of the 15th Annual ACM-SIAM Symposium on Discrete Algorithms*. ACM, New York, 798–807.

KRAUTHGAMER, R., AND LEE, J. R. 2005. The black-box complexity of nearest-neighbor search. *Theo. Comp. Sci. 348*, 262–276.

KUSHILEVITZ, E., OSTROVSKY, R., AND RABANI, Y. 2000. Efficient search for approximate nearest neighbor in high dimemsional spaces. *SIAM J. Comput. 30*, 457–474.

PANIGRAHY, R. 2006. Entropy based nearest neighbor search in high dimensions. In *Proceedings of the 17th Annual ACM-SIAM Symposium on Discrete Algorithms*. ACM, New York, 1186–1195.

PREPARATA, F. P., AND SHAMOS, M. I. 1990. *Computational Geometry: An Introduction*, 3rd ed. Springer-Verlag, Berlin, Germany.

SABHARWAL, Y., SEN, S., AND SHARMA, N. 2006. Nearest neighbors search using point location in balls with applications to approximate Voronoi decompositions. *J. Comput. Sys. Sci. 72*, 955–977.

SAMET, H. 1990. *The Design and Analysis of Spatial Data Structures*. Addison-Wesley, Reading, MA.

VLEUGELS, J., AND OVERMARS, M. 1998. Approximating Voronoi diagrams of convex sites in any dimension. *Int. J. Comput. Geom. Appl. 8*, 201–222.

YAO, A. C. 1982. On constructing minimum spanning trees in k-dimensional spaces and related problems. *SIAM J. Comput. 11,* 4, 721–736.