Nearest-Neighbor Searching Under Uncertainty II*

Pankaj K. Agarwal† Boris Aronov‡ Sariel Har-Peled§ Jeff M. Philips¶

Ke Yi‖ Wuzhou Zhang**

May 6, 2017

Abstract

Nearest-neighbor search, which returns the nearest neighbor of a query point in a set of points, is an important and widely studied problem in many fields, and it has wide range of applications. In many of them, such as sensor databases, location-based services, face recognition, and mobile data, the location of data is imprecise. We therefore study nearest-neighbor queries in a probabilistic framework in which the location of each input point is specified as a probability distribution function. We present efficient algorithms for (i) computing all points that are nearest neighbors of a query point with nonzero probability; and (ii) estimating the probability of a point being the nearest neighbor of a query point, either exactly or within a specified additive error.

1 Introduction

Nearest-neighbor search is a fundamental problem in data management. It has applications in such diverse areas as spatial databases, information retrieval, data mining, pattern recognition, etc. In its simplest form, it asks for preprocessing a set S of n points in $\mathbb{R}^d$ into a data structure so that given any query point q, the nearest neighbor (NN) of q in S can be reported efficiently. This problem has been studied extensively in database, machine learning, and computational geometry communities, and is now

*A preliminary version of this article appeared in *Proceedings of the ACM Symposium on Principles of Database Systems* (PODS), 2013. The title "*Nearest-Neighbor Searching Under Uncertainty I*" has been reserved for the journal version of [AESZ12]. Most of the work on this paper was done while W. Zhang was at Duke University.

†Department of Computer Science; Duke University; Durham, NC, 27708, USA; pankaj@cs.duke.edu; http://www.cs.duke.edu/~pankaj/. Research on this paper was partially supported by NSF under grants CCF-09-40671, CCF-10-12254, CCF-11-61359, and IIS-14-08846.

‡Polytechnic School of Engineering, New York University; New-York City, NY, USA; Research on this paper has been partially supported by NSF grants CCF-08-30691, CCF-11-17336, and CCF-12-18791, and by NSA MSP Grant H98230-10-1-0210.

§Department of Computer Science; University of Illinois; 201 N. Goodwin Avenue; Urbana, IL, 61801, USA; sariel@illinois.edu; http://sarielhp.org/. Research on this paper was partially supported by NSF grants CCF-09-15984 and CCF-12-17462.

¶School of Computing; The University of Utah; The research on this paper was partially supported by NSF CCF-1350888, IIS-1251019, and ACI-1443046.

‖Hong Kong University of Science and Technology; K. Yi is supported by HKRGC under grants GRF-621413 and GRF-16211614.

**Apple Inc. Research on this paper was partially supported by NSF under grants CCF-09-40671, CCF-10-12254, CCF-11-61359, and IIS-14-08846.

relatively well understood. However, in some of the applications mentioned above, data are imprecise and are often modeled as probabilistic distributions. This has led to a flurry of research activities on query processing over probabilistic data, including the NN problem; see [Agg09, DRS09] for surveys on uncertain data, and see, e.g., [CXY+10, LS07] for application scenarios of NN search under uncertainty.

Despite many efforts devoted to the probabilistic NN problem, it still lacks a theoretical foundation. Specifically, not only are we yet to understand its complexity (is the problem inherently more difficult than on precise data?), but we also lack efficient algorithms to solve it. Furthermore, existing solutions all use heuristics without nontrivial performance guarantees. This paper addresses some of these issues.

1.1 Problem definition

An *uncertain* point P in $\mathbb{R}^2$ is represented as a continuous probability distribution defined by a probability density function (pdf) $f_P\colon \mathbb{R}^2 \to \mathbb{R}_{\geq 0}$; f_P may be a parametric pdf such as a uniform distribution or a Gaussian distribution, or may be a non-parametric pdf such as a histogram[1]. The *uncertainty region* of P (or the *support* of f_P) is the set of points for which f_P is positive, i.e., $\mathrm{Sup}\ f_P = \{x \in \mathbb{R}^2 \mid f_P(x) > 0\}$. We assume P has a bounded uncertainty region: if f_P is Gaussian, we work with the truncated Gaussian, as in [BSI08, CCMC08]. We also consider the case where P is represented as a discrete distribution defined by a finite set $P = \{p_1, \ldots, p_k\} \subset \mathbb{R}^2$ along with a set of *location probabilities* $\{w_1, \ldots, w_k\} \subset (0, 1]$, where $w_i = \Pr[P \text{ is } p_i]$ and $\sum_{i=1}^k w_i = 1$; and we say that P has a discrete distribution of *description complexity* k. Let $\mathcal{P} = \{P_1, \ldots, P_n\}$ be a set of n uncertain points in $\mathbb{R}^2$, and let $d(\cdot, \cdot)$ be the Euclidean distance.

Fix a point $q \in \mathbb{R}^2$ and an integer $i \in \{1, \ldots, n\}$. We define $\pi_i(q) = \pi(P_i, q)$ to be the probability of $P_i \in \mathcal{P}$ being the nearest neighbor of q, referred to as the *quantification probability* of q (for P_i). Next, let $g_{q,i}$ be the pdf of the distance between q and P_i. That is,

$$g_{q,i}(r) = \Pr[r \leq d(q, P_i) \leq r + \mathrm{d}r]/\mathrm{d}r.$$

See Figure 1 for an example of $g_{q,i}$. Let $G_{q,i}(r) = \int_0^r g_{q,i}(r')\mathrm{d}r'$ denote the cumulative distribution function (cdf) of the distance between q and P_i. Note that if P_i is the NN of q and $d(P_i, q) = r$ then $d(P_j, q) > r$ for all $j \neq i$. Therefore $\pi_i(q)$ can be expressed as follows:

$$\pi_i(q) = \int_0^\infty g_{q,i}(r) \prod_{j \neq i} \big(1 - G_{q,j}(r)\big)\mathrm{d}r. \tag{1}$$

If P_i's are represented by discrete distributions, then Eq. (1) can be rewritten as follows:

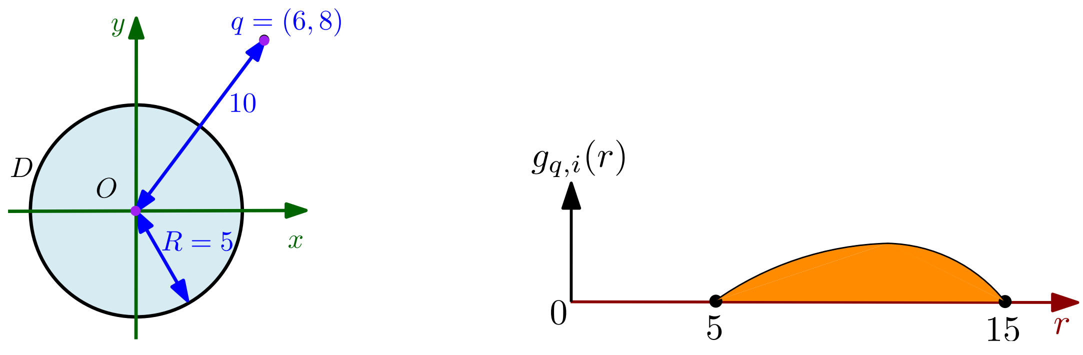
$$\pi_i(q) = \sum_{p_{is} \in P_i} w_{is} \prod_{j \neq i} \Big(1 - G_{q,j}\big(d(p_{is}, q)\big)\Big), \tag{2}$$

where $G_{q,j}(r) = \sum_{d(p_{jt}, q) \leq r} w_{jt}$.

Given a set $\mathcal{P}$ of n uncertain points, the *probabilistic nearest neighbor* (PNN) problem is to preprocess $\mathcal{P}$ into a data structure so that, for any given query point q, we can efficiently return all pairs $(P_i, \pi_i(q))$ with $\pi_i(q) > 0$.

Usually, the PNN problem is divided into the following two subproblems, which are often considered separately.

[1] If the location of data is precise, we call it *certain*. The probabilistic model we use is often called the *locational model*, where the location of an uncertain point follows the given distribution. This is to be contrasted with the *existential model*, where each point has a precise location but it appears with a given probability.



(a) (b)

Figure 1: (a) P_i is represented by a uniform distribution defined on a disk D of radius $R = 5$ and centered at origin O, $q = (6, 8)$; (b) $g_{q,i}(r)$, the pdf of the distance function between q and P_i.

Nonzero NNs. The first subproblem is to find all the P_i's with $\pi_i(q) > 0$ without computing the actual quantification probabilities, i.e., to find

$$\mathsf{NN}_{\neq 0}(q, \mathcal{P}) = \{P_i \mid \pi_i(q) > 0\}.$$

If the point set $\mathcal{P}$ is obvious from the context, we drop the argument $\mathcal{P}$ from $\mathsf{NN}_{\neq 0}(q, \mathcal{P})$, and write it as $\mathsf{NN}_{\neq 0}(q)$. Note that $\mathsf{NN}_{\neq 0}(q)$ depends (besides q) only on the uncertainty regions of the uncertain points, but not on the actual pdf's.

A possible approach to compute nearest neighbors is to use Voronoi diagrams. For example, the standard Voronoi diagram of a set of certain points in $\mathbb{R}^2$ is the planar subdivision so that all points in the same face have the same nearest neighbor. In our case, we define the *nonzero Voronoi diagram*, denoted by $\mathcal{V}_{\neq 0}(\mathcal{P})$, to be the subdivision of $\mathbb{R}^2$ into maximal connected regions such that $\mathsf{NN}_{\neq 0}(q)$ is the same for all points q within each region. That is, for a subset $\mathcal{T} \subseteq \mathcal{P}$, let

$$\mathrm{cell}_{\neq 0}(\mathcal{T}) = \{q \in \mathbb{R}^2 \mid \mathsf{NN}_{\neq 0}(q) = \mathcal{T}\}. \quad (3)$$

Although there are 2^n subsets of $\mathcal{P}$, we will see below that only a small number of them have nonempty Voronoi cells. The planar subdivision $\mathcal{V}_{\neq 0}(\mathcal{P})$ is induced by all the nonempty $\mathrm{cell}_{\neq 0}(\mathcal{T})$'s for $\mathcal{T} \subseteq \mathcal{P}$. The *(combinatorial) complexity* of $\mathcal{V}_{\neq 0}(\mathcal{P})$ is the total number of vertices, edges, and faces in $\mathcal{V}_{\neq 0}(\mathcal{P})$. The complexity of the Voronoi diagram is often regarded as a measure of the complexity of the corresponding nearest-neighbor problem.

In this paper, we study the worst-case complexity of $\mathcal{V}_{\neq 0}(\mathcal{P})$ and how it can be efficiently constructed. In addition, once we have $\mathcal{V}_{\neq 0}(\mathcal{P})$, it can be preprocessed into a point-location structure to support $\mathsf{NN}_{\neq 0}$ queries in logarithmic time.

Computing quantification probabilities. The second subproblem is to compute the quantification probability $\pi_i(q)$ for a given q and P_i. Exact values of these probabilities are often unstable — a far away point can affect these probabilities — and computing them requires complex n-dimensional integration (see Eq. (1)), which is often expensive. As such, we resort to computing $\pi_i(q)$ approximately within a given additive error tolerance $\varepsilon \in (0, 1)$. More precisely, we aim at returning a value $\widehat{\pi}_i(q)$ such that $|\pi_i(q) - \widehat{\pi}_i(q)| \leq \varepsilon$.

1.2 Previous work

Nonzero NNs. [SE08] showed that if the uncertainty regions of the points in $\mathcal{P}$ are disks, then the complexity of $\mathcal{V}_{\neq 0}(\mathcal{P})$ is $O(n^4)$ (though they did not use this term explicitly); they did not offer any lower bound. If one only considers those cells of $\mathcal{V}_{\neq 0}(\mathcal{P})$ in which $\mathsf{NN}_{\neq 0}(q)$ contains only one uncertain point P_i, i.e., only P_i has a non-zero probability of being the NN of q, they showed that the complexity of these cells is $O(n)$. Note that for such a cell, we always have $\pi_i(q) = 1$ for any q in the cell, so they form the *guaranteed Voronoi diagram*. Probably unaware of the work by [SE08], [CXY+10] proved an exponential upper bound for the complexity of the nonzero Voronoi diagram, which they referred to as UV-diagram.

The nonzero Voronoi diagram is not the only way to find the nonzero NNs. [CKP04] designed a branch-and-prune solution based on the R-tree. Recently, [ZCM+13] proposed to combine the nonzero Voronoi diagram with R-tree-like bounding rectangles. These methods do not provide any nontrivial performance guarantees.

Computing quantification probabilities. Computing the quantification probabilities has attracted much attention in the database community. [CKP04] used numerical integration, which is quite expensive. [CCMC08] and [BEK+11] proposed some filter-refinement methods to give upper and lower bounds on the quantification probabilities. [KKR07] took a random sample from the continuous distribution of each uncertain point to convert it to a discrete one, so that the integration becomes a sum, and they clustered each sample to further reduce the complexity of the query computation. [DYM+05] considered the problem of reporting points P_i for which $\pi_i(q)$ exceeds some given threshold. We note that these methods are best-effort based: they do not always give the ε-error that we aim at — how tight the resulting bounds are depends on the data.

Other variants of the problem. The PNN problem we focus on in this paper is the most commonly studied version of the problem, but many variants and extensions have been considered.

Besides using the quantification probability, one can also consider the expected distance from a query point q to an uncertain point, and return the one minimizing the expected distance as the nearest neighbor; this was studied by [AESZ12]. This NN definition is easier since the expected distance to each uncertain point can be computed separately, whereas the quantification probability involves the interaction among all uncertain points. However, the expected nearest neighbor is not a good indicator under large uncertainty (see [YTX+10] for details).

Instead of returning only the nearest neighbor, one can ask to return the k nearest neighbors in a ranked order (the kNN problem). If we use expected distance, the ranking of points is straightforward, namely, rank them in a non-decreasing order of the expected distance from the query point. However, when quantification probabilities are considered, many different criteria for ranking the results are possible, leading to different problem variants [JCLY11].

Various combinations of these extensions have been studied in the literature; see, e.g., [BSI08, CCCX09, KCS14, LS07, YTX+10].

1.3 Our results

The main results of this paper are the following:

(i) A $\Theta(n^3)$ bound on the combinatorial complexity $\mathcal{V}_{\neq 0}(\mathcal{P})$, an improved quadratic bound on the complexity of $\mathcal{V}_{\neq 0}(\mathcal{P})$ for a special case, and an efficient randomized algorithm for computing $\mathcal{V}_{\neq 0}(\mathcal{P})$;

(ii) Near-linear-size data structures for answering $\mathsf{NN}_{\neq 0}$ queries in polylogarithmic or sublinear time;

(iii) Efficient data structures for computing the quantification probabilities of a query point approximately.

We now describe these results in more detail:

Nonzero Voronoi diagrams. We first study (in Section 2) the complexity of $\mathcal{V}_{\neq 0}(\mathcal{P})$. Suppose the uncertainty region of each $P_i \in \mathcal{P}$ is a disk and $d(\cdot,\cdot)$ is the L_2 metric. We show that $\mathcal{V}_{\neq 0}(\mathcal{P})$ has $O(n^3)$ complexity, and that this bound is tight in the worst case even if all uncertainty-region disks have the same radius. This significantly improves the bound in [SE08] and closes the problem. We also show that the $O(n^3)$ bound holds for a much larger class of uncertainty regions, namely, even if each uncertainty region is a semialgebraic set of constant description complexity; see Section 2 for the definition of a semialgebraic set.

If the disks are pairwise disjoint and the ratio of their radii is at most λ, then the complexity of $\mathcal{V}_{\neq 0}(\mathcal{P})$ is $O(\lambda n^2)$, and we prove a lower bound of $\Omega(n^2)$. Again, this bound holds for a larger class of uncertainty regions.

We show that if each point in $\mathcal{P}$ has a discrete distribution of description complexity at most k, then $\mathcal{V}_{\neq 0}(\mathcal{P})$ has $O(kn^3)$ complexity.

We present a randomized, output-sensitive algorithm for computing $\mathcal{V}_{\neq 0}(\mathcal{P})$ in $O(n^2 \log n + \mu)$ expected time, where μ is the complexity of $\mathcal{V}_{\neq 0}(\mathcal{P})$. $\mathcal{P}$ can be preprocessed into a point-location structure of size $O(\mu)$ that can answer an $\mathsf{NN}_{\neq 0}$ query in $O(\log n + t)$ time, where t is the output size [dBCKO08].

Answering $\mathsf{NN}_{\neq 0}$ queries. Since the complexity of $\mathcal{V}_{\neq 0}(\mathcal{P})$ can be cubic in the worst case, in Section 3 we present near-linear size data structures for answering $\mathsf{NN}_{\neq 0}$ queries efficiently. In particular, if the uncertainty region of each point is a disk then an $\mathsf{NN}_{\neq 0}$ query can be answered in $O(\log n + t)$ time using $O(n \operatorname{polylog}(n))$ space, where t is the output size. If each point of $\mathcal{P}$ has a discrete distribution of size at most k, then an $\mathsf{NN}_{\neq 0}$ query can be answered in $O(N^{1/2} \log^3 N + t)$ time using $O(N \log^2 N)$ space, where t is the output size and $N = nk$. These results rely on geometric data structures for answering simplex range queries and their variants; see [Aga16] for a recent survey.

Computing quantification probabilities. Next, in Section 4, we focus our attention on computing quantification probabilities for a query point q, i.e., reporting the values of $\pi_i(q)$ for all P_i's for which $\pi_i(q) > 0$. We begin in Section 4.1, by describing a data structure that can compute quantification probabilities exactly if each P_i has a discrete distribution of size at most k. Its size is $O(N^4)$ and it can return all t positive quantification probabilities for a query point in time $O(\log N + t)$, where $N = nk$ as above. Since computing quantification probabilities is expensive even for points with discrete distributions, we mostly focus on computing them approximately.

We present two data structures for approximating the quantification probabilities efficiently. The first (see Section 4.2) is a Monte-Carlo algorithm for estimating $\pi_i(q)$ for any P_i and q within additive error ε with probability at least $1-\delta$, for parameters $\varepsilon, \delta \in (0,1)$. We argue that if each uncertain point has a discrete distribution of size at most k, then we can estimate $\pi_i(q)$ within additive error ε with probability at least $1-\delta$ by using $s_{\varepsilon,\delta} = O((1/\varepsilon^2)\log(N/\delta))$ random instantiations of $\mathcal{P}$. (Note that there are at most $1/\varepsilon$ P_i's for which $\pi_i(q) > \varepsilon$.) Consequently, we can preprocess $\mathcal{P}$ into a data structure of size $O((n/\varepsilon^2)\log(N/\delta))$ so that for any query point $q \in \mathbb{R}^2$, $\pi_i(q)$ for all P_i's can be estimated within additive error ε in $O((1/\varepsilon^2)\log(N/\delta)\log n)$ time, with probability at least $1-\delta$. The algorithm explicitly computes the estimates of $\pi_i(q)$'s for at most $s_{\varepsilon,\delta}$ points and sets the estimate to 0 for the rest of the points. We also show that this approach works even if the distribution of each P_i is continuous, by approximating a continuous distribution with a discrete one. A key observation is that it suffices to sample a polynomial number of points from the distribution of each P_i to ensure that the error in the quantification probability is at most ε.

Next, in Section 4.3 we describe a deterministic algorithm for computing $\pi_i(q)$ approximately if each point has a discrete distribution of size at most k. We show that $\mathcal{P}$ can be preprocessed into a data structure of $O(N)$ size so that for any $q \in \mathbb{R}^2$ and for any $\varepsilon \in (0,1)$, $\pi_i(q)$, for all $i \in \{1, \ldots, n\}$, can be computed with additive error at most ε in $O(\rho k \log(\rho/\varepsilon) + \log N)$ time, where ρ is the ratio of the largest to the smallest location probabilities over all possible locations of points in $\mathcal{P}$. We show that there are at most $m(\rho, \varepsilon) = \rho k \ln(\rho/\varepsilon) + k - 1$ points of $\mathcal{P}$ for which $\pi_i(q) > \varepsilon$. The algorithm explicitly estimates $\pi_i(q)$ for at most $m(\rho, \varepsilon)$ points and sets the estimate to 0 for the rest of the points.

2 Nonzero probabilistic Voronoi diagram

Let $\mathcal{P}$ be a set of n uncertain points as described earlier. We analyze the combinatorial structure of $\mathcal{V}_{\neq 0}(\mathcal{P})$ and describe algorithms for constructing it. We first consider the case when the distribution of each point is continuous and then consider the discrete case.

2.1 Continuous case

For simplicity, we first assume that the uncertainty region of each P_i is a circular disk D_i of radius r_i centered at c_i.

We first observe that the structure of $\mathcal{V}_{\neq 0}(\mathcal{P})$ does not depend on the actual pdf of P_i's. What really matters is the uncertainty region D_i. More precisely, for each $1 \leq i \leq n$ and for $q \in \mathbb{R}^2$, let

$$\Delta_i(q) = \max_{p \in D_i} d(q, p) = d(q, c_i) + r_i,$$
$$\delta_i(q) = \min_{p \in D_i} d(q, p) = \max\{d(q, c_i) - r_i, 0\}$$

be the maximum and minimum possible distance, respectively, from q to P_i.

The following lemma, whose proof is straightforward, characterizes the structure of $\mathcal{V}_{\neq 0}(\mathcal{P})$.

Lemma 2.1. *For a point $q \in \mathbb{R}^2$, a point $P_i \in \mathcal{P}$ belongs to $\mathsf{NN}_{\neq 0}(q, \mathcal{P})$ if and only if*

$$\delta_i(q) < \Delta_j(q) \text{ for all } 1 \leq j \neq i \leq n.$$

Let $\Delta\colon \mathbb{R}^2 \to \mathbb{R}$ denote the *lower envelope*[2] of $\Delta_1, \ldots, \Delta_n$; that is, for any $q \in \mathbb{R}^2$,

$$\Delta(q) = \min_{1 \leq i \leq n} \Delta_i(q).$$

The projection of the graph of $\Delta(x)$ onto the xy-plane is the additive-weighted Voronoi diagram of the points $c_1, \ldots, c_n$, where the weight of c_i is r_i, and the weighted distance from q to c_i is $d(q, c_i) + r_i$, for $i = 1, \ldots, n$. Let $\mathbb{M}$ denote this planar subdivision. It has linear complexity and each of its edges is a hyperbolic arc; see [AB86]. Lemma 2.1 implies that, for any $q \in \mathbb{R}^2$,

$$\mathsf{NN}_{\neq 0}(q, \mathcal{P}) = \{P_i \mid \delta_i(q) < \Delta(q)\}. \tag{4}$$

[2]The *lower envelope*, L_F, of a set F of functions is their pointwise minimum, i.e., $L_F(x) = \min_{f \in F} f(x)$. The *upper envelope*, U_F, of F is the pointwise maximum, i.e., $U_F(x) = \max_{f \in F} f(x)$.

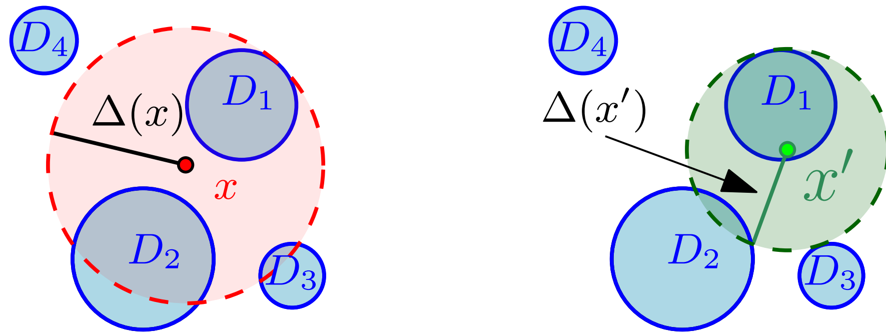


Figure 2: $\mathcal{P} = \{P_1, \ldots, P_5\}$, $\Delta(x) = \Delta_1(x)$, $\mathsf{NN}_{\neq 0}(x, \mathcal{P}) = \{P_1, P_2, P_3\}$, $\Delta(x') = \Delta_1(x')$, $\mathsf{NN}_{\neq 0}(x', \mathcal{P}) = \{P_1, P_2\}$, and x' lies on an edge of $\mathcal{V}_{\neq 0}(\mathcal{P})$.

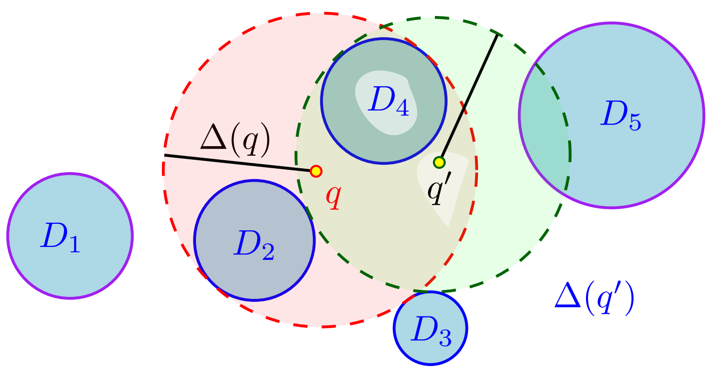


Figure 3: The point q is a break point of γ_3 and q' is an intersection point of γ_2 and γ_3.

See Figure 2. It also implies that, as we move x continuously in $\mathbb{R}^2$, $\mathsf{NN}_{\neq 0}(x, \mathcal{P})$ remains the same until $\delta_i(x)$, for some $1 \leq i \leq n$, becomes equal to $\Delta(x)$ (e.g., x' in Figure 2). This observation was made in earlier papers as well; see, e.g. [CCMC08, CKP04]. Using this observation we can now characterize $\mathcal{V}_{\neq 0}(\mathcal{P})$.

For $i = 1, \ldots, n$, let $\gamma_i = \{x \in \mathbb{R}^2 \mid \delta_i(x) = \Delta(x)\}$ be the zero set of the function $\Delta(x) - \delta_i(x)$. Set $\Gamma = \{\gamma_1, \ldots, \gamma_n\}$.

The curve γ_i partitions the plane into two open regions: $\Delta(x) < \delta_i(x)$ and $\Delta(x) > \delta_i(x)$. By Eq. (4), $P_i \in \mathsf{NN}_{\neq 0}(x, \mathcal{P})$ for all points x inside the latter region and for none of the points x inside the former region. It is well known that, for any fixed $j \neq i$, $\gamma_{ij} = \{x \in \mathbb{R}^2 \mid \delta_i(x) = \Delta_j(x)\}$ is a hyperbolic curve [AB86]. The curve γ_i is composed of pieces of γ_{ij}, for $j \neq i$. We refer to the endpoints of these pieces as *breakpoints* of γ_i. They are the intersection points of γ_i with an edge of $\mathbb{M}$ and correspond to points q such that the disk of radius $\Delta(q)$ centered at q touches (at least) two disks of $\mathcal{D}$ from inside, touches D_i from outside, and does not contain any disk of $\mathcal{D}$ in its interior. See Figure 3. Formally, we say that a disk D_1 touches a disk D_2 from the *outside* (resp. *inside*) if $\partial D_1 \cap \partial D_2 \neq \emptyset$ and $\operatorname{int} D_1 \cap \operatorname{int} D_2 = \emptyset$ (resp. $\operatorname{int} D_2 \subseteq \operatorname{int} D_1$).

Lemma 2.2. *The curve γ_i, $1 \leq i \leq n$, has at most $2n$ breakpoints, and it can be computed in $O(n \log n)$ time.*

Proof: Let $\Gamma_i = \{\gamma_{ij} \mid j \neq i, 1 \leq j \leq n\}$. It can be verified that a ray emanating from c_i intersects the hyperbolic curve γ_{ij}, for any $j \neq i$, in at most one point, so γ_{ij} can be viewed as the graph of a function in polar coordinates with c_i as the origin. That is, let $\gamma_{ij} \colon [0, 2\pi) \to \mathbb{R}_{\geq 0}$, where $\gamma_{ij}(\theta)$ is the distance from c_i to γ_{ij} in direction θ. Then, γ_i is the lower envelope of Γ_i. Since each pair of curves

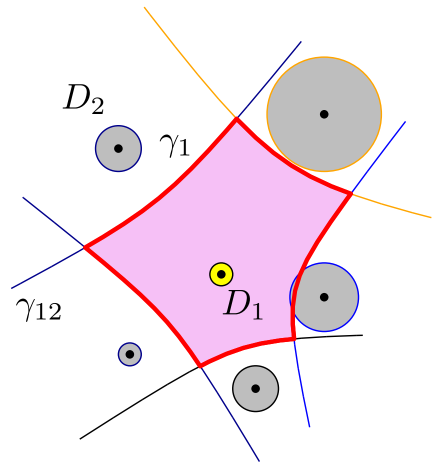


Figure 4: An example of γ_1.

in Γ_i intersects at most twice, a well-known result on lower envelopes implies that γ_i has at most $2n$ breakpoints, and that it can be computed in $O(n \log n)$ time [SA95]. See Figure 4 for an example. ∎

Let $\mathcal{A}(\Gamma)$ denote the planar subdivision induced by Γ: its vertices are the breakpoints of γ_i's and the intersection points of two curves in Γ, its edges are the portions of γ_i's between two consecutive vertices, and its cells are the maximal connected regions of the plane that do not intersect any curve of Γ. We refer to vertices, edges, and cells of $\mathcal{A}(\Gamma)$ as its 0-, 1-, and 2-dimensional *faces*.

For a face ϕ (of any dimension), and for any two points $x, y \in \phi$, the sets $\{P_i \mid \delta_i(x) < \Delta(x)\}$ and $\{P_j \mid \delta_j(y) < \Delta(y)\}$ are the same; we denote this set by $\mathcal{P}_\phi$. Furthermore, if x, y lie in two neighboring faces ϕ and ϕ', respectively, then $\mathcal{P}_\phi \neq \mathcal{P}_{\phi'}$. The following lemma is an immediate consequence of Eq. (4).

Lemma 2.3. *For all points x lying in a face ϕ of $\mathcal{A}(\Gamma)$, $\mathsf{NN}_{\neq 0}(x, \mathcal{P}) = \mathcal{P}_\phi$.*

For a subset $\mathcal{T} \subseteq \mathcal{P}$, let $\mathrm{cell}_{\neq 0}(\mathcal{T})$ be as defined in Eq. (3). An immediate corollary of the above lemma is:

Corollary 2.4. *(i) For any $\mathcal{T} \subseteq \mathcal{P}$, $\mathrm{cell}_{\neq 0}(\mathcal{T}) \neq \emptyset$ if and only if there is a face ϕ of $\mathcal{A}(\Gamma)$ with $\mathcal{T} = \mathcal{P}_\phi$.*
(ii) The planar subdivision $\mathcal{A}(\Gamma)$ coincides with $\mathcal{V}_{\neq 0}(\mathcal{P})$.

We now bound the complexity of $\mathcal{A}(\Gamma)$ and thus of $\mathcal{V}_{\neq 0}(\mathcal{P})$.

Theorem 2.5. *Let $\mathcal{P} = \{P_1, \ldots, P_n\}$ be a set of n uncertain points in $\mathbb{R}^2$ whose uncertainty regions are disks. Then $\mathcal{V}_{\neq 0}(\mathcal{P})$ has $O(n^3)$ complexity. Moreover, it can be computed in $O(n^2 \log n + \mu)$ expected time, where μ is the complexity of $\mathcal{V}_{\neq 0}(\mathcal{P})$.*

Proof: Using a standard perturbation argument (see, e.g., [SA95]), it suffices to bound the complexity of $\mathcal{V}_{\neq 0}(\mathcal{P})$ when the disks corresponding to the uncertainty regions of the points of $\mathcal{P}$ are in general position, so we can assume that the degree of every vertex in $\mathcal{V}_{\neq 0}(\mathcal{P})$ is constant. Since $\mathcal{V}_{\neq 0}(\mathcal{P})$ is a planar subdivision and the degree of every vertex is constant, the number of edges and cells in $\mathcal{V}_{\neq 0}(\mathcal{P})$ is proportional to the number of its vertices. Hence, it suffices to bound the number of vertices. Let $\Gamma = \{\gamma_1, \ldots, \gamma_n\}$ be the set of curves as defined above. By Lemma 2.2, each γ_i has $O(n)$ breakpoints, so there are a total of $O(n^2)$ breakpoints. We claim that each pair of curves γ_i and γ_j intersect $O(n)$ times — each such intersection point corresponds to a point $v \in \mathbb{R}^2$ such that the disk of radius $\Delta(v)$ centered

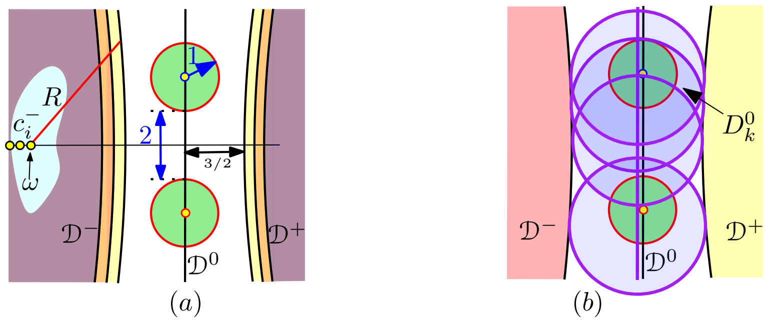


Figure 5: (a) $\Omega(n^3)$ lower bound construction with $m = 3$; only some disks are drawn. (b) Illustration of the proof.

at v touches D_i and D_j from the outside and another disk D_k of $\mathcal{D}$, the one realizing the value of $\Delta(v)$, from the inside (e.g., q' in Figure 3). For a fixed k, we show that there are at most two points v such that $\delta_i(v) = \delta_j(v) = \Delta_k(v)$. Note that $\delta_i(v) = \Delta_k(v)$ represents either an empty set or one hyperbolic branch, and the same holds for $\delta_j(v) = \Delta_k(v)$. Two such hyperbolic branches intersect at most twice, implying that $\delta_i(v) = \delta_j(v) = \Delta_k(v)$ contributes at most two vertices. Hence, the number of vertices in $\mathcal{V}_{\neq 0}(\mathcal{P})$ is $O(n^3)$, as claimed.

By Lemma 2.2, Γ can be computed in $O(n^2 \log n)$ time. The planar subdivision $\mathcal{A}(\Gamma)$ can be computed in $O(n \log n + \mu)$ expected time using randomized incremental method [AS00], where μ is the complexity of $\mathcal{V}_{\neq 0}(\mathcal{P})$. Hence $\mathcal{V}_{\neq 0}(\mathcal{P})$ can be computed in $O(n^2 \log n + \mu)$ expected time. ■

The above argument is quite general and extends to a large class of uncertainty regions. In particular, a two-dimensional *semialgebraic set* is a subset of $\mathbb{R}^2$ obtained from a finite number of sets of the form $\{x \in \mathbb{R}^2 \mid g(x) \geq 0\}$, where g is a bivariate polynomial with real coefficients, by Boolean operations (union, intersection, and complement). A semialgebraic set has *constant description complexity* if the number of polynomials defining the set as well as the maximum degree of these polynomials is a constant. For example, a polygon with constant number of edges and a region defined by a constant number of quadratic arcs are semialgebraic sets of constant description complexity.

Suppose the uncertainty region of each point in $\mathcal{P}$ is a semialgebraic set of constant description complexity, denoted by σ_i.

The analysis for the case of disks shows that a vertex of $\mathcal{V}_{\neq 0}(\mathcal{P})$ is the center of a disk that touches uncertainty regions of three different points. Fix a triple of uncertainty regions $\sigma_i, \sigma_j, \sigma_k$. Since they are semialgebraic sets of constant complexity, there are only $O(1)$ disks that are tangent to σ_1, σ_2, and σ_3 simultaneously. Therefore, $\mathcal{V}_{\neq 0}(\mathcal{P})$ has $O(n^3)$ vertices, which in view of the above discussion implies that $\mathcal{V}_{\neq 0}(\mathcal{P})$ has $O(n^3)$ combinatorial complexity. Assuming an extension of the real RAM model of computation in which the roots of polynomials of constant degree can be computed exactly in $O(1)$ time, the randomized algorithm described above can be extended to this case as well. Omitting further details, we conclude the following:

Theorem 2.6. *Let $\mathcal{P} = \{P_1, \ldots, P_n\}$ be a set of n uncertain points in $\mathbb{R}^2$ whose uncertainty regions are semialgebraic sets of constant description complexity. Then $\mathcal{V}_{\neq 0}(\mathcal{P})$ has $O(n^3)$ complexity. Moreover, it can be computed in $O(n^2 \log n + \mu)$ expected time, where μ is the complexity of $\mathcal{V}_{\neq 0}(\mathcal{P})$.*

Theorem 2.7. *There exists a set $\mathcal{P}$ of n uncertain points whose uncertainty regions are disks such that $\mathcal{V}_{\neq 0}(\mathcal{P})$ has $\Omega(n^3)$ vertices.*

Proof: Assume that $n = 4m$ for some $m \in \mathbb{N}^+$. We choose two parameters $R = 8n^2$ and $\omega = 1/n^2$. We construct three families of disks: $\mathcal{D}^- = \{D_1^-, \ldots, D_m^-\}$, $\mathcal{D}^+ = \{D_1^+, \ldots, D_m^+\}$, and $\mathcal{D}^0 = \{D_1^0, \ldots, D_{2m}^0\}$. The radius of all disks in $\mathcal{D}^- \cup \mathcal{D}^+$ is R and their centers lie on the x-axis; the radius of all disks in $\mathcal{D}^0$ is 1 and their centers lie on the y-axis. More precisely, for $1 \le i, j \le m$, the center of D_i^- is $c_i^- = (-R - 3/2 - (i-1)\omega, 0)$ and the center of D_j^+ is $c_j^+ = (R + 3/2 + (j-1)\omega, 0)$, and for $1 \le k \le 2m$, the center of D_k^0 is $(0, 4(k-m) - 2)$. See Figure 5(a).

We claim that for every triple i, j, k with $1 \le i, j \le m$ and $1 \le k \le 2m$, there are two disks each of which touches D_i^- and D_j^+ from the outside and D_k^0 from the inside and does not contain any disk of $\mathcal{D}^- \cup \mathcal{D}^+ \cup \mathcal{D}^0$ in its interior. See Figure 5(b).

Fix such a triple i, j, k. Since the radii of D_i^- and D_j^+ are the same, the locus b_{ij} of the centers of disks that simultaneously touch D_i^- and D_j^+ from the outside is the bisector of their centers, i.e., b_{ij} is the vertical line $x = (x(c_i^-) + x(c_j^+))/2 = (j-i)\omega/2$. Let σ_{ij} denote the intersection point of b_{ij} and the x-axis; $\sigma_{ij} = (\frac{1}{2}(j-i)\omega, 0)$. A point on b_{ij} can be represented by its y-coordinate; we will not distinguish between the two. For y-value a, let ξ_a be the disk centered at a and simultaneously touching D_i^- and D_j^+ from the outside. The radius of ξ_a is

$$\|a - c_i^-\| - R = \sqrt{a^2 + \|c_i^- - \sigma_{ij}\|^2} - R = \sqrt{a^2 + \left(R + 3/2 + \left(\frac{i+j}{2} - 1\right)\omega\right)^2} - R.$$

The radius of ξ_a is thus at least $3/2$, and for $a \in [-4m, 4m]$, it is at most 2 (using the fact that $R \ge 8n^2$ and $\omega = 1/n^2$). Hence, for $a \in [-4m, 4m]$, ξ_a contains at most one disk of $\mathcal{D}^0$ in its interior, and obviously ξ_a does not contain any disk of $\mathcal{D}^- \cup \mathcal{D}^+$ in its interior.

Let $a_k = 4(k-m) - 2$. Then, the disk ξ_{a_k} contains D_k^0 in its interior because the distance between the centers of D_k^0 and ξ_{a_k} is at most $m\omega \le 1/(4n)$, the radius of D_k^0 is 1, and the radius of ξ_{a_k} is at least $3/2$. On the other hand, the disk ξ_a for $a = a_k \pm 2$ does not contain D_k^0 in its interior because the radius of ξ_a is at most 2 and the distance between the center of D_k^0 and ξ_a is at least 2. Therefore, by a continuity argument, there is a value $a^+ \in [a_k, a_k + 2]$ at which ξ_{a^+} touches D_k^0 from the inside. Similarly, there is a value $a^- \in [a_k - 2, a_k]$ at which ξ_{a^-} touches D_k^0 from the inside.

This proves the claim that there are two disks touching D_i^- and D_j^+ from the outside and D_k^0 from the inside and not containing any disk of $\mathcal{D}^- \cup \mathcal{D}^+ \cup \mathcal{D}^0$ in its interior. In other words, each triple i, j, k contributes two vertices to $\mathcal{V}_{\neq 0}(\mathcal{P})$. Hence, $\mathcal{V}_{\neq 0}(\mathcal{P})$ has $\Omega(n^3)$ vertices. ■

Next, we show that the maximum complexity of $\mathcal{V}_{\neq 0}(\mathcal{P})$ is $\Omega(n^3)$ even if the uncertainty regions of points in $\mathcal{D}$ are disks of the same radius.

Theorem 2.8. *There exists a set $\mathcal{P}$ of n uncertain points, whose uncertainty regions are disks of the same radius, for which $\mathcal{V}_{\neq 0}(\mathcal{P})$ has $\Omega(n^3)$ vertices.*

Proof: Assume that $n = 3m$ for some $m \in \mathbb{N}^+$. We choose two parameters $\theta = \frac{\pi}{2} \cdot \frac{1}{(m+1)}$, and a sufficiently small positive number ω. We construct three families of disks: $\mathcal{D}^- = \{D_1^-, \ldots, D_m^-\}$, $\mathcal{D}^+ = \{D_1^+, \ldots, D_m^+\}$, and $\mathcal{D}^0 = \{D_1^0, \ldots, D_m^0\}$. Without loss of generality, we set the radius of all disks to 1. The centers of disks in $\mathcal{D}^- \cup \mathcal{D}^+$ lie on the x-axis, and the centers of disks in $\mathcal{D}^0$ lie in the first quadrant. More precisely, for $1 \le i, j \le m$, the center of D_i^- is $c_i^- = (-2 - (i-1)\omega, 0)$ and the center of D_j^+ is $c_j^+ = (2 + (j-1)\omega, 0)$, and for $1 \le k \le m$, the center of D_k^0 is $(2 - 2\cos(k\theta), 2\sin(k\theta))$. See Figure 6(a).

We claim that for every triple i, j, k with $1 \le i, j, k \le m$, there is a disk touching D_i^- and D_j^+ from the outside and D_k^0 from the inside and not containing any disk of $\mathcal{D}^- \cup \mathcal{D}^+ \cup \mathcal{D}^0$ in its interior.

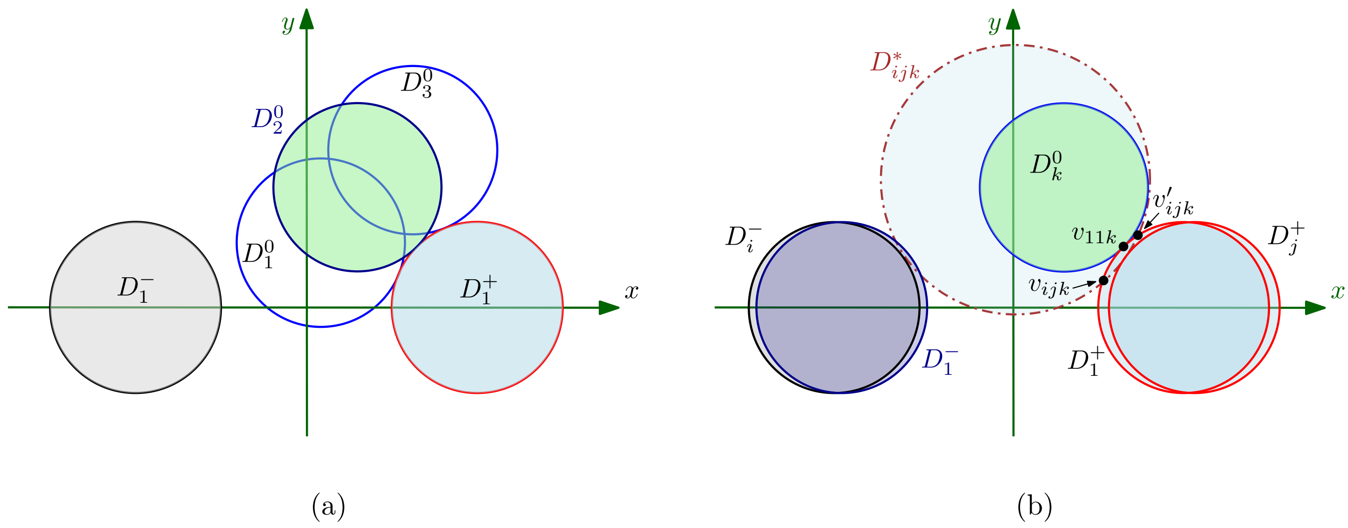


Figure 6: (a) $\Omega(n^3)$ lower bound construction using disks of same radius with $m = 3$; only some disks are drawn. (b) Illustration of the proof.

First of all, this is true for $i = j = 1$ and $1 \le k \le m$. Note that D_1^- is centered at $(-2, 0)$, D_1^+ is centered at $(2, 0)$, and D_k^0 touches D_1^+ from the outside. Since the radius of D_1^- and D_1^+ is the same, the locus b_{11} of the centers of disks that simultaneously touch D_1^- and D_1^+ from the outside is the bisector of their centers, i.e., b_{11} is y-axis. Fix a value k. It is easy to see that the disk D_{11k}^* centered at $(0, 2\tan(k\theta))$ with the radius $\frac{2}{\cos(k\theta)} - 1$ touches D_1^- and D_1^+ from the outside and D_k^0 from the inside. Furthermore, we show that D_{11k}^* does not contain disks in $\mathcal{D}^0$ in its interior (obvious for $\mathcal{D}^- \cup \mathcal{D}^+$). Since every disk in $\mathcal{D}^0$ touches D_1^+ from the outside, D_{11k}^* containing a disk of $\mathcal{D}^0$ in its interior would imply that D_{11k}^* intersects the interior of D_1^+, a contradiction.

Next, we show that it holds for $1 < i, j \le m$ and $1 \le k \le m$. The key idea is that D_i^- (resp. D_j^+) got placed by translating ("perturbing") D_1^- (resp. D_1^+) so little that the disk D_{ijk}^* touching D_i^- and D_j^+ from the outside and D_k^0 from the inside does not contain any disk of $\mathcal{D}^- \cup \mathcal{D}^+ \cup \mathcal{D}^0$ in its interior as for the case when $i = j = 1$. We argue this using some elementary geometry. See Figure 6(b). Let v_{ijk} and v'_{ijk} be the two intersection points of ∂D_{ijk}^* and ∂D_1^+, for $1 \le i, j, k \le m$. Such two intersection points coincide with each other when $i = j = 1$, and furthermore, they always exist due to our construction. Note that v_{11k} is also the intersection point of ∂D_k^0 and ∂D_1^+. It is trivial to see that as the parameter ω gets smaller, v_{ijk} and v'_{ijk} lie closer to v_{11k} along ∂D_1^+. They all coincide with v_{11k} when ω becomes 0. Since ω is a sufficiently small positive number, we are assured that v_{ijk} lies between $v_{11(k-1)}$ and v_{11k} along ∂D_1^+, i.e., D_{ijk}^* does not contain D_{k-1}^0, not to mention $D_1^0, \ldots, D_{k-2}^0$. Similarly, D_{ijk}^* does not contain $D_{k+1}^0, \ldots, D_m^0$. Moreover, D_{ijk}^* does not contain any disk of $\mathcal{D}^- \cup \mathcal{D}^+$. Hence, there is a disk touching D_i^- and D_j^+ from the outside and D_k^0 from the inside and not containing any disk of $\mathcal{D}^- \cup \mathcal{D}^+ \cup \mathcal{D}^0$ in its interior, for $1 \le i, j, k \le m$.

This proves our claim, and finishes our $\Omega(n^3)$ lower bound construction when the disks have the same radius. ■

We prove a refined bound on the complexity of $\mathcal{V}_{\neq 0}(\mathcal{P})$ if the uncertainty regions in $\mathcal{D}$ are pairwise-disjoint disks and the ratio of the radii of the largest to the smallest disk is bounded by λ.

Lemma 2.9. *If $\mathcal{P} = \{P_1, \ldots, P_n\}$ is a set of n uncertain points in $\mathbb{R}^2$ whose uncertainty regions are pairwise-disjoint disks with radii in the range $[1, \lambda]$, a pair of curves in Γ intersects in $O(\lambda)$ points.*

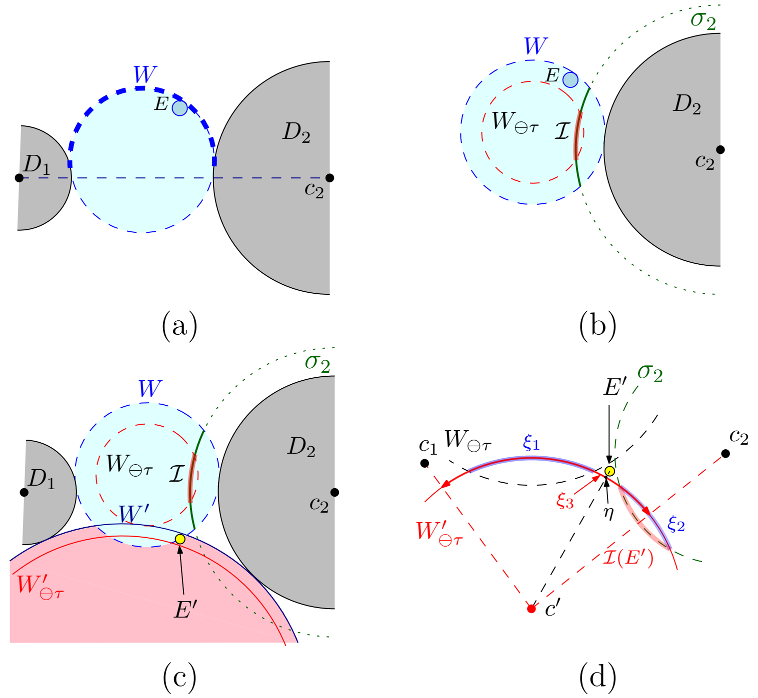


Figure 7: An illustration for the proof of Lemma 2.9.

Proof: Fix a pair of curves γ_1 and γ_2 in Γ. Let D_1 and D_2 be the disks corresponding to γ_1 and γ_2, and let c_1 and c_2 be their centers, respectively. By applying rotation and translation to the plane, we can assume D_1 and D_2 are centered on the x-axis, with D_1 to the left of D_2.

For a parameter t, $1 \le t \le \lambda$, let $\mathcal{D}$ denote the set of all the disks associated with $\mathcal{P}$, excluding D_1 and D_2, with radii between t and $2t$. An intersection point $q \in \gamma_1 \cap \gamma_2$ corresponds to a *witness* disk W centered at q that touches both D_1 and D_2 from the outside, touches exactly one other disk $E \in \mathcal{D}$ from the inside, and properly contains no disks of $\mathcal{D}$. The family of disks that touch both D_1 and D_2 from the outside is a *pencil*, which sweeps over a portion of the plane as the tangency points with D_1 and D_2 move continuously and monotonically in the y-direction. A disk of $\mathcal{D}$ can contribute at most two intersection points to $\gamma_1 \cap \gamma_2$, as its boundary gets swept over at most twice by the circles of the pencil.

We break ∂W into two curves, *top* and *bottom*, at W's tangency points with D_1 and D_2. For a disk $E \in \mathcal{D}$, if its tangency point with its witness disk W is on the top portion of W, then it is a *top tangency event*, otherwise it is a *bottom tangency event*. See Figure 7(a). Let $\mathcal{D}_1$ (resp. $\mathcal{D}_2$) be the set of disks in $\mathcal{D}$ that are closer to D_1 (resp. D_2).

Below we show that the number of top tangency events involving disks in $\mathcal{D}_2$ is $O(\lambda/t)$. Other tangency events are handled by a symmetric argument.

We remove from $\mathcal{D}_2$ all the disks within distance $T = \xi t$ from D_2, where ξ is a sufficiently large constant. The ring with outer radius $\mathsf{r}(D_2) + 4T$ and inner radius $\mathsf{r}(D_2)$ has area

$$\alpha = \pi\big((\mathsf{r}(D_2) + 4T)^2 - (\mathsf{r}(D_2))^2\big) = O(T\mathsf{r}(D_2) + T^2) = O(t^2 + \lambda t),$$

as $\mathsf{r}(D_2) \le \lambda$. Disks removed from $\mathcal{D}_2$ have the following properties:

(i) they are interior-disjoint,

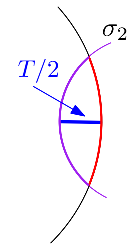
(ii) their radii lie in the interval $[t, 2t]$,

(iii) they are contained in the aforementioned ring, and

(iv) the area of each such disk is at least πt^2.

Hence, the number of removed disks is $O((t^2 + \lambda t)/t^2) = O(\lambda/t)$.

Consider the circle σ_2 of radius $\mathsf{r}(D_2) + T/2$ centered at c_2. Consider any disk $E \in \mathcal{D}_2$ and its witness disk W touching both D_1 and D_2 from the outside. If E has not been removed from $\mathcal{D}_2$, then $\mathsf{r}(W) \geq (T + 2t)/2$; in particular it is larger than $T/2$ and the center of W lies outside σ_2. Let $W_{\ominus\tau}$ be the disk concentric with W with radius $\mathsf{r}(W) - \tau$, where $\tau = 4t$. The interior of $W_{\ominus\tau}$ is disjoint from all disks in $\mathcal{D}_2$, as E touches W from inside and W does not fully contain any other disks from $\mathcal{D}_2$.



The witness disk W covers an arc of length at least $T/2$ on σ_2. Indeed, neither of these two disks contains the center of the other, and the inner distance between the two intersection arcs is $T/2$, see figure on the right. Similarly, let $\mathcal{I}(E)$ be the arc $W_{\ominus\tau} \cap \sigma_2$. By the same argument, we have that $\mathcal{I}(E)$ is of length at least $T/2 - \tau = \Omega(t)$.

The circumference of σ_2 is $2\pi(\mathsf{r}(D_2) + T/2) = O(\lambda)$, so if the arcs $\mathcal{I}(E)$, for $E \in \mathcal{D}_2$, are pairwise disjoint, we are done, as this implies that there could be at most $\lambda/(T/2 - \tau) = O(\lambda/t)$ such arcs and thus the size of the original $\mathcal{D}_2$, including the disks that were deleted from $\mathcal{D}_2$ is $O(\lambda/t)$. See Figure 7(b).

We now prove the claim that for any two disks $E, E' \in \mathcal{D}_2$ realizing a top tangency event, $\mathcal{I}(E)$ and $\mathcal{I}(E')$ are disjoint.

Let W (resp. W') be the witness disk that is tangent to D_1, D_2 and E (resp. E'). Assume that the tangency of W with D_2 is clockwise to the tangency of W' with D_2 (i.e., E is "above" E'). If $W_{\ominus\tau}$ and $W'_{\ominus\tau}$ are disjoint then the corresponding arcs $\mathcal{I}(E)$ and $\mathcal{I}(E')$ are obviously disjoint, so assume that $W_{\ominus\tau}$ and $W'_{\ominus\tau}$ intersect; see Figure 7(c).

Let c' be the center of $W'_{\ominus\tau}$. We define three circular arcs on $\partial W'_{\ominus\tau}$. Let $\xi_1 = \partial W'_{\ominus\tau} \cap W_{\ominus\tau}$, let ξ_2 be the portion of $\partial W'_{\ominus\tau}$ lying in the disk bounded by σ_2, and let ξ_3 be the portion of $\partial W'_{\ominus\tau}$ lying in the wedge formed by the rays $c'c_1$ and $c'c_2$; see Figure 7(d). It can be verified that $\xi_1 \subset \xi_3$ and the right endpoint of ξ_3 lies inside σ_2 and thus on ξ_2.

Next, let $\eta \in \partial W'_{\ominus\tau}$ be the intersection point of $\partial W'_{\ominus\tau}$ with the segment connecting c' and the center of E'; since E' lies in the exterior of $W'_{\ominus\tau}$, η exists. Since E' realizes a top tangency event, $\eta \in \xi_3$. Furthermore, E' lies in the exterior of $W_{\ominus\tau}$ and $E' \in \mathcal{D}_2$, which implies that $\eta \notin \xi_1$ and it lies to the right of ξ_1. Similarly, E' lies in the exterior of σ_2 and the right endpoint of ξ_3 lies on ξ_2, therefore η lies to the left of the arc ξ_2. In other words, η separates ξ_1 and ξ_2, implying that $\xi_1 \cap \xi_2 = \emptyset$, which in turn implies that the top endpoint of $\mathcal{I}(E')$ does not lie inside $W_{\ominus\tau}$. Hence, $\mathcal{I}(E) \cap \mathcal{I}(E') = \emptyset$, as claimed.

Finally, We repeat the above counting argument, for $t = 1, 2, 4, \ldots, 2^m$, where $m = \lceil \log_2 \lambda \rceil$, concluding that the number of intersection points between γ_1 and γ_2 is bounded by $\sum_{i=1}^{m} O(\lambda/2^i) = O(\lambda)$. This completes the proof of the lemma. ∎

Theorem 2.10. *Let $\mathcal{P} = \{P_1, \ldots, P_n\}$ be a set of n uncertain points in $\mathbb{R}^2$ such that their uncertainty regions are pairwise-disjoint disks and that the ratio of the largest and the smallest radii of the disks is at most λ. Then, the complexity of $\mathcal{V}_{\neq 0}(\mathcal{P})$ is $O(\lambda n^2)$, and it can be computed in $O(n^2 \log n + \mu)$ expected time, where μ is the complexity of $\mathcal{V}_{\neq 0}(\mathcal{P})$. Furthermore, there exists such a set $\mathcal{P}$ of uncertain points for which $\mathcal{V}_{\neq 0}(\mathcal{P})$ has $\Omega(n^2)$ complexity.*

Proof: The upper bound on the complexity of $\mathcal{V}_{\neq 0}(\mathcal{P})$ follows from Lemma 2.9. By the same argument as in the proof of Theorem 2.5, $\mathcal{V}_{\neq 0}(\mathcal{P})$ can be computed in $O(n^2 \log n + \mu)$ time, where μ is the number of vertices in $\mathcal{V}_{\neq 0}(\mathcal{P})$.

Next we show that there exists a set $\mathcal{P}$ of n uncertain points in $\mathbb{R}^2$ such that $\mathcal{V}_{\neq 0}(\mathcal{P})$ has $\Omega(n^2)$ vertices. Assume that $n = 2m$ for some positive integer m. All the disks D_i have the same radius 1, centered at $c_i = (4(i-m)-2, 0)$, for $1 \leq i \leq 2m$. Any pair (P_i, P_j) satisfying that $j-i \geq 2$ and $j+i$ is even determines 2 vertices: $v_1 = (2(i+j-2m-1), (j-i)^2-1)$, and $v_2 = (2(i+j-2m-1), 1-(j-i)^2)$, of $\mathcal{V}_{\neq 0}$ (realized

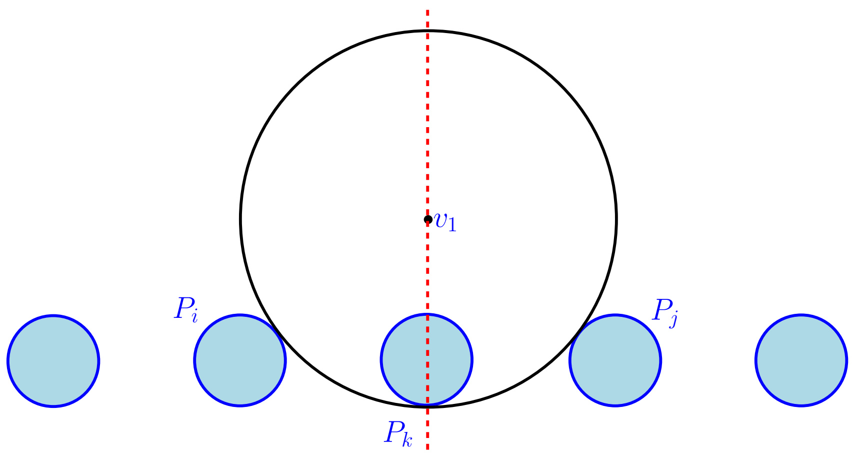


Figure 8: Any pair (P_i, P_j) satisfying $j - i \geq 2$ determines 2 vertices of $\mathcal{V}_{\neq 0}$. Only the vertex v_1 is shown.

with P_k, $k = \frac{j+i}{2}$) (Figure 8). Any pair (P_i, P_j) satisfying that $j - i \geq 2$ and $j + i$ is odd determines 2 vertices: $v_1 = (2(i+j-2m-1), (j-i)\sqrt{(j-i)^2 - 4})$, and $v_2 = (2(i+j-2m-1), (i-j)\sqrt{(j-i)^2 - 4})$, of $\mathcal{V}_{\neq 0}$ (realized with P_k, $k = \lfloor \frac{j+i}{2} \rfloor$ or $k = \lceil \frac{j+i}{2} \rceil$). One can verify that $\delta_i(v) = \delta_j(v) = \Delta_k(v) \leq \Delta_l(v)$, for $1 \leq l \leq n$, $v \in \{v_1, v_2\}$. Hence, we obtain a lower bound of $\Omega(n^2)$ for the complexity of $\mathcal{V}_{\neq 0}$. ■

Remarks. We note that the proof of Lemma 2.9 is essentially a packing argument, and therefore can be extended to the case when each uncertainty region is a convex α-fat semialgebraic set of constant description complexity. A convex set C is called *α-fat*, if there exist two concentric disks D and D' so that $D \subseteq C \subseteq D'$ and the ratio between the radii of D' and D is at most α. The constant of proportionality also depends on α and the description complexity of the sets defining the uncertainty regions. This in turn implies that $\mathcal{V}_{\neq 0}(\mathcal{P})$ has $O(\lambda n^2)$ complexity if the uncertainty regions of $\mathcal{P}$ are pairwise-disjoint convex α-fat sets, for some constant $\alpha \geq 1$, and the ratio of the size of the largest to the smallest region is bounded by λ. Extension of the proof of Lemma 2.9 to this case, however, is even more technical, so we have decided not to state this generalized result as a theorem, especially since, in practice, a fat convex set can be approximated by a circular disk.

Storing $\mathcal{P}_\phi$'s for $\mathcal{V}_{\neq 0}(\mathcal{P})$. We store the index i of each uncertain point P_i instead of P_i itself. If we store $\mathcal{P}_\phi$ for each cell ϕ of $\mathcal{V}_{\neq 0}(\mathcal{P})$ explicitly, the size increases by a factor of n. However, we observe that for two adjacent cells ϕ, ϕ' of $\mathcal{V}_{\neq 0}(\mathcal{P})$, i.e., two cells that share a common edge, $|\mathcal{P}_\phi \oplus \mathcal{P}_{\phi'}| = 1$, where $\oplus$ denotes the symmetric difference of two sets. Therefore, using a persistent data structure [DSST89], we can store $\mathcal{P}_\phi$ for all cells of $\mathcal{V}_{\neq 0}(\mathcal{P})$ in $O(\mu)$ space, where μ is the complexity of $\mathcal{V}_{\neq 0}(\mathcal{P})$, so that for any cell ϕ, $\mathcal{P}_\phi$ can be retrieved in $O(\log n + |\mathcal{P}_\phi|)$ time.[3] By combining this with a planar point-location data structure [dBCKO08], we obtain the following:

Theorem 2.11. *Let $\mathcal{P}$ be a set of n uncertain points in $\mathbb{R}^2$, and let μ be the complexity of $\mathcal{V}_{\neq 0}(\mathcal{P})$. Then, $\mathcal{V}_{\neq 0}(\mathcal{P})$ can be preprocessed in $O(\mu \log \mu)$ time into a data structure of size $O(\mu)$ so that, for a query point $q \in \mathbb{R}^2$, $\mathsf{NN}_{\neq 0}(q, \mathcal{P})$ can be computed in $O(\log n + t)$ time, where t is the output size.*

[3]If the curves of Γ intersect transversally at every vertex, it suffices to store $\mathcal{P}_\phi$ for each cell of $\mathcal{V}_{\neq 0}(\mathcal{P})$. Otherwise one may have to store $\mathcal{P}_\phi$ for edges and vertices of $\mathcal{V}_{\neq 0}(\mathcal{P})$. This does not affect the asymptotic performance of the data structure.

2.2 Discrete case

We now analyze the complexity of $\mathcal{V}_{\neq 0}(\mathcal{P})$ when the distribution of each point P_i in $\mathcal{P}$ is discrete. Let $P_i = \{p_{i1}, \ldots, p_{ik}\}$. For $1 \leq j \leq k$, let $w_{ij} = \Pr[P_i \text{ is } p_{ij}]$. As in the previous section, for a point x, let

$$\Delta_i(q) = \max_{1\leq j\leq k} d(q, p_{ij}) \quad \text{and} \quad \delta_i(q) = \min_{1\leq j\leq k} d(q, p_{ij}).$$

Note that the projection of the graph of Δ_i (resp. δ_i) onto the xy-plane is the farthest-point (resp. nearest-point) Voronoi diagram of P_i. Let $\Delta(q) = \min_{1\leq i\leq n} \Delta_i(q)$. For each i, let $\gamma_i = \{x \in \mathbb{R}^2 \mid \delta_i(x) = \Delta(x)\}$, and set $\Gamma = \{\gamma_1, \ldots, \gamma_n\}$. Then $\mathcal{V}_{\neq 0}(\mathcal{P})$ is the planar subdivision $\mathcal{A}(\Gamma)$ induced by Γ (cf. Corollary 2.4).

We define a few functions that will help analyze the structure of $\mathcal{V}_{\neq 0}(\mathcal{P})$. We first define a function $f : \mathbb{R}^2 \times \mathbb{R}^2 \to \mathbb{R}$ as

$$f(x, p) = d^2(x, p) - \|x\|^2 = \|p\|^2 - 2\langle x, p\rangle. \tag{5}$$

For $1 \leq i \leq n$, define

$$\varphi_i(x) = \min_{1\leq j\leq k} f(x, p_{ij}) \quad \text{and} \quad \Phi_i(x) = \max_{1\leq j\leq k} f(x, p_{ij}).$$

Finally, we define

$$\Phi(x) = \min_{1\leq i\leq n} \Phi_i(x).$$

The following lemma is straightforward.

Lemma 2.12. *For any $i \leq n$ and for any $q \in \mathbb{R}^2$, $\delta_i(q) = r$ if and only if $\varphi_i(q) = r^2 - \|q\|^2$.*

Lemma 2.13. *For any pair i, j, $1 \leq i \neq j \leq n$, let $\gamma_{ij} = \{x \in \mathbb{R}^2 \mid \delta_i(x) = \Delta_j(x)\}$, then γ_{ij} is a convex polygonal curve with $O(k)$ vertices.*

Proof: By Lemma 2.12, for any pair i, j and for any $x \in \mathbb{R}^2$, $\delta_i(x) = \Delta_j(x)$ if and only if $\varphi_i(x) = \Phi_j(x)$. Hence, γ_{ij} is also the zero set of the function $\Phi_j(x) - \varphi_i(x)$.

Φ_j is the upper envelope of k linear functions, and thus is a piecewise-linear convex function. Similarly, φ_i, the lower envelope of k linear functions, is a piecewise-linear concave function. Hence, $\Phi_j(x) - \varphi_i(x)$ is a piecewise-linear convex function, which implies that $\gamma_{ij} = \{x \in \mathbb{R}^2 \mid \Phi_j(x) = \varphi_i(x)\}$ is a convex polygonal curve. Since γ_{ij} is the projection of the intersection curve of the graphs of Φ_j and φ_i, each of which is the surface of an unbounded convex polyhedron with at most k faces, γ_{ij} has $O(k)$ vertices. ∎

Theorem 2.14. *Let $\mathcal{P} = \{P_1, \ldots, P_n\}$ be a set of n uncertain points in $\mathbb{R}^2$, where each P_i has a discrete distribution of size at most k. The complexity of $\mathcal{V}_{\neq 0}(\mathcal{P})$ is $\mu = O(kn^3)$, and it can be computed in $O(n^2 \log n + \mu)$ expected time. Furthermore, it can be preprocessed in additional $O(\mu)$ time into a data structure of size $O(\mu)$ so that an $\mathsf{NN}_{\neq 0}(q)$ query can be answered in $O(\log \mu + t)$, where t is the output size.*

Proof: We follow the same argument as in the proof of Theorem 2.5. We need to bound the number of intersection points between a pair of curves γ_i and γ_j. Fix an index u. Let $\gamma_{iu} = \{x \in \mathbb{R}^2 \mid \delta_i(x) = \Delta_u(x)\}$ and $\gamma_{ju} = \{x \in \mathbb{R}^2 \mid \delta_j(x) = \Delta_u(x)\}$. By Lemma 2.13, each of γ_{iu} and γ_{ju} is a convex polygonal curve in $\mathbb{R}^2$ with $O(k)$ vertices. Since two convex polygonal curves in general position with n_1 and n_2 vertices intersect in at most $n_1 + n_2$ points, γ_{iu} and γ_{ju} intersect at $O(k)$ points. Hence, γ_i and γ_j intersect at $O(kn)$ points, implying that $\mathcal{V}_{\neq 0}(\mathcal{P})$ has $O(kn^3)$ vertices. The running time follows from the proof of Theorem 2.5. ∎

3 Data structures for $\mathsf{NN}_{\neq 0}$ queries

With the maximum size of $\mathcal{V}_{\neq 0}$ being $\Theta(n^3)$, we present $O(n \operatorname{polylog}(n))$-size data structures that circumvent the need for constructing $\mathcal{V}_{\neq 0}(\mathcal{P})$ and answer $\mathsf{NN}_{\neq 0}$ queries in poly-logarithmic or sublinear time. They rely on geometric data structures for answering range-searching queries and their variants; see [Aga16] for a recent survey.

An $\mathsf{NN}_{\neq 0}(q)$ query is answered in two stages. The first stage computes $\Delta(q)$, and the second stage computes all points $P_i \in \mathcal{P}$ for which $\delta_i(q) < \Delta(q)$. We build a separate data structure for each stage. We first describe the one for the continuous case and then for the discrete case.

Continuous case. We assume that the uncertainty region of each point P_i is a disk D_i of radius r_i centered at c_i. Recall from Section 2 that the projection of the graph of the function Δ onto the xy-plane, a planar subdivision $\mathbb{M}$, is the (additive-weighted) Voronoi diagram of the points $c_1, \ldots, c_n$, and it has linear complexity. Hence $\mathbb{M}$ can be preprocessed in $O(n \log n)$ time into a data structure of size $O(n)$ so that for a query point $q \in \mathbb{R}^2$, $\Delta(q)$ can be computed in $O(\log n)$ time [dBCKO08].

Next we wish to report all points $P_i \in \mathcal{P}$ for which $\delta_i(q) < \Delta(q)$, i.e., for which D_i intersects the disk of radius $\Delta(q)$ centered at q. Note that the projection of the graph of the lower envelope of $\{\delta_1, \ldots, \delta_n\}$ is also an (additive-weighted) Voronoi diagram of the points $c_1, \ldots, c_n$ and has linear complexity. Recently [KMR+16] have described a data structure of size $O(n \operatorname{polylog}(n))$ that can answer the above query in $O(\log n + t)$ time, where t is the output size. It can be constructed in $O(n \operatorname{polylog}(n))$ randomized expected time. We thus obtain the following:

Theorem 3.1. *Let $\mathcal{P} = \{P_1, \ldots, P_n\}$ be a set of n uncertain points in $\mathbb{R}^2$ so that the uncertainty region of each P_i is a disk. $\mathcal{P}$ can be preprocessed into a data structure of size $O(n \operatorname{polylog}(n))$, so that an $\mathsf{NN}_{\neq 0}(q)$ query can be answered in $O(\log n + t)$ time, where t is the output size. The data structure can be constructed in $O(n \operatorname{polylog}(n))$ randomized expected time.*

Remarks. (i) Note that Theorem 3.1 gives a better result than Theorem 2.11 but the data structure based on $\mathcal{V}_{\neq 0}(\mathcal{P})$ is simpler and more practical.

(ii) If we use L_1 or L_∞ metric to compute the distance between points and use disks in L_1 or L_∞ metric (i.e., a diamond or a square), then an $\mathsf{NN}_{\neq 0}(q)$ query can be answered in $O(\log^2 n + t)$ time using $O(n \log^2 n)$ space: the first stage remains the same and the second stage reduces to reporting a set of axis-aligned squares that intersect a query axis-aligned square [Aga16].

Discrete case. Next, we consider the case when each P_i has a discrete distribution of size at most k; set $N = nk$. The functions Δ_i and δ_i are now more complex and thus the data structure for $\mathsf{NN}_{\neq 0}(q)$ queries is more involved. As in Section 2.2, instead of working with the functions δ_i and Δ_i, we work with φ_i and Φ_i. By Lemma 2.12, the problem of reporting all points P_i with $\delta_i(q) < \Delta(q)$ is equivalent to returning the points with $\varphi_i(q) < \Phi(q)$. As for the continuous case, we construct two data structures—the first one computes $\Phi(q)$ for a query point $q \in \mathbb{R}^2$ and the second one reports all P_i's with $\varphi_i(q) < \Phi(q)$. Note that $\varphi_i(q) < \Phi(q)$ if and only if the point $\widehat{q} = (q, \Phi(q)) \in \mathbb{R}^3$ lies above the graph of φ_i. By triangulating each face of φ_i and Φ_i if necessary, we can assume that each φ_i is a triangulated concave surface and each Φ_i is a triangulated convex surface.

We now describe the data structure for computing $\Phi(q)$. Note that $\Phi(q) = \Phi_j(q)$ if Φ_j is the first surface in the set $\{\Phi_1, \ldots, \Phi_n\}$ intersected by ℓ_q, the vertical line passing through q, in the $(+z)$-direction. We first construct a 3-level partition tree [Aga16] on the set of triangles in the graphs of $\Phi_1, \ldots, \Phi_n$, denoted by Σ, so that the triangles of Σ intersected by ℓ_q, for a query point $q \in \mathbb{R}^2$, can be reported efficiently. The partition tree stores a family of *canonical* subsets of triangles in Σ so that for any query point q, the triangles of Σ intersected by ℓ_q can be reported as the union of $O(\sqrt{N}\log^2 n)$ canonical subsets in $O(\sqrt{N}\log^2 n)$ time. Let F_q denote the family of canonical subsets reported by the query procedure. The size of the data structure is $O(N\log^2 n)$, and it can be constructed in $O(N\log^2 N)$ randomized expected time [Aga16]. Next, for each canonical subset C, let C^* be the set of planes supporting the triangles in C. We construct the lower envelope L_C of C^* (by regarding each plane in C^* as the graph of a linear function), which has size $O(|C|)$, and preprocess L_C into an $O(|C|)$-size data structure so that for a query point $q \in \mathbb{R}^2$, $L_C(q)$ can be computed in $O(\log|C|)$ time [SA95]. Summing over all canonical subsets of the partition tree, the overall size of the data structure is $O(N\log^2 N)$ and it can be constructed in $O(N\log^3 N)$ randomized expected time.

Given a query point $q \in \mathbb{R}^2$, we first query the partition tree and compute the family F_q of canonical subsets. For each canonical set $C \in F_q$, we compute $L_C(q)$ and return the minimum among them as $\Phi(q)$. Since, the procedure spends $O(\log N)$ time for each canonical subset, the overall query time is $O(\sqrt{N}\log^3 N)$. The correctness of the procedure follows from the following observation: ℓ_q intersects all triangles of a canonical subset $C \in F_q$, so for each triangle $\tau \in C$ and its supporting plane τ^*, $\ell_q \cap \tau = \ell_q \cap \tau^*$. Therefore $L_C(q)$ is the same as the (height of the) first intersection point of ℓ_q with a triangle of C, and $\Phi(q) = \min_{C\in F_q} L_C(q)$.

Next, we describe the data structure for reporting the points P_i with $\varphi_i(q) < \Phi(q)$. It is very similar to the one just described, except one twist. First, as above, we construct a 3-level partition tree on the triangles in the graphs of $\varphi_1, \ldots, \varphi_n$. Let C be a canonical subset constructed by the partition tree, and let C^* be the set of planes supporting C. Using a result by [AC09] (see also [Aga16]), we preprocess C^*, in $O(|C|\log|C|)$ randomized expected time, into a data structure of size $O(|C^*|)$ so that for a query point $\widehat{q} = (q, \Phi(q))$, all t_C planes of C^* lying below $\hat{q}$ can be reported in $O(\log N + t_C)$ time. Summing over all canonical subsets of the partition tree, the overall size of the data structure is $O(N\log^2 N)$, and it can be constructed in $O(N\log^3 N)$ randomized expected time.

Given a query point $q \in \mathbb{R}^2$, we first query the partition tree and compute the family F_q of canonical subsets. For each canonical set $C \in F_q$, we next report all planes of C^* lying below $\hat{q}$. The overall query time is $O(\sqrt{N}\log^3 N + t)$, where t is the output size.[4] The correctness of the procedure follows from the same argument as above, namely, since ℓ_q intersects all triangles of a canonical subset $C \in F_q$, a triangle in C lies below $\widehat{q}$ if and only if the plane supporting it lies below $\widehat{q}$.

Putting everything together, we can construct, in $O(N\log^3 N)$ randomized expected time, a data structure of $O(N\log^2 N)$ size that can answer an $\mathsf{NN}_{\neq 0}$ query in $O(\sqrt{N}\log^3 N)$ time.

Finally, we remark that the 3-level partition tree can be replaced by a multi-level data structure of size $O(N^2\log^2 N)$ so that the set of triangles intersected by ℓ_q can be returned as the union of $O(\log^3 N)$ canonical subsets [Aga16]. Using this data structure, we can answer an $\mathsf{NN}_{\neq 0}$ query in $O(\log^4 N)$ time using $O(N^2\log^2 N)$ space. We thus obtain the following:

Theorem 3.2. *Let $\mathcal{P}$ be a set of n uncertain points in $\mathbb{R}^2$, each with a discrete distribution of size at most k; set $N = nk$. $\mathcal{P}$ can be preprocessed into a data structure of size $O(N\log^3 N)$ so that an $\mathsf{NN}_{\neq 0}(q)$ query can be answered in $O(\sqrt{N}\log^3 N + t)$ time, or into a data structure of size $O(N^2\log^2 N)$ with $O(\log^4 N + t)$ query time, where t is the output size. The expected preprocessing times are $O(N\log^3 N)$ and $O(N^2\log^3 N)$ time, respectively.*

[4] We note that if ℓ_q passes through the boundary of a triangle of some φ_i, then P_i may be reported multiple times. If the points of P_i are in general position, then the degree of each vertex of φ_i is constant, so P_i will be reported $O(1)$ times. However if points in P_i are in a degenerate position, then additional care is needed, using standard techniques such as symbolic perturbation, to ensure that P_i is reported only $O(1)$ times.

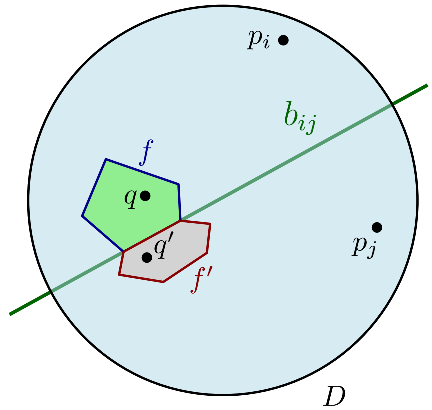


Figure 9: An illustration of the proof. Inside the unit disk D, two adjacent faces f and f' share a portion of the bisector b_{ij} defined by p_i and p_j.

4 Quantification probabilities

We now turn our attention to the second part of answering probabilistic NN queries, namely, returning the quantification probabilities that are positive. We begin with a data structure for computing quantification probabilities exactly for the case when each uncertain point has a discrete distribution of size at most k. Since computing these quantities exactly is quite expensive and they are small for most of the points, we focus on computing quantification probabilities approximately.

4.1 The exact algorithm

Assuming each point in $\mathcal{P}$ has a discrete distribution of size at most k, we build the *probabilistic Voronoi diagram* $\mathcal{V}_{\mathrm{Pr}}(\mathcal{P})$ that decomposes $\mathbb{R}^2$ into a set of cells, so that any point q in a cell has the same $\pi_i(q)$ value for all $P_i \in \mathcal{P}$; that is, for any point q in this cell, we know exactly the probability of each point $P \in \mathcal{P}$ being the NN of q.

Lemma 4.1. *Let $\mathcal{P}$ be a set of n uncertain points in $\mathbb{R}^2$, each with a discrete distribution of size at most k; set $N = nk$. The complexity of $\mathcal{V}_{\mathrm{Pr}}(\mathcal{P})$ is $O(N^4)$. Moreover, there exists a set $\mathcal{P}$ of n uncertain points in $\mathbb{R}^2$ with $k = 2$ such that $\mathcal{V}_{\mathrm{Pr}}(\mathcal{P})$ has size $\Omega(n^4)$.*

Proof: We first prove the upper bound. There are N possible locations. Each pair of possible locations determines a bisector, resulting in $O(N^2)$ bisectors. These bisectors partition the plane into $O(N^4)$ convex cells so that the order of all distances to each of the nk possible locations, and thus by Eq. (2) also all the quantification probabilities, are preserved within each cell. Therefore, the resulting planar subdivision is a refinement of $\mathcal{V}_{\mathrm{Pr}}(\mathcal{P})$, and thus $O(n^4k^4)$ is an upper bound on the complexity of $\mathcal{V}_{\mathrm{Pr}}(\mathcal{P})$.

Next, we show that there exists a set $\mathcal{P}$ of n uncertain points in $\mathbb{R}^2$ with $k = 2$ such that $\mathcal{V}_{\mathrm{Pr}}(\mathcal{P})$ has size $\Omega(n^4)$. For simplicity, we describe a degenerate configuration of points, but the argument can be generalized to a non-degenerate configuration as well, by being more careful. For every $1 \le i \le n$, $P_i \in \mathcal{P}$ has two possible locations p_i and p_i', each with probability 0.5. Let D be the unit disk centered at the origin. We choose $p_1, \ldots, p_n$ inside D so that the bisector b_{ij} of every pair p_i, p_j, for $i < j$, is a distinct line and all pairs of bisectors intersect inside D. We place all p_i''s at the same location far away from D, say, at $\overline{p} = (100, 0)$. Note that the bisector of $\overline{p}$ and p_i, for any $i \le n$, does not intersect D, so for any point $q \in D$, $d(p_i, q) < d(\overline{p}, q)$.

Let $\mathcal{A}$ be the arrangement of the bisectors $\{b_{ij} \mid 1 \le i < j \le n\}$. Since all pairs of bisectors intersect inside D, $\mathcal{A} \cap D$ has $\Theta(n^4)$ faces. Let f, f' be any two adjacent faces of $\mathcal{A}$ inside D, let b_{ij} be the bisector separating f and f', and let q, q' be arbitrary points in the interior of f, f', respectively. Without loss of generality, assume that $d(p_i, q) < d(p_j, q)$, then $d(p_i, q') > d(p_j, q')$. See Figure 9. Suppose there are r, $0 \le r < n-1$, points of $\{p_1, \ldots, p_n\}$ that are closer to q than p_i, i.e., p_i (resp. p_j) is the $(r+1)$-st NN of q (resp. q') among $\{p_1, \ldots, p_n\}$. Then, by Eq. (2),

$$\pi_i(q) = 0.5 \cdot (1-0.5)^r + 0.5 \cdot (1-0.5)^{n-1} = 0.5^{r+1} + 0.5^n, \text{ and}$$
$$\pi_j(q) = 0.5 \cdot (1-0.5)^{r+1} + 0.5 \cdot (1-0.5)^{n-1} = 0.5^{r+2} + 0.5^n.$$

Symmetrically, $\pi_i(q') = 0.5^{r+2} + 0.5^n$ and $\pi_j(q') = 0.5^{r+1} + 0.5^n$. In particular $\pi_i(q) \ne \pi_i(q')$ and $\pi_j(q) \ne \pi_j(q')$. In other words, any two adjacent faces of $\mathcal{A}$ inside D have distinct quantification probability vectors, implying that $\mathcal{V}_{\Pr}(\mathcal{P})$ has $\Omega(n^4)$ complexity. ∎

As in Section 2.1, we can store the quantification probabilities for all faces of $\mathcal{V}_{\Pr}(\mathcal{P})$ by using $O(1)$ storage per face. Hence, by preprocessing $\mathcal{V}_{\Pr}(\mathcal{P})$ for point-location queries, for a query point q, we can report all t quantification probabilities that are positive in $O(\log N + t)$ time.

Theorem 4.2. *Let $\mathcal{P}$ be a set of n uncertain points in $\mathbb{R}^2$, each with a discrete distribution of size at most k; set $N = nk$. $\mathcal{P}$ can be preprocessed in time $O(N^4 \log N)$ time into a data structure of size $O(N^4)$ that can report all t positive quantification probabilities of a query point in time $O(\log N + t)$.*

4.2 A Monte-Carlo algorithm

In this section we describe a simple Monte-Carlo approach to build a data structure for quickly computing $\widehat{\pi}_i(q)$ for all P_i for any query point q, which approximates the quantification probability $\pi_i(q)$. For a fixed value s, to be specified later, the preprocessing step works in s rounds. In the j-th round the algorithm creates a sample $R_j = \{r_{j1}, r_{j2}, \ldots, r_{jn}\} \subseteq \mathbb{R}^2$ by choosing each r_{ji} using the distribution of P_i. For each $j \in \{1, \ldots, s\}$, we construct the Voronoi diagram $\mathsf{Vor}(R_j)$ in $O(n \log n)$ time and preprocess it for point-location queries in additional $O(n \log n)$ time.

To estimate quantification probabilities of a query q, we initialize a counter $c_i = 0$ for each point P_i. For each $j \in \{1, \ldots, s\}$, we find the point $r_{ji} \in R_j$ whose cell in $\mathsf{Vor}(R_j)$ contains the query point q, and increment c_i by 1. Finally we estimate $\widehat{\pi}_i(q) = c_i/s$. Note that at most s distinct c_i's have nonzero values, so we can implicitly set the remaining $\widehat{\pi}_i(q)$'s to 0.

Discrete case. If each $P_i \in \mathcal{P}$ has a discrete distribution of size k, then this algorithm can be implemented very efficiently. Each r_{ji} can be selected in $O(\log k)$ time after preprocessing each P_i, in $O(k)$ time, into a balanced binary tree [MR95]. Thus, total preprocessing takes $O(s(n(\log n + \log k)) + nk) = O(nk + sn \log(nk))$ time and $O(sn)$ space, and each query takes $O(s \log n)$ time.

It remains to determine the value of s so that $|\pi_i(q) - \widehat{\pi}_i(q)| \le \varepsilon$ for all P_i and all queries q, with probability at least $1-\delta$. For fixed q, P_i, and instantiation R_j, let X_{ji} be the random indicator variable, which is 1 if r_{ji} is the NN of q and 0 otherwise. Since $\mathsf{E}[\mathrm{X}_{ji}] = \pi_i(q)$ and $\mathrm{X}_i \in \{0,1\}$, applying the Chernoff-Hoeffding bound [MR95] to

$$\widehat{\pi}_i(q) = \frac{c_i}{s} = \frac{1}{s}\sum_{j=1}^{s} \mathrm{X}_{ji},$$

we obtain that

$$\Pr\left[|\widehat{\pi}_i(q) - \pi_i(q)| \geq \varepsilon\right] \leq 2\exp(-2\varepsilon^2 s). \tag{6}$$

For each cell of $\mathcal{V}_{\Pr}(\mathcal{P})$, we choose one point, and let Q be the resulting set of points. If $|\widehat{\pi}_i(q)-\pi_i(q)| \leq \varepsilon$ for every point $q \in Q$, then $|\widehat{\pi}_i(q) - \pi_i(q)| \leq \varepsilon$ for every point $q \in \mathbb{R}^2$. Since there are n different values of i, by applying the union bound to Eq. (6), the probability that there exist a point $q \in \mathbb{R}^2$ and an index $i \in \{1, \ldots, n\}$ with $|\widehat{\pi}_i(q) - \pi_i(q)| \geq \varepsilon$ is at most $2n|Q|\exp(-2\varepsilon^2 s)$. Hence, by setting

$$s = \frac{1}{2\varepsilon^2} \ln \frac{2n|Q|}{\delta},$$

$|\widehat{\pi}_i(q)-\pi_i(q)| \leq \varepsilon$ for all $q \in \mathbb{R}^2$ and for all $i \in \{1, \ldots, n\}$, with probability at least $1-\delta$. By Lemma 4.1, $|Q| = O(n^4k^4)$, so we obtain the following result.

Theorem 4.3. *Let $\mathcal{P}$ be a set of n uncertain points in $\mathbb{R}^2$, each with a discrete distribution of size k, and let $\varepsilon, \delta \in (0, 1)$ be two parameters. $\mathcal{P}$ can be preprocessed, in*

$$O(nk + (n/\varepsilon^2)\log(nk)\log(nk/\delta))$$

time, into a data structure of size $O((n/\varepsilon^2)\log(nk/\delta))$, which computes, for any query point $q \in \mathbb{R}^2$, in $O((1/\varepsilon^2)\log(nk/\delta)\log n)$ time, a value $\widehat{\pi}_i(q)$ for every P_i such that $|\pi_i(q) - \widehat{\pi}_i(q)| \leq \varepsilon$ for all i with probability at least $1 - \delta$.

Continuous case. There are two technical issues in extending this technique and analysis to continuous distributions. First, how we instantiate a certain point r_i from each P_i. Herein we assume the representation of the pdf is such that this can be done in constant time for each P_i.

Second, we need to bound the number of distinct queries that need to be considered to apply the union bound as we did above. Since $\pi_i(q)$ may vary continuously with the query location, unlike the discrete case, we cannot hope for a bounded number of distinct results. However, we just need to define a finite set $\bar{Q}$ of query points so that for any point $q \in \mathbb{R}^2$, there is a point $q' \in \bar{Q}$ such that $\max_i |\pi_i(q) - \pi_i(q')| \leq \varepsilon/2$. Then, we can choose s large enough so that it permits at most $\varepsilon/2$ error on each query in $\bar{Q}$. Specifically, choosing $s = O((1/\varepsilon^2)\log(n|\bar{Q}|/\delta))$ is sufficient, so all that remains is to bound $|\bar{Q}|$.

To choose $\bar{Q}$, we show that each pdf of P_i can be approximated with a discrete distribution of size $O((n^2/\varepsilon^2)\log(n/\delta))$, and then reduce the problem to the discrete case.

For parameters $\alpha > 0$ and $\delta' \in (0, 1)$, set

$$k(\alpha) = \frac{c}{\alpha^2}\log\frac{1}{\delta'},$$

where c is a constant. For each $i \in \{1, \ldots, n\}$, we choose a random sample $\overline{P}_i \subset P_i$ of size $k(\alpha)$, according to the distribution defined by the location pdf f_i of P_i. We regard $\overline{P}_i$ as an uncertain point with uniform location probability. Set $\overline{\mathcal{P}} = \{\overline{P}_1, \ldots, \overline{P}_n\}$.

For a point $q \in \mathbb{R}^2$, let $\overline{G}_{q,i}$ denote the cdf of the distance between q and $\overline{P}_i$, i.e., $\overline{G}_{q,i}(r) = \Pr[d(q, \overline{P}_i) \leq r]$, or equivalently, it is the probability of $\overline{P}_i$ lying in the disk of radius r centered at q. A well-known result in the theory of random sampling [LLS01, VC71] implies that for all $q \in \mathbb{R}^2$ and $r \geq 0$,

$$\left|G_{q,i}(r) - \overline{G}_{q,i}(r)\right| \leq \alpha, \tag{7}$$

with probability at least $1 - \delta'$, provided that the constant c in $k(\alpha)$ is chosen sufficiently large.

Let $\overline{\pi}_i(q)$ denote the probability of $\overline{P}_i$ being the NN of q in $\overline{\mathcal{P}}$. We prove the following:

Lemma 4.4. *For any $q \in \mathbb{R}^2$ and for any fixed $i \in \{1, \ldots, n\}$,*

$$|\pi_i(q) - \overline{\pi}_i(q)| \leq \alpha n,$$

with probability at least $1 - \delta'$.

Proof: Recall that by Eq. (1),

$$\pi_i(q) = \int_0^\infty g_{q,i}(r) \prod_{j \neq i} (1 - G_{q,j}(r)) \mathrm{d}r.$$

Using Eq. (7), and the fact that $G_{q,j}(r), \overline{G}_{q,j}(r) \in [0,1]$ for all j, we obtain

$$\pi_i(q) \leq \int_0^\infty g_{q,i}(r) \prod_{j \neq i} (1 - \overline{G}_{q,j}(r)) \mathrm{d}r + (n-1)\alpha.$$

Note that $\prod_{j \neq i}(1 - \overline{G}_{q,j}(r))$ is the probability that the closest point of q in $\overline{\mathcal{P}} \setminus \{\overline{P}_i\}$ is at least distance r away from q. Let $h_{q,i}$ be the pdf of the distance between q and its closest point in $\overline{\mathcal{P}} \setminus \{\overline{P}_i\}$. Then

$$\prod_{j \neq i} (1 - \overline{G}_{q,j}(r)) = \int_r^\infty h_{q,i}(\theta) \mathrm{d}\theta.$$

Therefore

$$\pi_i(q) \leq \int_0^\infty \int_r^\infty g_{q,i}(r) h_{q,i}(\theta) \mathrm{d}\theta \mathrm{d}r + (n-1)\alpha.$$

By reversing the order of integration, we obtain

$$\begin{aligned}
\pi_i(q) &\leq \int_0^\infty \int_0^\theta h_{q,i}(\theta) g_{q,i}(r) \mathrm{d}r \mathrm{d}\theta + (n-1)\alpha = \int_0^\infty h_{q,i}(\theta) G_{q,i}(\theta) \mathrm{d}\theta + (n-1)\alpha \\
&\leq \int_0^\infty h_{q,i}(\theta)(\overline{G}_{q,i}(\theta) + \alpha) \mathrm{d}\theta + (n-1)\alpha \qquad \text{(using Eq. (7))} \\
&= \int_0^\infty h_{q,i}(\theta) \overline{G}_{q,i}(\theta) \mathrm{d}\theta + n\alpha = \overline{\pi}_i(q) + n\alpha.
\end{aligned}$$

A similar argument shows that $\pi_i(q) \geq \overline{\pi}_i(q) - n\alpha$. This completes the proof of the lemma. ∎

Thus, by setting $\alpha = \varepsilon/(2n)$, a random sample $\overline{P}_i$ of size $O((n^2/\varepsilon^2)\log(n/\delta))$ from each P_i ensures that

$$|\pi_i(q) - \overline{\pi}_i(q)| \leq \varepsilon/2 \tag{8}$$

for all queries. By choosing $\delta' = \delta/(2n)$, Eq. (8) holds for all $i \in \{1, \ldots, n\}$ with probability at least $1 - \delta/2$.

We consider $\mathcal{V}_{\mathrm{Pr}}(\overline{\mathcal{P}})$, choose one point from each of its cells, and set $\bar{Q}$ to be the resulting set of points. For a point $q \in \mathbb{R}^2$, let $\bar{q} \in \bar{Q}$ be the representative point of the cell of $\mathcal{V}_{\mathrm{Pr}}(\overline{\mathcal{P}})$ that contains q. Then, $|\pi_i(q) - \overline{\pi}_i(\bar{q})| < \varepsilon/2$ for all points $q \in \mathbb{R}^2$ and $i \in \{1, \ldots, n\}$, with probability at least $1 - \delta/2$.

Now applying the analysis for the discrete case to the point set $\overline{\mathcal{P}}$, if we choose

$$s = O\Big(\frac{1}{\varepsilon^2} \log \frac{n|\bar{Q}|}{\delta}\Big),$$

then $|\overline{\pi}_i(q) - \widehat{\pi}_i(q)| < \varepsilon$ for all points $q \in \mathbb{R}^2$ and for all $i \in \{1, \ldots, n\}$ with probability at least $1 - \delta/2$. Since

$$\left|\overline{P}_i\right| = k\Big(\frac{\varepsilon}{2n}\Big) = O\Big(\frac{n^2}{\varepsilon^2}\log\frac{n}{\delta}\Big),$$

by Lemma 4.1,

$$|\bar{Q}| = O\Big(n^4\Big(k\Big(\frac{\varepsilon}{2n}\Big)\Big)^4\Big) = O\Big(\frac{n^{12}}{\varepsilon^8}\log^4\frac{n}{\delta}\Big).$$

Putting everything together, we obtain the following.

Theorem 4.5. *Let $\mathcal{P} = \{P_1, \ldots, P_n\}$ be a set of n uncertain points in $\mathbb{R}^2$ so a random instantiation of P_i can be performed in $O(1)$ time, and let $\varepsilon, \delta \in (0,1)$ be two parameters. $\mathcal{P}$ can be preprocessed in $O((n/\varepsilon^2)\log(n/\varepsilon\delta)\log n)$ time into a data structure of size $O((n/\varepsilon^2)\log(n/\varepsilon\delta))$ that computes for any query point $q \in \mathbb{R}^2$, in $O((1/\varepsilon^2)\log(n/\varepsilon\delta)\log n)$ time, a value $\widehat{\pi}_i(q)$ for every P_i such that $|\pi_i(q) - \widehat{\pi}_i(q)| \leq \varepsilon$ for all i with probability at least $1 - \delta$.*

4.3 Spiral search algorithm

If the distribution of each point in $\mathcal{P}$ is discrete, then there is an alternative approach to approximate the quantification probabilities for a given query q: set a parameter $m > 1$, choose the m points of $S = \bigcup_{i=1}^n P_i$ that are closest to q, and use only these m points to estimate $\pi_i(q)$ for each P_i. We show that this works for a small value of m when, for each P_i, each location is approximately equally likely, but is not efficient if the location probabilities vary significantly.

Recall that w_{ij} is the location probability of a point $p_{ij} \in P_i$. Set $S = \bigcup_{i=1}^n P_i$ to be the set of all possible locations of points in $\mathcal{P}$. We define the quantity

$$\rho = \frac{\max_{i,j} w_{ij}}{\min_{i,j} w_{ij}}, \tag{9}$$

the ratio of the largest to the smallest location probability over all points of S, as the *spread* of location probabilities. Set

$$m(\rho, \varepsilon) = \rho k \ln(1/\varepsilon) + k - 1.$$

Fix a query point $q \in \mathbb{R}^2$. Let $\overline{S} \subseteq S$ be the $m(\rho, \varepsilon)$ nearest neighbors of q in S, $\overline{P}_i = \overline{S} \cap P_i$, and $\overline{\mathcal{P}} = \{\overline{P}_1, \ldots, \overline{P}_n\}$. Note that $\overline{w}_i = \sum_{p_{i,a} \in \overline{P}_i} w_{i,a}$ is not necessarily equal to 1, so we cannot regard $\overline{P}_i$ as an uncertain point in our model, but still it will be useful to think of $\overline{P}_i$ as an uncertain point that does not exist with probability $1 - \overline{w}_i$.

For a set Y of points and another point $\xi \in \mathbb{R}^2$, let

$$Y[\xi] = \left\{p \in Y \;\middle|\; d(q,p) \leq d(q,\xi)\right\}.$$

For a point $p := p_{i,a} \in P_i$, the probability that p is the NN of q in $\mathcal{P}$, denoted by $\eta(p; q)$, is

$$\eta(p;q) = w_{i,a} \prod_{j \neq i}\Big(1 - \sum_{p_{j,\ell} \in P_j[p]} w_{j,\ell}\Big). \tag{10}$$

Moreover,

$$\pi_i(q) = \sum_{p_{i,a} \in P_i} \eta(p_{i,a}; q). \tag{11}$$

For each $i \leq n$, $q \in \mathbb{R}^2$, and $p_{i,a} \in \overline{P}_i$, we analogously define the quantities $\widehat{\eta}(p_{i,a}; q)$ and $\widehat{\pi}_i(q)$ using Eq. (10) and Eq. (11) but replacing P_j with $\overline{P}_j$ for every $j \in \{1, \ldots, n\}$. Intuitively, if $\overline{\mathcal{P}}$ were a family of uncertain points, then $\widehat{\pi}_i(q)$ would be the probability of $\overline{P}_i$ being the NN of q in $\overline{\mathcal{P}}$.

Lemma 4.6. *For all* $i \in \{1, \ldots, n\}$,

$$\widehat{\pi}_i(q) \leq \pi_i(q) \leq \widehat{\pi}_i(q) + \varepsilon.$$

Proof: Fix a point $p \in P_i$. If $p \in \overline{P}_i$, then for all $j \neq i$, $\overline{P}_j[p] = P_j[p]$, therefore by Eq. (10), $\eta(p; q) = \widehat{\eta}(p; q)$.

Hence, by Eq. (11),

$$\pi_i(q) = \sum_{p \in \overline{P}_i} \eta(p; q) + \sum_{p \in P_i \setminus \overline{P}_i} \eta(p; q) = \sum_{p \in \overline{P}_i} \widehat{\eta}(p; q) + \sum_{p \in P_i \setminus \overline{P}_i} \eta(p; q) = \widehat{\pi}_i(q) + \sum_{p \in P_i \setminus \overline{P}_i} \eta(p; q). \tag{12}$$

Therefore $\widehat{\pi}_i(q) \leq \pi_i(q)$. Next, we bound the second term in the right hand side of Eq. (12). Let $p \in P_i \setminus \overline{P}_i$. Set $x_j = |P_j[p]|$, for $j \neq i$, and $m' = \sum_{j \neq i} x_j$. Since $P_i \setminus \overline{P}_i \neq \emptyset$, $|\overline{P}_i| \leq k - 1$ and $m' = |\overline{S}| - |\overline{P}_i| \geq \rho k \ln(1/\varepsilon)$. Note that each $w_{j,a} \geq 1/\rho k$. Therefore,

$$\begin{aligned} \eta(p; q) &= w_{i,a} \prod_{j \neq i} \Big(1 - \sum_{p_\ell \in P_j[p]} w_{j,\ell}\Big) \leq w_{i,a} \prod_{j \neq i} \Big(1 - \frac{x_j}{\rho k}\Big) \leq w_{i,a} \prod_{j \neq i} \exp\left(-x_j/\rho k\right) \\ &= w_{i,a} \exp\left(-m'/\rho k\right) \leq w_{i,a} \varepsilon. \end{aligned}$$

Consequently,

$$\sum_{p \in P_i \setminus \overline{P}_i} \eta(p; q) \leq \sum_{p \in P_i \setminus \overline{P}_i} \varepsilon w_{i,a} \leq \varepsilon. \tag{13}$$

Plugging Eq. (13) in Eq. (12), we obtain $\pi_i(q) \leq \widehat{\pi}_i(q) + \varepsilon$, as claimed. This completes the proof of the lemma. ∎

For any i, if $P_i \cap \overline{S}(q) = \emptyset$, then we can implicitly set $\widehat{\pi}_i(q)$ to 0. Using the data structure by [AC09], S can be preprocessed in $O(N \log N)$ randomized expected time into a data structure of $O(N)$ size so that $m := m(\rho, \varepsilon)$ nearest neighbors of a query point can be reported in $O(m + \log N)$ time. We thus obtain the following result.

Theorem 4.7. *Let $\mathcal{P}$ be a set of n uncertain points in $\mathbb{R}^2$, each with a discrete distribution of size at most k, let ρ be the spread of the location probabilities, and let $N = nk$. $\mathcal{P}$ can be preprocessed in $O(N \log N)$ expected time into a data structure of size $O(N)$, so that for a query point $q \in \mathbb{R}^2$ and a parameter $\varepsilon \in (0, 1)$, it can compute, in time $O(\rho k \log(1/\varepsilon) + \log N)$, values $\widehat{\pi}_i(q)$ for all $P_i \in \mathcal{P}$ such that $|\pi_i(q) - \widehat{\pi}_i(q)| \leq \varepsilon$ for all $i \in \{1, \ldots, n\}$.*

Remarks. (i) This approach is not efficient when the spread of location probabilities is unbounded. In this case, one may have to retrieve $\Omega(n)$ points. Another approach may be to ignore points with weight smaller than ε/k, since even k such weights from a single uncertain point P_i cannot contribute more than ε to $\pi_i(q)$. However, the union of all such points may distort other probabilities.

Consider the following example. Let $p_1 \in P_1 \in \mathcal{P}$ be the closest point to the query point q. Let $w(p_1) = 3\varepsilon$. Let the next $n/2$ closest points $p_3, \ldots, p_{n/2+2}$ be from different uncertain points $P_3, \ldots, P_{n/2+2}$ and each have weights $w(p) = 2/n \ll \varepsilon/k$. Let the next closest point $p_2 \in P_2 \in \mathcal{P}$ have weight $w(p_2) = 5\varepsilon$. With probability $\pi_{p_1}(q) = 3\varepsilon$ the nearest neighbor is p_1. The probability that p_2 is the nearest neighbor is $\pi_{p_2}(q) = (5\varepsilon)(1-3\varepsilon)(1-2/n)^{n/2} < (5\varepsilon)(1-3\varepsilon)(1/e) < 2\varepsilon$. Thus, p_1 is more likely to be the nearest neighbor than p_2. However, if we ignore points $p_3, \ldots, p_{n/2+2}$ because they have small weights, then we calculate p_2 has probability $\widehat{\pi}_{p_2}(q) = (1-3\varepsilon)(5\varepsilon) > 4\varepsilon$ for being the nearest neighbor (assuming that ε is small enough). So $\pi_2(q)$ will be off by more than 2ε and it would incorrectly appear that p_2 is more likely to be the nearest neighbor than p_1.

(ii) Though the data structure by [AC09] is optimal theoretically, it is too complex to be implemented. Instead, one may use the order-m Voronoi diagram to retrieve the m closest points (in unsorted order) to q. This would yield a data structure with $O(m(nk-m))$ space and $O(m(nk-m)\log(nk) + nk\log^3(nk))$ expected preprocessing time [ABMS98], while preserving the query time $O(\log(nk) + m)$, where $m = O(\rho k \log(\rho/\varepsilon))$. Alternatively, one may use quad-trees and a branch-and-bound algorithm to retrieve m points of S closest to q [Har11].

5 Conclusions

In this paper, we investigated NN queries in a probabilistic framework in which the location of each input point is specified as a probability distribution function. We presented efficient methods for returning all points with non-zero probability of being the nearest neighbor, estimating the quantification probabilities and using it for threshold NN queries. We conclude by mentioning two open problems:

(i) The lower-bound constructions for the complexity of $\mathcal{V}_{\neq 0}(\mathcal{P})$ are created very carefully, and these configurations are unlikely to occur in practice. A natural question is to characterize the sets of uncertain points for which the complexity of $\mathcal{V}_{\neq 0}(\mathcal{P})$ is near linear.

(ii) Are there simple and practical linear-size data structures for answering $\mathsf{NN}_{\neq 0}$ queries in sublinear time?

(iii) Can we extend the spiral search method to continuous distributions (at least for some simple, well-behaved distributions, such as Gaussian), so that the query time is always sublinear?

References

[AB86] P. F. Ash and E. D. Bolker. Generalized Dirichlet tessellations. *Geometriae Dedicata*, 20:209–243, 1986.

[ABMS98] P. K. Agarwal, M. de Berg, J. Matoušek, and O. Schwarzkopf. Constructing levels in arrangements and higher order Voronoi diagrams. *SIAM J. Comput.*, 27:654–667, 1998.

[AC09] P. Afshani and T. M. Chan. Optimal halfspace range reporting in three dimensions. In *Proc. 20th ACM-SIAM Sympos. Discrete Algs.*, pages 180–186, 2009.

[AESZ12] P. K. Agarwal, A. Efrat, S. Sankararaman, and W. Zhang. Nearest-neighbor searching under uncertainty. In *Proc. 31st ACM Sympos. Principles Database Syst.*, pages 225–236, 2012.

[Aga16] P. K. Agarwal. Range searching. In J. E. Goodman, J. O'Rourke, and C. Toth, editors, *Handbook of Discrete and Computational Geometry*, chapter 36, page to apepar. CRC Press LLC, 3rd edition, 2016.

[Agg09] C. C. Aggarwal. *Managing and Mining Uncertain Data*. Springer-Verlag, 2009.

[AS00] P. K. Agarwal and M. Sharir. Arrangements and their applications. In J.-R. Sack and J. Urrutia, editors, *Handbook of Computational Geometry*, pages 49–119. North-Holland Publishing Co., Amsterdam, 2000.

[BEK+11] T. Bernecker, T. Emrich, H.-P. Kriegel, N. Mamoulis, M. Renz, and A. Zuefle. A novel probabilistic pruning approach to speed up similarity queries in uncertain databases. In *Proc. 27th IEEE Int. Conf. Data Eng.*, pages 339–350, 2011.

[BSI08] G. Beskales, M. A. Soliman, and I. F. IIyas. Efficient search for the top-k probable nearest neighbors in uncertain databases. *Proc. VLDB Endow.*, 1(1):326–339, 2008.

[CCCX09] R. Cheng, L. Chen, J. Chen, and X. Xie. Evaluating probability threshold k-nearest-neighbor queries over uncertain data. In *Proc. 12th Int. Conf. Ext. Database Tech.*, pages 672–683, 2009.

[CCMC08] R. Cheng, J. Chen, M. Mokbel, and C. Chow. Probabilistic verifiers: Evaluating constrained nearest-neighbor queries over uncertain data. In *Proc. 24th IEEE Int. Conf. Data Eng.*, pages 973–982, 2008.

[CKP04] R. Cheng, D. V. Kalashnikov, and S. Prabhakar. Querying imprecise data in moving object environments. *IEEE Trans. Know. Data Eng.*, 16(9):1112–1127, 2004.

[CXY+10] R. Cheng, X. Xie, M. L. Yiu, J. Chen, and L. Sun. Uv-diagram: A Voronoi diagram for uncertain data. In *Proc. 26th IEEE Int. Conf. Data Eng.*, pages 796–807, 2010.

[dBCKO08] M. de Berg, O. Cheong, M. van Kreveld, and M. H. Overmars. *Computational Geometry: Algorithms and Applications*. Springer-Verlag, 3rd edition, 2008.

[DRS09] N. N. Dalvi, C. Ré, and D. Suciu. Probabilistic databases: Diamonds in the dirt. *Commun. ACM*, 52(7):86–94, 2009.

[DSST89] J. R. Driscoll, N. Sarnak, D. D. Sleator, and R. E. Tarjan. Making data structures persistent. *J. Comput. Syst. Sci.*, 38:86–124, 1989.

[DYM+05] X. Dai, M. L. Yiu, N. Mamoulis, Y. Tao, and M. Vaitis. Probabilistic spatial queries on existentially uncertain data. In *Proc. 9th Int. Sympos. Spatial Temporal Databases*, pages 400–417, 2005.

[Har11] S. Har-Peled. *Geometric Approximation Algorithms*, volume 173 of *Mathematical Surveys and Monographs*. Amer. Math. Soc., 2011.

[JCLY11] J. Jestes, G. Cormode, F. Li, and K. Yi. Semantics of ranking queries for probabilistic data. *IEEE Trans. Know. Data Eng.*, 23(12):1903–1917, 2011.

[KCS14] P. Kamousi, T. M. Chan, and S. Suri. Closest pair and the post office problem for stochastic points. *Comput. Geom. Theory Appl.*, 47(2):214–223, 2014.

[KKR07] H.-P. Kriegel, P. Kunath, and M. Renz. Probabilistic nearest-neighbor query on uncertain objects. In *Proc. 12th Int. Conf. Database Sys. Adv. App.*, pages 337–348, 2007.

[KMR+16] H. Kaplan, W. Mulzer, L. Roditty, P. Seiferth, and M. Sharir. Dynamic planar voronoi diagrams for general distance functions and their algorithmic applications. *CoRR*, abs/1604.03654, 2016.

[LLS01] Y. Li, P. M. Long, and A. Srinivasan. Improved bounds on the sample complexity of learning. *J. Comput. Syst. Sci.*, 62(3):516–527, 2001.

[LS07] V. Ljosa and A. K. Singh. APLA: Indexing arbitrary probability distributions. In *Proc. 23rd IEEE Int. Conf. Data Eng.*, pages 946–955, 2007.

[MR95] R. Motwani and P. Raghavan. *Randomized Algorithms.* Cambridge University Press, Cambridge, UK, 1995.

[SA95] M. Sharir and P. K. Agarwal. *Davenport-Schinzel Sequences and Their Geometric Applications.* Cambridge University Press, 1995.

[SE08] J. Sember and W. Evans. Guaranteed Voronoi diagrams of uncertain sites. In *Proc. 20th Canad. Conf. Comput. Geom.*, pages 207–210, 2008.

[VC71] V. N. Vapnik and A. Y. Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Theory Probab. Appl.*, 16:264–280, 1971.

[YTX+10] S. M. Yuen, Y. Tao, X. Xiao, J. Pei, and D. Zhang. Superseding nearest neighbor search on uncertain spatial databases. *IEEE Trans. Know. Data Eng.*, 22(7):1041–1055, 2010.

[ZCM+13] P. Zhang, R. Cheng, N. Mamoulis, M. Renz, A. Zufile, Y. Tang, and T. Emrich. Voronoi-based nearest neighbor search for multi-dimensional uncertain databases. In *Proc. 29th IEEE Int. Conf. Data Eng.*, pages 158–169, 2013.