Basic Measures for Imprecise Point Sets in $\mathbb{R}^d$

Heinrich Kruger

Masters Thesis
Game and Media Technology
Department of Information and Computing Sciences
Utrecht University
September 2008

INF/SCR-08-07

Abstract

Most algorithms in computational geometry tend to assume that all input is exact, with no imprecision or error. Most real-world data however, has some imprecision (for example due to measurement error). Thus, there exists a need for algorithms that can produce meaningful output for imprecise input data. In this thesis, I present results on the computation of upper and lower bounds on various basic measures (such as diameter, width, closest pair, volume of smallest enclosing ball and volume of minimum axis aligned bounding box) for imprecise point sets in $\mathbb{R}^d$. I model the imprecision by representing an imprecise point set as a set of regions (balls or polytopes), such that each point may lie anywhere within one of the regions. This work is an extension of previous research by Löffler and van Kreveld on imprecise point sets in $\mathbb{R}^2$, to higher dimensions.

Contents

List of Figures

Chapter 1

Introduction

1.1 The Problem of Imprecision

Computational geometers generally assume that input data to algorithms is exact, without any error or imprecision. Thus, an algorithm taking a set of points as input, will work on the assumption that the exact location of every point in the input is known precisely. In most real-world applications, however, there is almost always some imprecision in the input data. Thus, there is a need for algorithms that take imprecise data as input, and produce meaningful output, taking the imprecise nature of the input into account.

There are two types of imprecision encountered in geometric computation. Arithmetic imprecision is imprecision that arises due to the limited precision of arithmetic operations that can be performed by computers. This type of imprecision is very well studied and a vast body of work exists on developing algorithms that are robust in the face of such numerical error. Almost all of this work, however, does not take imprecision in the input data into account. The type of imprecision that I am interested in here is data imprecision: imprecision in the input data to an algorithm. Such imprecision generally arises due to measurement error (no measuring equipment is capable of yielding 100% exact data). Data imprecision can also arise due to data being stored using limited precision floating point data structures. While data imprecision has received an increasing amount of research attention over the last decade, this type of imprecision is not nearly as well studied as arithmetic imprecision.

1.2 Dealing with Imprecise Data

A number of different approaches to dealing with imprecise data can be found in the literature. For example, Abellanas et al. [1] study the *tolerance* of a geometric structure — they show how to compute the maximum perturbation of a set of points in $\mathbb{R}^2$ such that the topological structure of the Delaunay triangulation of the points remains unchanged. Averbakh and Bereg [2] study facility location problems for imprecise points modelled as rectangles in $\mathbb{R}^2$ and compute a minimax regret solution — that is, they compute a solution that has the best worst-case performance. In Chapter 2, I provide a more detailed overview of various different approaches to dealing with imprecise data.

I use the same approach as Löffler and van Kreveld [21] for dealing with imprecise point sets. I model a set of imprecise points as a set of regions $\mathcal{R} \subset \mathcal{P}(\mathbb{R}^d)$. If $\mathcal{R} = \{R_1, \ldots, R_n\}$, then each imprecise point p_i is allowed to lie anywhere within the region R_i. Now, for a set $\mathcal{R}$ and a measure $\mu : \mathcal{P}(\mathbb{R}^d) \to \mathbb{R}$ that maps a set of points in $\mathbb{R}^d$ to a real number, I want to compute an upper and a lower bound for μ over all point sets P such that $p_i \in P$ lies somewhere in $R_i \in \mathcal{R}$. In other words, if $\mathcal{F}_{\mathcal{R}}$ is the set of all mappings $f : \mathcal{R} \to \mathbb{R}^d$ such that $\forall R \in \mathcal{R},\ f(R) \in R$ then I want to compute

$$\mu_{\text{max}} = \max_{f \in \mathcal{F}_{\mathcal{R}}} \mu(f(\mathcal{R})) \quad \text{and} \tag{1.1}$$

$$\mu_{\text{min}} = \min_{f \in \mathcal{F}_{\mathcal{R}}} \mu(f(\mathcal{R})) \tag{1.2}$$

where, for $f \in \mathcal{F}_{\mathcal{R}},\ f(\mathcal{R}) = \{f(R) \mid R \in \mathcal{R}\}$.

Löffler and van Kreveld studied measures for imprecise points modelled as squares or discs in $\mathbb{R}^2$. Löffler [18] notes that in $\mathbb{R}^2$, squares represent a natural model for imprecision that arises due to points being stored using floating point data structures, while discs are a good model for various types of measurement error. The natural extension of the square model to higher dimensions would be to model points as hypercubes. I chose to model points as convex polytopes instead, since this is more general and any other convex model can be approximated using convex polytopes. Thus, I study measures for imprecise points in $\mathbb{R}^d$ modelled as either balls or convex polytopes of constant complexity.

1.3 Measures of Point Sets

Löffler and van Kreveld [21] studied the problems of computing upper and lower bounds for the *diameter* (the largest distance between any two points), the *closest pair* (the smallest distance between any two points), the *smallest enclosing ball*, the *minimum volume axis-aligned bounding box* and the *width* (the smallest distance between a pair of parallel hyperplanes such that the set of points is contained between the hyperplanes) for sets of imprecise points in $\mathbb{R}^2$. Here, I study the same problems in higher dimensional spaces.

These basic measures of point sets have a wide range of applications. For example, the width of an object has applications in stereolithography [24]. The minimum enclosing ball of a set of points is often used in facility location problems. Axis-aligned bounding boxes have applications in ray-tracing and collision detection, since it is easier to test whether a ray (or a robot) intersects an axis-aligned box than some more complex object.

The problems of computing these measures, for exact point sets, are well studied in computational geometry. It is well known that the smallest axis-aligned box of any set of points can be computed in linear time. In $\mathbb{R}^3$ Chazelle et al. [8] give algorithms to compute the diameter of a point set in $O(n^{1+\varepsilon})$ time and an $O(n^{8/5+\varepsilon})$ time algorithm for computing the width of a set of points. The smallest enclosing ball of a set of points is a well known example of an LP-type problem and can be solved in $O(d^2n) + e^{O(\sqrt{d \ln d})}$ expected time in $\mathbb{R}^d$ [22]. Finally, Bentley [5], showed that the closest pair of a point set in $\mathbb{R}^d$ can be computed in $O(n \log n)$ time.

1.3.1 Summary of Results

Table 1.1 shows a summary of my results. Some of these problems have been studied previously. For example, computing the *largest closest pair* for a set of imprecise points requires finding a mapping $f \in \mathcal{F}_{\mathcal{R}}$ such that the minimum distance between any two points p and q in $f(\mathcal{R})$ is as large is possible. Fiala et al. [11] showed that this problem is NP-Hard.

Also, the *smallest minimum enclosing ball* of a set of imprecise points, is the smallest ball that intersects every $R \in \mathcal{R}$. The problem of computing such a ball is known to belong to the well studied class of LP-type problems[22].

Other results — such as the *largest minimum volume axis-aligned bounding box* for any set of imprecise points, the *smallest minimum volume axis-aligned bounding box* for points modelled as balls and the *largest minimum enclosing ball* for points modelled as balls or as convex polytopes — are direct extensions of the results of Löffler and van Kreveld [21] in $\mathbb{R}^2$ to higher dimensions.

Table 1.1: Summary of results.

Problem	Model	Smallest	Largest
Diameter	Convex Polytopes Balls	open open	$O(dn^2)$ $O(dn^2)$
Closest Pair	Convex Polytopes Balls	$e^{O(\sqrt{d \ln d})}O(n^2)^*$ $O(dn^2)$	NP-Hard NP-Hard
Axis-aligned Minimum Bounding Box	axis-aligned boxes Convex Polytopes Balls	$O(dn)$ open $O(n^{7+\varepsilon})$ (in $\mathbb{R}^3$)	$O(dn)$ $O(d^2n) + 2^{O(d^2 \log d)}$ $O(d^2n) + 2^{O(d^2 \log d)}$
Minimum Enclosing Ball	Convex Polytopes Balls	$2^{O(d)}O(n)^*$ $2^{O(d)}O(n)^*$	$O(d^3n) + e^{O(\sqrt{d \ln d})*}$ $2^{O(d)}O(n)^*$
Width	Convex Polytopes Balls	$n^{O(d)}$ $O(d^2n^{d+2})$	NP-hard open

1.4 Outline

The remainder of this document is structured as follows. In Chapter 2, I give a brief overview of previous research on algorithms for imprecise point sets. The results of my own research are given in Chapter 3. Finally, in Chapter 4, I conclude with a discussion of my results and an overview of some open problems and possible directions for further research.

* These are the expected running times for randomised algorithms

Chapter 2

Related Work

Over the last decade, there has been increasing research interest into algorithms for imprecise data. Most of this work however, focuses on problems in $\mathbb{R}^2$, one exception to this is the work by Bandyopadhyay and Snoeyink [4] who consider imprecise point sets in $\mathbb{R}^3$. Also, most existing research on imprecise point sets focuses on computation of *Convex Hulls* [23, 20, 15] and *Delaunay triangulations* [1, 4, 17, 19]. It is however still interesting to look at these results, to see the various different approaches to dealing with imprecise data.

2.1 Epsilon Geometry

Guibas et al. [14] developed *Epsilon Geometry* as a framework for developing robust algorithms using imprecise computations. In the epsilon geometry framework, a geometric predicate $P(X)$ is implemented to return a pair $\varepsilon^-, \varepsilon^+$ such that for any $\varepsilon \geq \varepsilon^+$, there is some X' in a ball of radius ε centred at X such that $P(X')$ is true; for $\varepsilon < \varepsilon^-$, no such X' exists and for $\varepsilon^- \leq \varepsilon < \varepsilon^+$ the existence of such X' can not be determined, due to imprecision.

Guibas et al. [15] apply the epsilon geometry framework to the computation of the convex hull of a set of points in $\mathbb{R}^2$. They define a polygon to be $(-\varepsilon)$-convex if it remains convex when each vertex is perturbed by some distance less or equal to ε. They show that for any set P of points in $\mathbb{R}^2$, it is possible to compute a $(-\varepsilon)$-convex polygon H whose vertices are points in P, such that each point in P has distance at most δ from H where $\delta \geq 2\varepsilon$. They call such a polygon H a $(-\varepsilon)$-convex δ-hull of P.

While this framework was developed as a means of coping with arithmetic imprecision, it may also be possible to use epsilon geometry to develop algorithms for imprecise data, by treating any computation involving imprecise data as an imprecise computation.

2.2 Tolerance

Abellanas et al. [1] study Delaunay triangulations of imprecise point sets in $\mathbb{R}^2$. However, instead of computing a Delaunay triangulation, they study the structural tolerance of the Delaunay triangulation for a set of points. They define the *tolerance* of a structure (such as the Delaunay triangulation of a set of points P) as the maximum δ such that the topology of the structure remains unchanged if all points in P are perturbed by some distance less or equal to δ.

Bandyopadhyay and Snoeyink [4] study Delaunay tessellations of point sets in $\mathbb{R}^d$. Their approach is very similar to that of Abellanas et al., but instead of computing the tolerance of the Delaunay triangulation, they focus on almost-Delaunay simplices. For a set P of points in $\mathbb{R}^d$, they define $Q \subset P$ with $|Q| = d$ as almost-Delaunay with threshold ε if ε is the minimum perturbation of P required for the perturbed Q to become a Delaunay simplex.

2.3 Preprocessing Regions

Löffler and Snoeyink [19] study Delaunay triangulations of imprecise points modelled as disjoint unit discs in $\mathbb{R}^2$. Their approach is quite different from those of Abellanas et al. [1] and Bandyopadhyay and Snoeyink [4] discussed above. They show that it is possible to preprocess the set of regions $\mathcal{R}$ in $O(n \log n)$ time, such that for any $f \in \mathcal{F}_{\mathcal{R}}$, the Delaunay triangulation of $f(\mathcal{R})$ can be computed in linear time.

2.4 Minimax Regret

Averbakh and Bereg [2] study weighted facility location problems under the L_1 metric in $\mathbb{R}^2$ for imprecise points modelled as rectangles with imprecise weights modelled as intervals. For the Euclidean (L_2) metric they consider only precise points with imprecise weights modelled as intervals. They deal with the imprecision by computing a *minimax regret* solution.

Let X^* be the set of all possible optimal solutions. Averbakh and Bereg compute a solution x such that the maximum over all $x^* \in X^*$ of difference between $F(x) - F(x^*)$ is as small as possible (where F is the objective function).

2.5 Upper and Lower Bounds

As I discussed in Chapter 1, my work extends the results of Löffler and van Kreveld [21] in $\mathbb{R}^2$ to higher dimensions. In this section, I discuss some other research on computing upper and lower bounds for some geometric problems on imprecise point sets.

Nagai and Tokura [23] study the convex hull and diameter of imprecise point sets modelled as convex regions in $\mathbb{R}^2$. As an upper bound for the convex hull of a set of imprecise points, they compute the union of all possible convex hulls. Similarly they obtain a lower bound for the convex hull by computing the intersection of all possible convex hulls. More precisely, if $\mathrm{CH}(P)$ denotes the convex hull of a set of points P, they compute

$$\begin{aligned} \overline{\mathrm{CH}}(\mathcal{R}) &= \bigcup_{f \in \mathcal{F}_{\mathcal{R}}} \mathrm{CH}(f(\mathcal{R})) \quad \text{and} \\ \underline{\mathrm{CH}}(\mathcal{R}) &= \bigcap_{f \in \mathcal{F}_{\mathcal{R}}} \mathrm{CH}(f(\mathcal{R})) \end{aligned}$$

These bounds have the property that for any $f \in \mathcal{F}_{\mathcal{R}}$,

$$\underline{\mathrm{CH}}(\mathcal{R}) \subseteq \mathrm{CH}(f(\mathcal{R})) \subseteq \overline{\mathrm{CH}}(\mathcal{R}).$$

This approach has the disadvantage that in many cases $\nexists f \in \mathcal{F}_{\mathcal{R}}$ such that $\mathrm{CH}(f(\mathcal{R})) = \underline{\mathrm{CH}}(\mathcal{R})$ or $\mathrm{CH}(f(\mathcal{R})) = \overline{\mathrm{CH}}(\mathcal{R})$.

Löffler and van Kreveld [20] also study upper and lower bounds for the convex hull of a set of imprecise points in $\mathbb{R}^2$. Their approach is similar to that of Löffler and van Kreveld [21] in that they computes upper and lower bounds on measures (namely the area and perimeter) of the convex hull. This approach has the advantage that there is always guaranteed to be some $f \in \mathcal{F}_{\mathcal{R}}$ such that the convex hull of $f(\mathcal{R})$ actually attains the lower or upper bound on the area or perimeter. A disadvantage however is that in many cases these bounds are somewhat harder to compute than those of Nagai and Tokura. Löffler and van Kreveld give algorithms ranging from $O(n \log n)$ time to compute the smallest perimeter convex hull of points modelled as squares or as parallel line segments, to $O(n^{10})$ to compute the largest perimeter convex hull of points modelled as non-intersecting squares. Some of the problems they study, for example computing the largest area and largest perimeter convex hulls of points modelled as line segments are even NP-Hard. Nagai and Tokura on the other hand show that for points modelled as convex polygons or as discs, both $\underline{CH}(\mathcal{R})$ and $\overline{CH}(\mathcal{R})$ can be computed in $O(n \log n)$ time.

For the maximum diameter of an imprecise point set, Nagai and Tokura use the same definition as Löffler and van Kreveld and develop an $O(n\alpha(n) \log n)$ algorithm to compute the maximum width when each region is defined as the intersection of as set of halfplanes. For the minimum diameter however, they compute the diameter of $\underline{\mathrm{CH}}(\mathcal{R})$ in $O(n \log n)$ time. Löffler and van Kreveld point out that the minimum diameter of a set of imprecise points may be determined by more than one pair of points simultaneously, thus the lower bound computed by Nagai and Tokura may be smaller than the actual smallest possible diameter

$$\min_{f \in \mathcal{F}_{\mathcal{R}}} \max_{1 \leq i < j \leq n} \|f(R_i), f(R_j)\|.$$

Chapter 3

Results

3.1 Notation

As mentioned in Chapter 1, I studied measures of imprecise point sets in $\mathbb{R}^d$. Before I discuss my results, I give a brief overview of notation used.

I model a set of n imprecise points as a set $\mathcal{R} = \{R_1, \ldots, R_n\}$ where each R_i is a closed, convex subset of $\mathbb{R}^d$. In the case where $\mathcal{R}$ is a set of polytopes, it is assumed that all polytopes in $\mathcal{R}$ have constant complexity. Note that some dependence on d is unavoidable, for example a simplex in $\mathbb{R}^d$ has $O(d)$ vertices, however for the purposes of algorithm analysis I treat the complexity of all regions as constant.

I use $\mathcal{F}_{\mathcal{R}}$ to refer to the set of all mappings $f : \mathcal{R} \to \mathbb{R}^d$ such that $\forall R \in \mathcal{R},\ f(R) \in R$. Additionally, for any $f : S \to S'$ mapping elements of set S to elements of set S', if $T \subseteq S$, then $f(T)$ refers to the set $\{f(t) \mid t \in T\}$. Finally, the Euclidean distance between two points $p, q \in \mathbb{R}^d$ is denoted by $\|p, q\|$.

3.2 Diameter

The diameter of a set of points P, is defined as the maximum distance between any pair of points in P. Now, for an imprecise point set modelled as a set $\mathcal{R} = \{R_1, \ldots, R_n\}$ of regions, we want to compute the maximum and minimum diameter over all point sets $P = \{p_1, \ldots, p_n\}$ such that $p_i \in R_i$.

3.2.1 Maximum Diameter

For any imprecise point set modelled as a set of regions $\mathcal{R}$, the maximum diameter is defined as:

$$D_{\text{max}} = \max_{R_i, R_j \in \mathcal{R},\ i \neq j} \left(\max_{p_i \in R_i, p_j \in R_j} \|p_i, p_j\| \right) \tag{3.1}$$

Now, the maximum diameter can be computed by considering each pair $R_i, R_j \in \mathcal{R}$, with $i \neq j$ and computing the maximum distance between a point in R_i and a point in R_j. I shall show that the maximum diameter of a set of imprecise points modelled as balls or polytopes can be computed in $O(dn^2)$ time.

Balls

For any pair of balls b_i and b_j with centres c_i and c_j and radii ρ_i and ρ_j respectively, the maximum distance between a pair of points $p_i \in b_i$ and $p_j \in b_j$ is given by:

$$\max_{p_i \in b_i, p_j \in b_j} \|p_i, p_j\| = \|c_i, c_j\| + \rho_i + \rho_j.$$

Clearly, this can be computed in $O(d)$ time.

Convex Polytopes

Lemma 3.2.1. *If P and Q are polytopes and $p \in P$ and $q \in Q$ are points such that $\|p, q\| = \max_{p' \in P, q' \in Q} \|p', q'\| > 0$, then p is a vertex of P and q is a vertex of Q.*

Proof. Assume that p is not a vertex of P, then there exists a vector v, orthogonal to the line segment $\overline{pq}$ with $\|v\| > 0$ such that the point $p' = p + v \in P$. Now $\|p', q\| > \|p, q\|$ which is a contradiction. Thus p and q are vertices of P and Q respectively. □

Now, for any pair of polytopes P and Q of constant complexity,

$$\max_{p \in P, q \in Q} \|p, q\|$$

can be computed in $O(d)$ time by considering each pair p, q where p is a vertex of P and q is a vertex of Q.

Computing the Maximum Diameter

Now, the maximum diameter of a set of imprecise points modelled as balls or polytopes can be computed by considering every pair of regions R_i, R_j with $i \neq j$ and computing the maximum distance between a pair of points $p_i \in R_i$ and $p_j \in R_j$.

There are $O(n^2)$ such pairs R_i, R_j and for each pair it takes $O(d)$ time to compute

$$\max_{p_i \in R_i, p_j \in R_j} \|p_i, p_j\|.$$

Theorem 3.2.2. *The maximum diameter for a set of imprecise points modelled as balls or polytopes, can be computed in $O(dn^2)$ time.*

3.2.2 Minimum Diameter

For an imprecise point set, modelled as a set $\mathcal{R} = \{R_1, \ldots, R_n\}$, the minimum diameter of $\mathcal{R}$ is defined as:

$$D_{\text{min}} = \min_{f \in \mathcal{F}_{\mathcal{R}}} \left(\max_{1 \leq i < j \leq n} \|f(R_i), f(R_j)\| \right) \tag{3.2}$$

Note that in general

$$D_{\text{min}} \neq \min_{R_i, R_j \in \mathcal{R},\ i \neq j} \left(\max_{p_i \in R_i, p_j \in R_j} \|p_i, p_j\| \right).$$

Thus, an approach similar to that used above for computing the maximum diameter, does not work for computing the minimum diameter.

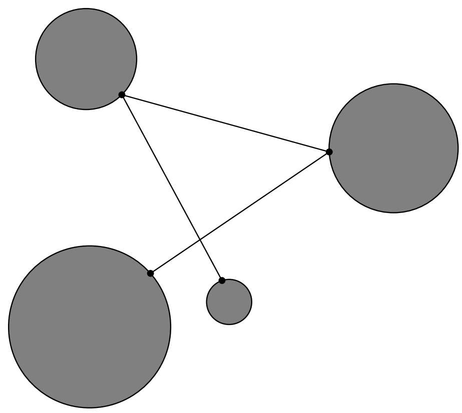

Figure 3.1: Several pairs of points can simultaneously determine the minimum diameter.

Computing the minimum diameter is made difficult by the fact that several pairs of points can simultaneously determine the minimum diameter (see for example Figure 3.1). In such a case, moving any point to reduce the distance between one pair of points, will increase the distance between another pair of points.

In $\mathbb{R}^2$, Löffler and van Kreveld [21] give an $O(n \log n)$ time algorithm to compute the minimum diameter for a set of imprecise points modelled as squares and an $O(n^{c\varepsilon^{-\frac{1}{2}}})$ time $(1+\varepsilon)$-approximation algorithm for points modelled as discs. These algorithms, however, do not generalise well to higher dimensions and I have been unable to find an algorithm to compute the minimum width of an imprecise point set in $\mathbb{R}^d$. Thus, this problem remains open.

3.3 Closest Pair

For any point set $P \subset \mathbb{R}^d$, the closest pair of points is the pair of points $p, q \in P$ such that the distance between p and q is minimum among all pairs of points in P. For an imprecise point set, modelled as a set of regions $\mathcal{R} = \{R_1, \ldots, R_n\}$ we want to compute the smallest and largest possible distance between the closest pair of points over all sets $P = \{p_1, \ldots, p_n\}$ such that $p_i \in R_i$.

3.3.1 Smallest Closest Pair

The smallest closest pair in a set of imprecise points modelled as a set of regions $\mathcal{R}$ is the pair of points $p_i \in R_i,\ p_j \in R_j,\ i \neq j$ such that

$$\|p_i, p_j\| = \min_{R_i, R_j \in \mathcal{R},\ i \neq j} \left(\min_{p_i \in R_i,\ p_j \in R_j} \|p_i, p_j\| \right). \tag{3.3}$$

Now, the smallest closest pair can be computed by considering pair $R_i, R_j \in \mathcal{R}$, with $i \neq j$ and computing the minimum distance between R_i and R_j. Clearly, there are $O(n^2)$ pairs R_i, R_j that need to be considered.

Balls

For any pair of balls b_i and b_j with centres c_i and c_j and radii ρ_i and ρ_j respectively, the minimum distance between a pair of points $p_i \in b_i$ and $p_j \in b_j$ is given by:

$$\min_{p_i \in b_i, p_j \in b_j} \|p_i, p_j\| = \max\{0, \|c_i, c_j\| - \rho_i - \rho_j\}.$$

Clearly, this can be computed in $O(d)$ time.

Theorem 3.3.1. *The smallest closest pair for a set of imprecise points modelled as balls in $\mathbb{R}^d$ can be computed in $O(dn^2)$ time.*

Unit Balls: If $\mathcal{R} = \{b_1, \dots, b_n\}$ is a set of unit balls, then the right hand side of Equation 3.3 can be rewritten as follows:

$$\min_{b_i, b_j \in \mathcal{R},\ i \neq j} \max\{0, \|c_i, c_j\| - 2\}.$$

Thus, the smallest closest pair can be found by computing the closest pair of the set $\{c_1, \dots, c_n\}$ of centres of balls in $\mathcal{R}$. Using the algorithm of Bentley [5], this can be done in $O(n \log n)$ time in any dimension.

Convex Polytopes

If P and Q are convex polytopes of constant complexity, then the minimum distance between a point $p \in P$ and a point $q \in Q$ can be computed in $e^{O(\sqrt{d \ln d})}$ expected time [13].

Theorem 3.3.2. *The smallest closest pair of a set of imprecise points modelled as convex polytopes in $\mathbb{R}^d$ can be computed in $e^{O(\sqrt{d \ln d})} O(n^2)$ expected time.*

3.3.2 Largest Closest Pair

For a set $\mathcal{R} = \{R_1, R_2, \dots, R_n\}$ of regions in $\mathbb{R}^d$, the largest closest pair for $\mathcal{R}$ is the pair of points $p_i \in R_i,\ p_j \in R_j,\ i \neq j$ such that

$$\|p_i, p_j\| = \max_{f \in \mathcal{F}_\mathcal{R}} \left(\min_{1 \leq i < j \leq n} \|f(R_i), f(R_j)\| \right). \tag{3.4}$$

Fiala et al. [11] showed that finding a mapping that maximises the minimum distance between a pair of points is NP-Hard. Thus, finding the largest closest pair of any set of imprecise points is also NP-Hard. For points modelled as discs in $\mathbb{R}^2$, Cabello [6] gives an $O(n^2)$ time algorithm that achieves an approximation ratio of $\frac{8}{3}$.

3.4 Axis Aligned Minimum Bounding Box

The minimum volume axis-aligned bounding box of a set of points P is the box B of minimum volume with every edge of B parallel to one of the axes in $\mathbb{R}^d$, such that $P \subset B$. In other words, B is the Cartesian product of the intervals $[x_i^-, x_i^+]$ for $i \in \{1, \dots, d\}$ where $\prod_{i \in \{1,\dots,d\}} (x_i^+ - x_i^-)$ is as small as possible, such that $P \subset B$.

3.4.1 Largest Axis-aligned Minimum Bounding Box

Löffler and van Kreveld [21] showed that to compute the largest axis-aligned minimum bounding box for any imprecise point set modelled as a set of regions $\mathcal{R}$ in $\mathbb{R}^2$, it is sufficient to compute the largest axis-aligned minimum bounding box of a subset $\mathcal{R}' \subset \mathcal{R}$ of sixteen regions, specifically the four most extreme regions in each of the four axis parallel directions. The same approach can also be used in higher dimensions.

If $\mathcal{R} = \{R_1, \ldots, R_n\}$ is an imprecise point set in $\mathbb{R}^d$, then for each $i \in \{1, \ldots, d\}$, let $\alpha_i^- : \{1, \ldots, n\} \to \{1, \ldots, n\}$ such that for $0 < j < k \leq n$, $\min_{p \in R_{\alpha_i^-(j)}} p_{x_i} \leq \min_{p \in R_{\alpha_i^-(k)}} p_{x_i}$ where $p = (p_{x_1}, \ldots, p_{x_d})$ and let $\alpha_i^+ : \{1, \ldots, n\} \to \{1, \ldots, n\}$ such that for $0 < j < k \leq n$, $\max_{p \in R_{\alpha_i^+(j)}} p_{x_i} \geq \max_{p \in R_{\alpha_i^+(k)}} p_{x_i}$. Now, let

$$\begin{aligned} \mathcal{R}_i^- &= \{R_{\alpha_i^-(j)} \in \mathcal{R} \mid j \in \{1, \ldots, 2d\}\} \\ \mathcal{R}_i^+ &= \{R_{\alpha_i^+(j)} \in \mathcal{R} \mid j \in \{1, \ldots, 2d\}\}. \end{aligned}$$

Finally, let

$$\mathcal{R}' = \bigcup_{i \in \{1, \ldots, d\}} \left(\mathcal{R}_i^- \cup \mathcal{R}_i^+\right).$$

In other words, $\mathcal{R}'$ is the set containing the $2d$ most extreme regions in each of the $2d$ axis parallel directions.

Lemma 3.4.1. *The largest axis-aligned minimum bounding box of $\mathcal{R}'$ is also the largest axis-aligned bounding box of $\mathcal{R}$.*

Proof. Assume this is not the case. Then some x_i^+ or x_i^- (say x_1^-) is determined by a region $R \in \mathcal{R} \setminus \mathcal{R}'$. At most $2d - 1$ of the regions in $\mathcal{R}_1^-$ can be involved in determining other boundaries of the minimum bounding box. Thus, $\exists R' \in \mathcal{R}_1^-$ such that R' is not involved in determining any boundary of the minimum bounding box and we are free to place the point corresponding to region R' anywhere in R'. By definition, $\min_{p \in R'} p_{x_1} \leq \min_{q \in R} q_{x_1}$. Now there are two cases to consider:

- If $\min_{p \in R'} p_{x_1} < \min_{q \in R} q_{x_1}$, we can place the point corresponding to region R' outside the bounding box, which is a contradiction.
- If $\min_{p \in R'} p_{x_1} = \min_{q \in R} q_{x_1}$ then using R' instead of R to determine x_1^- gives the same bounding box.

Thus, the largest axis-aligned minimum bounding box of $\mathcal{R}'$ is also the largest axis-aligned bounding box of $\mathcal{R}$. □

Each x_i^σ is determined by a point $p \in R \in \mathcal{R}_i^\sigma$ (where σ is either $+$ or $-$). Thus, to compute the largest axis-aligned minimum bounding box of $\mathcal{R}'$, we can consider each $C \subset \mathcal{R}'$ such that C contains a region R_i^σ from each $\mathcal{R}_i^\sigma$. Now, for each such C we need to find a unique point $p \in R$ for each $R \in C$ such that the minimum bounding box of these points is as large as possible. Clearly, the largest such bounding box over all C is the largest minimum bounding box of $\mathcal{R}$. Note that it is possible that a region R occurs in more than one of the sets $\mathcal{R}_i^\sigma$. Thus, there may be sets C such that for some $i \neq j$, $R = C \cap \mathcal{R}_i^\sigma = C \cap \mathcal{R}_j^\varsigma$ (here ς is also $+$ or $-$, but may be different from σ). In such a case, both x_i^σ and x_j^ς will be determined by the same point $p \in R$.

Now, there are $2^{O(d \log d)}$ sets C to consider. In the simplest case, when $|C| = 2d$ the largest minimum bounding box for C is the smallest box containing C, which can be computed in $O(d^2)$ time. However, if $|C| < 2d$, we need to find a point on the surface of each $R \in C$ such that the volume of the minimum bounding box is as large as possible. This may require $k^{O(d)}$ time, where k is a constant bounding the complexity of each $R \in C$.

Thus, the largest minimum bounding box of $\mathcal{R}'$ can be computed in $2^{O(d^2 \log d)}$ time. Additionally, $O(d^2 n)$ time is required to compute $\mathcal{R}'$.

Theorem 3.4.2. *The largest axis-aligned minimum bounding box for any set of imprecise points can be computed in* $O(d^2 n) + 2^{O(d^2 \log d)}$ *time.*

Special case: Axis-aligned Boxes

In the case where $\mathcal{R}$ is a set of axis-aligned boxes, we can take advantage of the fact that the faces of each $R \in \mathcal{R}$ are parallel to the faces of the box we want to compute, to obtain a more efficient algorithm, by treating each dimension independently. It is easy to see that the largest axis-aligned minimum bounding box of a set of imprecise points modelled as axis-aligned boxes can be obtained by maximising each $x_i^+ - x_i^-$.

Algorithm 3.1, computes the largest minimum volume axis-aligned bounding box for a set of imprecise points modelled as a set $\mathcal{R}$ of axis-aligned boxes. Clearly, it takes $O(n)$ time to compute x_i^+ and x_i^- for each $i \in \{1, \ldots, d\}$. Thus this algorithm requires $O(dn)$ time.

Algorithm 3.1 Largest axis-aligned minimum bounding box for axis-aligned boxes.

Input: A set $\mathcal{R}$ of axis-aligned boxes in $\mathbb{R}^d$
Output: The largest axis-aligned box B of $\mathcal{R}$

```
for each i ∈ {1, ..., d} do
    R⁻ ← arg min_{R∈𝓡} min_{p∈R} p_{x_i}
    a⁻ ← min_{p∈R⁻} p_{x_i}
    b⁻ ← min_{R∈𝓡\{R⁻}} min_{p∈R} p_{x_i}
    R⁺ ← arg max_{R∈𝓡} max_{p∈R} p_{x_i}
    a⁺ ← max_{p∈R⁻} p_{x_i}
    b⁺ ← max_{R∈𝓡\{R⁻}} max_{p∈R} p_{x_i}
    if R⁻ ≠ R⁺ then
        x_i⁻ ← a⁻
        x_i⁺ ← a⁺
    else if |a⁺ − b⁻| ≥ |b⁺ − a⁻| then
        x_i⁻ ← b⁻
        x_i⁺ ← a⁺
    else
        x_i⁻ ← a⁻
        x_i⁺ ← b⁺
    end if
end for
B ← [x_1⁻, x_1⁺] × [x_2⁻, x_2⁺] × ... × [x_d⁻, x_d⁺]
return B
```

3.4.2 Smallest Axis-aligned Minimum Bounding Box

For an imprecise point set modelled as a set of regions $\mathcal{R}$, the smallest axis-aligned minimum bounding box, is the box B of minimum volume, that intersects every $R \in \mathcal{R}$.

Axis-aligned Boxes

If $\mathcal{R} = \{R_1, \ldots, R_n\}$ is a set of axis-aligned boxes, then the smallest axis-aligned box intersecting all $R \in \mathcal{R}$ is easy to compute in linear time. Since the faces of the boxes in $\mathcal{R}$ are parallel to the faces of the bounding box we want to compute, we can treat each dimension independently and obtain the smallest minimum bounding box by minimising $x_i^+ - x_i^- \; \forall i \in \{1, \ldots, d\}$.

The smallest axis-aligned box intersecting each box in $\mathcal{R}$ can be computed by Algorithm 3.2. Clearly, for each $i \in \{1, \ldots, d\}$ it takes $O(n)$ time to compute x_i^+ and x_i^-. Thus, it takes $O(dn)$ time to compute the smallest axis-aligned bounding box of a set of imprecise points modelled as axis-aligned boxes.

Algorithm 3.2 Smallest axis-aligned minimum bounding box for axis-aligned boxes.

Input: A set $\mathcal{R}$ of axis-aligned boxes in $\mathbb{R}^d$
Output: The smallest box B intersecting all boxes in $\mathcal{R}$
  for each $i \in \{1, \ldots, d\}$ **do**
    $a^- \leftarrow \min_{R \in \mathcal{R}} \max_{p \in R} p_{x_i}$
    $a^+ \leftarrow \max_{R \in \mathcal{R}} \min_{p \in R} p_{x_i}$
    $x_i^+ \leftarrow a^+$
    if $a^+ \leq a^-$ **then**
      $x_i^- \leftarrow x_i^+$
    else
      $x_i^- \leftarrow a^-$
    end if
  end for
  $B \leftarrow [x_1^-, x_1^+] \times [x_2^-, x_2^+] \times \ldots \times [x_d^-, x_d^+]$
  return B

Balls

Colley et al. [9] give an algorithm to compute the minimum area axis-aligned rectangle intersecting each polygon in a set of convex polygons in $\mathbb{R}^2$. Löffler and van Kreveld [21] modified this algorithm to obtain an $O(n^2)$ algorithm for computing the smallest axis-aligned minimum bounding box for a set of imprecise points modelled as discs in $\mathbb{R}^2$.

A similar method can be used for imprecise points modelled as balls in $\mathbb{R}^3$. First, let B_0 be an initial box as computed by Algorithm 3.2.

Lemma 3.4.3. *If B_0 has zero volume, then there exists a box of zero volume intersecting all balls in $\mathcal{R}$.*

Proof. Clearly B_0 has zero volume if and only if there exists a plane P with $B_0 \subset P$ such that P is parallel to one of the xy, yz or xz-planes and P intersects every ball in $\mathcal{R}$. Now, there exists some axis-aligned rectangle $R \subset P$ such that R intersects every ball in $\mathcal{R}$. R is a box of zero volume, intersecting all balls in $\mathcal{R}$. □

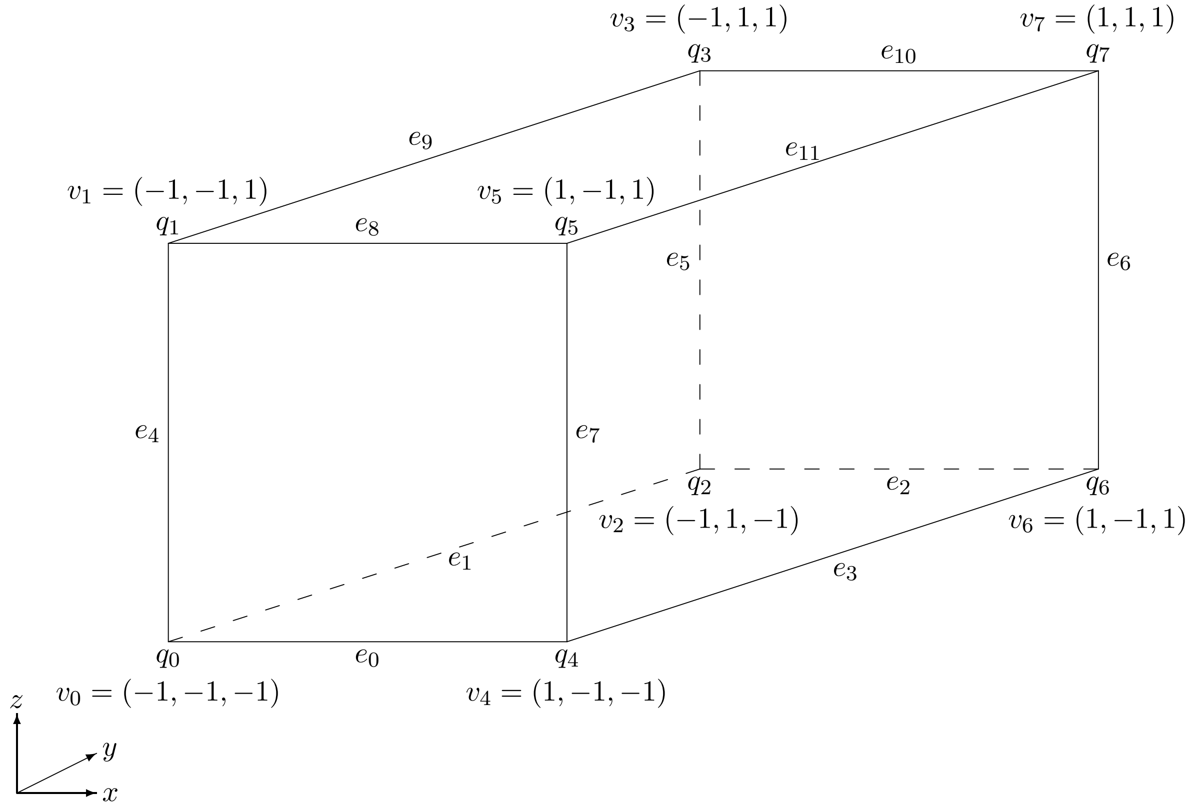

Figure 3.2: Numbering of corners and edges of a box in $\mathbb{R}^3$.

Now, we only need to consider the case where B_0 has positive volume. Let $\mathcal{R}' = \{b \in \mathcal{R} \mid b \cap B_0 = \emptyset\}$. If $\mathcal{R}' = \emptyset$ then B_0 is the minimum volume axis-aligned box intersecting every ball in $\mathcal{R}$, and we are done; so assume $\mathcal{R}' \neq \emptyset$. Clearly, any axis-aligned box that intersects every ball in $\mathcal{R}$ must contain B_0; so we only need to compute the smallest box containing B_0 and intersecting every ball in $\mathcal{R}'$.

Consider the subdivision of $\mathbb{R}^3$ induced by the six supporting planes of faces of B_0. This subdivision consists of 27 3-dimensional cells: B_0 itself, 6 face cells (each with exactly one face of B_0 on its boundary), 8 corner cells $C_0, \ldots, C_7$ (each of which has exactly one corner of B_0 on its boundary) and 12 edge cells $E_0, \ldots, E_{11}$ (each of which has exactly one edge and two corners of B_0 on its boundary), with the corners and edges numbered as in Figure 3.2.

Lemma 3.4.4. *For any ball $b \in \mathcal{R}'$, the centre c_b of b lies either in the interior of some corner cell C_i or in the interior of an edge cell E_j or on the boundary between an edge cell E_j and a corner cell C_i.*

Proof. Let b be a ball in $\mathcal{R}'$ with centre c_b and radius ρ_b. Clearly $c_b \notin B_0$. Now, assume that c_b lies in a face cell F (or on the boundary between F and an edge cell). Let P be the plane forming the boundary between F and B_0 and let $p \in P$ be the point in p closest to c_b. Clearly, the line segment $\overline{pc_b}$ is orthogonal to P, so $p \in B_0$. From the construction of B_0, it should be clear that $\|p, c_b\| \leq \rho_b$. Thus, $p \in b$ which implies that $b \cap B_0 \neq \emptyset$ which is a contradiction. So c_b must lie in the interior of either a corner cell or an edge cell, or on the boundary between a corner cell and an edge cell. $\square$

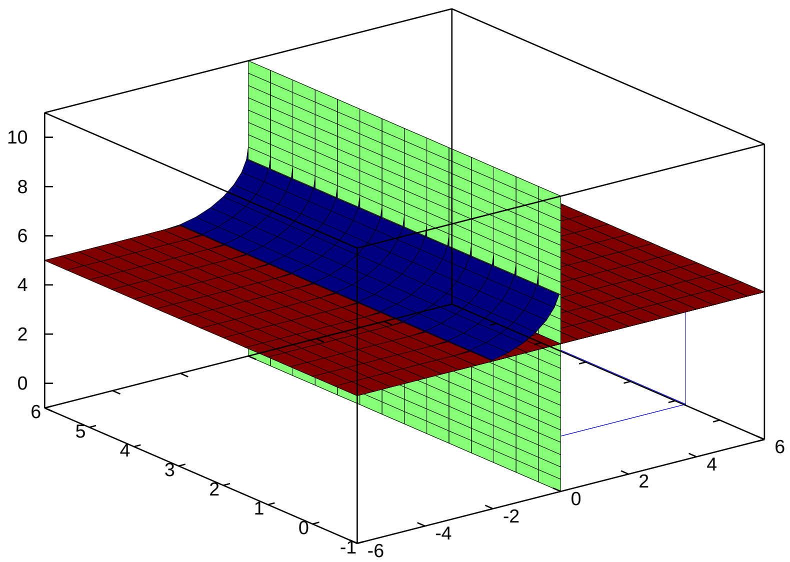

Figure 3.3: γ_b for a ball in an edge cell.

Now, we can obtain a partition of $\mathcal{R}'$ into 20 disjoint subsets

$$\mathcal{R}' = \left(\bigcup_{i \in \{1,\ldots,8\}} \mathcal{R}_{C_i} \right) \cup \left(\bigcup_{i \in \{1,\ldots,12\}} \mathcal{R}_{E_i} \right)$$

where

$$\begin{aligned} \mathcal{R}_{C_i} &= \{ b \in \mathcal{R}' \mid \text{The centre of } b \text{ lies in the interior of } C_i \} \\ \mathcal{R}_{E_i} &= \{ b \in \mathcal{R}' \mid \text{The centre of } b \text{ lies in the interior or on the boundary of } E_i \} \end{aligned}$$

In $\mathbb{R}^2$, Löffler and van Kreveld define a chain of arcs at each corner of the initial box, such that every corner of the smallest minimum bounding box of $\mathcal{R}$ must lie on or behind one of these chains. Similarly, for each corner or edge cell, I define a surface such that every corner and every edge of the smallest minimum bounding box must lie on or behind one of the surfaces.

Edge Cells: Clearly for any ball $b \in \mathcal{R}_{E_i}$ with radius ρ_b and centre c_b, a box $B \supset B_0$ intersects b if and only if B intersects the cylinder C_b with radius ρ_b such that C_b is parallel to edge e_i of B_0 and $b \subset C_b$. Now, let h_b and g_b be a pair of planes parallel to the faces of B_0 that intersect in edge e_i such that $c_b \in h_b \cap g_b$ and let H_b be the intersection of the halfspaces bounded by h_g and g_b such that $B_0 \subset H_b$. Now, let $\gamma_b = \partial C_b \cap H_b$ where ∂C_b is the boundary of C_b. It is easy to see that a box $B \supset B_0$ intersects b if and only if edge e_i of B lies in the region Ψ_b bounded by γ_b and the two supporting planes of B_0 that intersect in edge e_i (of B_0) such that $b \subset \Psi_b$.

Consider, for example, a ball $b \in \mathcal{R}_{E_{11}}$ with $c_b = (x_{c_b}, y_{c_b}, z_{c_b})$. In this case, γ_b is given by

$$\gamma_b = \{(x, y, z) \in \mathbb{R}^3 \mid (x - x_{c_b})^2 + (z - z_{c_b})^2 - \rho_b^2 = 0 \wedge x_{c_b} - x \geq 0 \wedge z_{c_b} - z \geq 0\}.$$

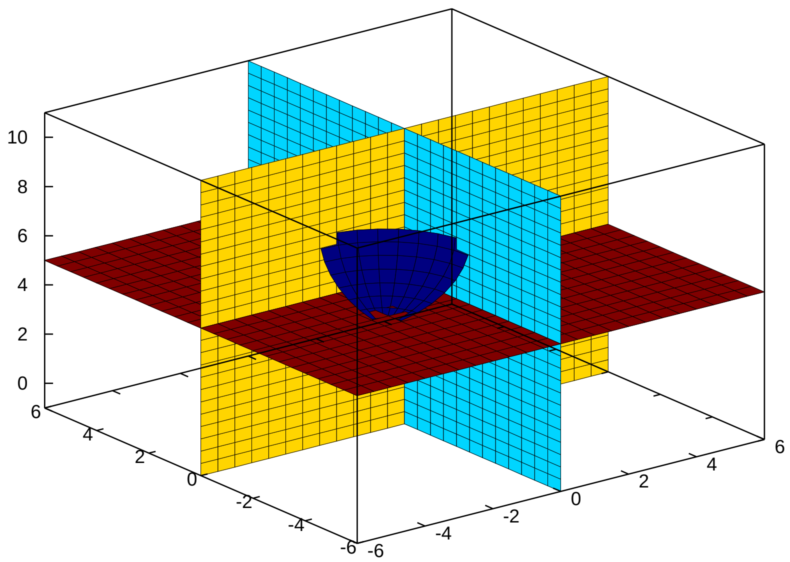

Figure 3.4: γ_b for a ball in a corner cell.

For a ball b in any other edge cell, γ_b can be defined in a similar manner, by replacing $\geq$ with $\leq$ or replacing $(x - x_{c_b})$ or $(z - z_{c_b})$ with $(y - y_{c_b})$ as appropriate.

Now, let

$$\Psi_{E_i} = \bigcap_{b \in \mathcal{R}_{E_i}} \Psi_b \quad \text{and}$$
$$\Gamma_{E_i} = \bigcup_{b \in \mathcal{R}_{E_i}} \gamma_b.$$

Clearly a box $B \supset B_0$ intersects every ball in $\mathcal{R}_{E_i}$ if and only if edge e_i of B lies in Ψ_{E_i}.

Corner Cells: For a corner cell C_i, let E_j, E_k and E_l be the three edge cells adjacent to C_i. For a ball $b \in \mathcal{R}_{C_i}$, we want to define a region Ψ_b such that a box B intersects b if and only if corner q_i of B lies in Ψ_b. Now, for a ball $b' \in \mathcal{R}_{E_j}$, edge e_j of a box B lies in $\Psi_{b'}$ if and only if corner q_i of B lies in $\Psi_{b'}$. So, let $\gamma_{b,x}, \gamma_{b,y}$ and $\gamma_{b,z}$ and $\Psi_{b,x}, \Psi_{b,y}$ and $\Psi_{b,z}$ each be defined exactly as if b lies in one of the edge cells E_j, E_k or E_l. Clearly, if a box $B \supset B_0$ intersects b then corner q_i of B must lie in the intersection of $\Psi_{b,x}, \Psi_{b,y}$ and $\Psi_{b,z}$, however this is not sufficient. To ensure that a box B intersects b if and only if corner q_i of B lies in Ψ_b, a fourth surface γ_b is needed.

Let H_b be the intersection of the halfspaces bounded by the three planes intersecting at c_b, each parallel to one of the planes bounding C_i, such that $B_0 \subset H_b$. Now, let $\gamma_b = \partial b \cap H_b$ and let Ψ_b be the region bounded by $\gamma_b, \gamma_{b,x}, \gamma_{b,y}, \gamma_{b,z}$ and the three planes that bound C_i, such that $b \subset \Psi_b$. Now, a box $B \supset B_0$ intersects b if and only if corner q_i of B lies in Ψ_b.

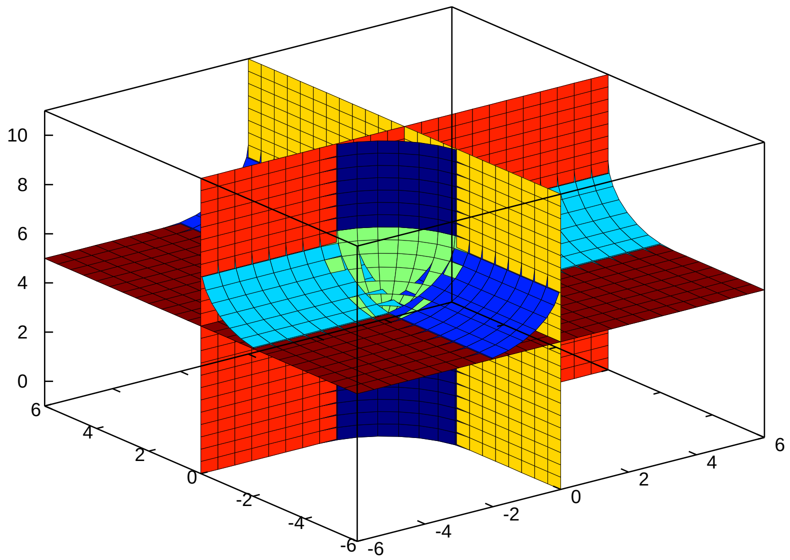

Figure 3.5: $\gamma_b, \gamma_{b,x}, \gamma_{b_y}$ and $\gamma_{b,z}$ for a ball in a corner cell.

For example, for a ball $b \in \mathcal{R}_{C_7}$, $\gamma_b, \gamma_{b,x}, \gamma_{b,y}$ and $\gamma_{b,z}$ are defined as follows:

$$\begin{aligned}
\gamma_b &= \{p = (x, y, z) \in \mathbb{R}^3 \mid \|p - c_b\|^2 - \rho_b^2 = 0 \wedge x_{c_b} - x \geq 0 \wedge y_{c_b} - y \geq 0 \wedge z_{c_b} - z \geq 0\} \\
\gamma_{b,x} &= \{(x, y, z) \in \mathbb{R}^3 \mid (y - y_{c_b})^2 + (z - z_{c_b})^2 - \rho_b^2 = 0 \wedge y_{c_b} - y \geq 0 \wedge z_{c_b} - z \geq 0\} \\
\gamma_{b,y} &= \{(x, y, z) \in \mathbb{R}^3 \mid (x - x_{c_b})^2 + (z - z_{c_b})^2 - \rho_b^2 = 0 \wedge x_{c_b} - x \geq 0 \wedge z_{c_b} - z \geq 0\} \\
\gamma_{b,z} &= \{(x, y, z) \in \mathbb{R}^3 \mid (x - x_{c_b})^2 + (y - y_{c_b})^2 - \rho_b^2 = 0 \wedge x_{c_b} - x \geq 0 \wedge y_{c_b} - y \geq 0\}.
\end{aligned}$$

For a ball in any other corner cell, $\gamma_b, \gamma_{b,x}, \gamma_{b,y}$ and $\gamma_{b,z}$ can be defined in a similar manner, by replacing $\geq$ with $\leq$ or replacing $(x - x_{c_b})$ or $(z - z_{c_b})$ with $(y - y_{c_b})$ as appropriate. See Figure 3.4 and Figure 3.5 for an illustration of the surfaces $\gamma_b, \gamma_{b,x}, \gamma_{b,y}$ and $\gamma_{b,z}$ that bound Ψ_b for a ball b in a corner cell — in these figures, $B_0 = [0, 5] \times [0, 5] \times [0, 5]$ (in the invisible octant).

Now, let

$$\begin{aligned}
\Psi_{C_i} &= \bigcap_{b \in \mathcal{R}_{C_i}} \Psi_b \quad \text{and} \\
\Gamma_{C_i} &= \bigcup_{b \in \mathcal{R}_{C_i}} (\gamma_b \cup \gamma_{b,x} \cup \gamma_{b,y} \cup \gamma_{b,z}).
\end{aligned}$$

Clearly, a box $B \supset B_0$ intersects every ball $b \in \mathcal{R}_{C_i}$ if and only if corner q_i of B lies in $\Psi_{C_i} = \bigcap_{b \in \mathcal{R}_{C_i}} \Psi_b$.

Computing the smallest minimum bounding box: Now a box B will intersect every ball in $\mathcal{R}$ if and only if $B_0 \subseteq B$ and every corner q_i of B lies in the region $\Psi'_i = \Psi_{C_i} \cap \Psi_{E_j} \cap \Psi_{E_k} \cap \Psi_{E_l}$ where E_j, E_k and E_l are the edge cells adjacent to C_i. For each C_i, let U_i be the surface bounding Ψ'_i. U_i

can be computed by rotating the set of surfaces $\Gamma_i' = \Gamma_{C_i} \cup \Gamma_{E_j} \cup \Gamma_{E_k} \cup \Gamma_{E_l}$ such that the vector v_i (as shown in Figure 3.2) coincides with the negative z-axis, computing the lower envelope and then rotating the result back to the original orientation.

Lemma 3.4.5. *If B is the smallest axis-aligned box intersecting each ball in $\mathcal{R}$ then at least one corner on every face of B lies on one of the surfaces U_i.*

Proof. Assume that for some face f of B, no corner lies on a surface U_i. Now, since B intersects every ball in $\mathcal{R}$, every corner q_i of f lies in the interior of the corresponding region Ψ_i'. For each corner q_i of f, let s_i be the line segment orthogonal to f connecting q_i to the surface U_i. Let s_j be the shortest of these line segments and let p be the endpoint of s_j on the surface U_j. Clearly, if we move face f of B such that corner q_j of f coincides with p then the resulting box B' has smaller volume than B and B' intersects every ball in $\mathcal{R}$. This is a contradiction, thus every face of B must have at least one corner q_i lying on the corresponding surface U_i. □

Now, let $B \supset B_0$ be an axis-aligned box, such that at least one corner on each face of B lies on one of the surfaces $U_1, \ldots, U_8$. Let $T \subseteq \{1, \ldots, 8\}$ be the set such that for each $i \in T$, corner q_i of B lies on U_i. It can be verified that $\exists\{i, j\} \subseteq T$ such that corners q_i and q_j are opposite corners of B or $\exists\{i, j, k\} \subseteq T$ such that each pair of corners q_i, q_j; q_j, q_k and q_i, q_k are opposite corners on some face of B.

The smallest axis-aligned box intersecting each $b \in \mathcal{R}$ can now be computed by considering every set $S = \{i, j\}$ such that corners q_i and q_j are opposite corners of B and every $S = \{i, j, k\}$ such that each pair of corners q_i, q_j; q_j, q_k and q_i, q_k are opposite corners on some face of B_0. For each such S, we need to compute the smallest box intersecting each $b \in \mathcal{R}$ such that for each $i \in S$ corner q_i lies on U_i. To compute this box we can consider every possible set $\Phi = \{\phi_i \mid i \in S\}$ where ϕ_i is a 2-dimensional cell of U_i, and compute the smallest box intersecting each ball in $\mathcal{R}$ with corner $q_i \in \phi_i$ for each $\phi_i \in \Phi$.

For any Φ, we can compute the smallest box B_Φ with corner $q_i \in \phi_i$ for each $\phi_i \in \Phi$. If B_Φ intersects every $b \in \mathcal{R}$, then nothing further needs to be done for Φ.

Suppose $\exists b \in \mathcal{R}$ such that $B_\Phi \cap b = \emptyset$. This implies that $\exists l \notin S$ such that corner q_l of B_Φ does not lie in Ψ_l'. Thus, the smallest box B with corner $q_i \in \phi_i$ for each $\phi_i \in \Phi$, such that B intersects every ball in $\mathcal{R}$, has q_l lying on surface U_l.

For each $l \notin S$ and each $\phi_i \in \Phi$, let $\phi_{i,l}$ be the set of all points on U_l such that there exists a box with corner $q_i \in \phi_i$ and corner $q_l \in \phi_{i,l}$. If q_i and q_l are adjacent corners, then $\phi_{i,l}$ is the orthogonal projection of ϕ_i onto U_l. Otherwise if q_i and q_l are opposite corners on the same face f, then $\phi_{i,l}$ is the intersection of U_l with the region between two planes h and g parallel to f where the distance between h and g is as small as possible such that ϕ_i is contained between h and g. Finally, if q_l and q_i do not lie on the same face, then $\phi_{i,l} = U_l$. Now, let $\phi_{\Phi,l} = \bigcap_{\phi_i \in \Phi} \phi_{i,l}$.

If $\phi_{\Phi,l} = \emptyset$, then either every box with corner $q_i \in \phi_i$ for each $\phi_i \in \Phi$ has corner $q_l \in \Psi_l'$, or no such box has corner $q_l \in \Psi_l'$. In this case, there is clearly nothing further to be done for Φ.

On the other hand, if $\phi_{\Phi,l} \neq \emptyset$ and corner q_l of B_Φ does not lie in Ψ_l', then the smallest box intersecting every ball in $\mathcal{R}$ with corner $q_i \in \phi_i$ for each $\phi_i \in \Phi$, has corner $q_l \in \phi_{\Phi,l}$. Thus, for each 2-dimensional cell ϕ_l of ϕ_{Φ_l}, we need to (recursively) compute the smallest box with corner $q_i \in \phi_i$ for each $\phi_i \in \Phi \cup \phi_l$.

Analysis: The upper envelope of a set of n surface patches in $\mathbb{R}^3$ has complexity $O(n^{2+\varepsilon})$ and can be computed in $O(n^{2+\varepsilon})$ expected time for $\varepsilon > 0$ [25]. Thus, the initial box B_0 and the surfaces $U_1, \ldots, U_8$ can all be computed in $O(n^{2+\varepsilon})$ time.

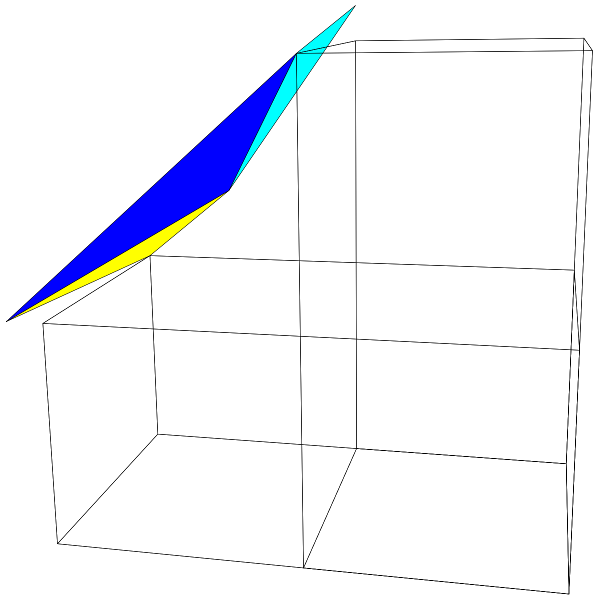

Figure 3.6: A polyhedron that may result in exponential running time when computing the smallest minimum bounding box.

Now, consider $S = \{i, j, k\}$ and let the complexity of a cell ϕ_i of U_i for $i \in S$ be given by $\kappa(\phi_i)$. For any set $\Phi = \{\phi_i, \phi_j, \phi_k\}$ it takes $O(\kappa(\phi_i)\kappa(\phi_j)\kappa(\phi_k))$ time to compute B_Φ and each $\phi_{\Phi,l}$. Now, the sum of all $\kappa(\phi_i)\kappa(\phi_j)\kappa(\phi_k)$ over all cells ϕ_i, ϕ_j, ϕ_k of U_i, U_j and U_k respectively, is $O(n^{6+\varepsilon})$. So, for each $S = \{i, j, k\}$ we spend a total of $O(n^{6+\varepsilon})$ time to compute all B_Φ and $\phi_{\Phi,l}$. This averages out to $O(1)$ time for each $\Phi = \{\phi_i, \phi_j, \phi_k\}$.

Testing whether B_Φ intersects every ball in $\mathcal{R}$, can clearly be done in $O(n)$ time. So if B_Φ intersects every ball in $\mathcal{R}$, or if $\phi_{\Phi,l} = \emptyset$, we spend an average of $O(n)$ time for each $\Phi = \{\phi_i, \phi_j, \phi_k\}$.

Otherwise, if B_Φ does not intersect U_l and $\phi_{\Phi,l} \neq \emptyset$, then we need to consider m_Φ cells ϕ_l, spending an average of $O(n)$ time for each. For fixed ϕ_i, ϕ_j, the sum of m_Φ over all cells ϕ_k of U_k is $O(n^{2+\varepsilon})$. So, for each $S = \{i, j, k\}$ we spend an average of $O(n)$ time for each $\Phi = \{\phi_i, \phi_j, \phi_k\}$. Thus, for each $S = \{i, j, k\}$, it takes a total of $O(n^{7+\varepsilon})$ time to compute the smallest box intersecting every ball in $\mathcal{R}$ with a corner on each of U_i, U_j and U_k.

Similarly, if $S = \{U_i, U_j\}$ it takes $O(n^{5+\varepsilon})$ time to compute the smallest box intersecting each $b \in \mathcal{R}$ with a corner on each of U_i and U_j. Clearly the running time is dominated by the case $|S| = 3$.

Theorem 3.4.6. *The total time required to compute the smallest minimum volume axis-aligned bounding box for a set of imprecise points modelled as balls in $\mathbb{R}^3$ is $O(n^{7+\varepsilon})$ for $\varepsilon > 0$.*

Convex Polytopes

Unfortunately, a similar approach does not work for imprecise points modelled as convex polytopes. Generalising the algorithm of Colley et al. [9] to convex polytopes in higher dimensions, may result in an exponential time algorithm. Consider, for example, the polyhedron P (in Figure 3.6) which is the convex hull of the points $\{(-5, -5, -5), (14, 11, -5), (7, 2, 5), (-2, -5, 15), (14, 14, 15)\}$ and an initial box B_0 with $z^- = 0, z^+ = 10, x^- = 14$ and $y^+ = -5$. Clearly, B_0 does not intersect P. Now, depending on the shape and position of any other polyhedra in $\mathcal{R}$ not intersecting B_0, the smallest minimum bounding box of $\mathcal{R}$ may be such that corner q_2 intersects P but not corner q_3, or vice versa. Figure 3.6 shows P and two boxes, one intersecting P only at corner q_2 and the other intersecting P only at corner q_3.

Thus, to compute the smallest minimum bounding box, we need to consider the case where P contributes to U_2 but not to U_3 and the case where P contributes to U_3 but not to U_2. Since we may need to do this for $O(n)$ polyhedra, we need to consider $O(2^n)$ such combinations, resulting in an algorithm with running time that is exponential in n.

At present, it is not clear how to construct an algorithm for computing the smallest minimum bounding box of a set of imprecise points modelled as convex polytopes, which avoids this issue. Also, it is not clear that such an algorithm is not possible. Thus, this problem remains open.

3.5 Minimum Enclosing Ball

For any point set P in $\mathbb{R}^d$, the smallest ball containing P can be computed in $O(d^2 n + e^{O(\sqrt{d \log d})})$ time [22]. Now, for imprecise points, we want to compute the smallest and largest possible minimum enclosing balls.

3.5.1 Smallest Minimum Enclosing Ball

It is easy to see that for any set of regions $\mathcal{R}$, the smallest minimum enclosing ball is the smallest ball B that intersects every $R \in \mathcal{R}$.

Matoušek et al. [22] note that the problem of finding the smallest ball intersecting a set of closed convex regions in $\mathbb{R}^d$ is an LP-type problem of combinatorial dimension $d+1$. Thus, the smallest minimum enclosing ball of an imprecise point set modelled as a set of convex regions $\mathcal{R}$, can be found in linear time, for fixed d.

The LP-type framework of Matoušek et al. requires a finite set H of constraints and a mapping $w : 2^H \rightarrow W$ where $(W, \leq)$ is a linearly ordered set with some minimum value $-\infty$. In addition, two primitive operations are required: a violation test and a basis computation.

Balls

Let $\mathcal{R} = \{b_1, \ldots, b_n\}$ be a set of balls where b_i has centre c_i and radius ρ_i. Now, we need to compute the smallest $\rho \in \mathbb{R}$ such that $\exists c \in \mathbb{R}^d$ such that $\forall b_i \in \mathcal{R},\ B(c, \rho) \cap b_i \neq \emptyset$ (where $B(c, \rho)$ is the ball with centre c and radius ρ).

Constraints We can let $H = \mathcal{R}$ and write the constraints as

$$\|c_i, c\| - \rho_i \leq \rho,\ \forall i \in \{1, \ldots, n\}$$

and let $W = \mathbb{R}$ where for any $G \subseteq H,\ w(G)$ is smallest $\rho \in \mathbb{R}$ such that $\exists c \in \mathbb{R}^d$ such that $B(c, \rho)$ intersects all balls in G.

Note that ρ is not required to be non-negative. If the solution has $\rho < 0$, then $B(c, |\rho|)$ is the largest ball contained in the common intersection of the balls in $\mathcal{R}$. To see this, note that minimising $\rho < 0$ maximises $|\rho| > 0$ and if $\rho < 0$, then $\|c, c_i\| - \rho \leq r_i$ implies that $\|c, c_i\| + |\rho| \leq r_i$, so the $B(c, |\rho|)$ is contained within ball b_i.

It is easy to see that this problem satisfies the *monotonicity* requirement for an LP-type problem. To see that it satisfies the *locality* requirement, I shall first show that the smallest ball intersecting each of a set of balls is unique.

Lemma 3.5.1 (Uniqueness). *If $\rho^* = w(G)$ then $\exists! c \in \mathbb{R}^d$ such that $B(c, \rho) \cap b_i \neq \emptyset,\ \forall b_i \in G$.*

Proof. By the definition of w, $\exists c \in \mathbb{R}^d$ such that $B(c, \rho) \cap b_i \neq \emptyset,\ \forall b_i \in G$. Now, assume $\exists c' \neq c$ such that $B(c', \rho)$ also intersects all balls in G. Let

$$\begin{aligned} G' &= \{B(c_i, \rho_i + \rho^*) \mid b_i \in G\} \\ C &= \bigcap_{b \in G'} b \end{aligned}$$

Clearly, $\exists b_i \in G$ such that $\|c, c_i\| = \rho^* + \rho_i$ and similarly for c'. Thus c and c' both lie on the boundary of C. Let p be the midpoint of the line segment $\overline{cc'}$. Clearly p lies in the interior of C so $\forall b_i \in G,\ \|p, c_i\| < \rho^* + \rho_i$, which is a contradiction. □

Lemma 3.5.2 (Locality). *Let $F \subseteq G \subseteq \mathcal{R}$ such that $-\infty < w(F) = w(G)$. Then for any $b \in \mathcal{R}$, $w(G) < w(G \cup \{b\}) \implies w(F) < w(F \cup \{b\})$.*

Proof. Let B_G and B_F be the smallest balls intersecting all balls in G and F respectively. By Lemma 3.5.1, $B_G = B_F$. Now

$$\begin{aligned} w(G) < w(G \cup \{b\}) &\implies B_G \cap b = \emptyset \\ &\implies B_F \cap b = \emptyset \\ &\implies w(F) < w(F \cup \{b\}) \end{aligned}$$

□

This LP-type problem is not *basis regular*, since the smallest ball may be determined by fewer than $d+1$ balls. Thus, $O(2^{d+1}n)$ primitive operations are required to compute the smallest ball.

Violation Test For the violation test, we are given a basis $U \subseteq \mathcal{R}$, $|U| \leq d+1$ and a ball $b \notin U$ and we need to determine whether $b \cap B_U = \emptyset$, where $B_U = B(c_U, \rho_U)$ is the smallest ball intersecting each ball in U. This can clearly be done in $O(d)$ time by testing whether $\|c_b, c_U\| - \rho_b \leq \rho_U$.

Basis Computation Given a basis $U \subseteq \mathcal{R}$ ($|U| \leq d+1$) and a ball $b \notin U$ such that b violates U (i.e. $B_U \cap b = \emptyset$), the basis computation must compute a basis for $U \cup \{b\}$.

Fischer and Gärtner [12] developed an LP-type algorithm for computing the smallest ball containing a set of balls. They show that for a basis U' the smallest ball containing all balls in U' can be computed in $O(d^3)$ time. By almost exactly the same argument as theirs (simply replacing their constraints with the ones given above), it can be shown that the smallest ball intersecting all balls in U' can be also computed in $O(d^3)$ time.

Now we can consider each $U'' \subset U$ (with $|U''| < d+1$) in order of increasing size and let $U' = U'' \cup \{b\}$. For each U' we compute $B_{U'}$, the smallest ball intersecting each ball in U'. Now, the first U' found, such that $B_{U'}$ intersects every ball in $U \cup \{b\}$ is returned as the basis for $U \cup \{B\}$. Since up to $O(2^d)$ subsets need to be considered, the basis computation requires $2^{O(d)}$ time.

Theorem 3.5.3. *The smallest ball intersecting each of a set $\mathcal{R}$ of n balls in $\mathbb{R}^d$, can be computed in $2^{O(d)}O(n)$ expected time.*

Convex Polytopes

Let $\mathcal{R} = \{P_1, \ldots, P_n\}$ be a set of convex polytopes in $\mathbb{R}^d$. Again, we need to compute the smallest $\rho \in \mathbb{R}$ such that $\forall P_i \in \mathcal{R},\ \exists B(c, \rho)$ with $B(c, \rho) \cap P_i \neq \emptyset$.

Constraints In general, the smallest ball intersecting a set of convex polytopes is not unique. However, we can ensure uniqueness by requiring the ball with smallest radius and lexicographically smallest centre that intersects all polytopes in $\mathcal{R}$. Again, we let $H = \mathcal{R}$ and write the constraints as:

$$\begin{array}{rcl} \rho & \geq & 0 \quad \text{and} \\ \min_{x \in P_i} \|x, c\| & \leq & \rho,\ \forall i \in \{1, \ldots, n\} \end{array}$$

and let $W = \{-\infty\} \cup \mathbb{R}^{d+1}$, where for $G \subseteq H$, $w(G)$ is the lexicographically smallest point $(\rho, c) \in \mathbb{R}^{d+1}$ such that $B(c, \rho)$ intersects all the polytopes in G.

As in the case for balls, this definition clearly satisfies the *monotonicity* and *locality* requirements for an LP-type problem, however it is not *basis regular*, so $O(2^{d+1}n)$ primitive operations are required.

Violation Test Given a basis $U \subseteq \mathcal{R}$ and a convex polytope $P \notin U$, the violation test needs to determine whether $B_U = B(c_U, \rho_U)$, the smallest ball intersecting each polytope in U, intersects P. To do this we need to test whether

$$\min_{x \in P} \|x, c_U\| \leq \rho_U.$$

Gärtner [13] showed that for two convex polytopes Q_1 and Q_2 with altogether m vertices,

$$\min\{\|x_1, x_2\| \mid x_1 \in Q_1,\ x_2 \in Q_2\}$$

can be computed in $O(d^2 m) + e^{O(\sqrt{d \ln d})}$ expected time. Thus, $\min_{x \in P} \|x, c_U\|$ can be computed in $e^{O(\sqrt{d \ln d})}$ expected time.

Basis Computation Given a basis $U \subseteq \mathcal{R}$ ($|U| \leq d+1$) and a convex polytope $P \notin U$ where P violates U, the basis computation needs to compute a basis for $U \cup \{P\}$.

Now, there are two cases to consider:

1. If the common intersection of all polytopes in $U \cup \{P\}$ is not empty, then the smallest ball intersecting every member of $U \cup \{P\}$ is the ball $B(c, 0)$ where c is the lexicographically smallest point in $\bigcap_{Q \in U \cup \{P\}} Q$. In this case, it is easy to compute a basis for $U \cup \{P\}$ in $e^{O(\sqrt{d \ln d})}$ expected time, using linear programming.

2. If $\bigcap_{Q \in U \cup \{P\}} Q = \emptyset$, we can test every $U'' \subseteq U$ (with $|U''| < d+1$) in order of increasing size, and test whether $B_{U'}$, the smallest ball intersecting every member of $U' = U'' \cup \{P\}$, intersects every member of $U \cup \{P\}$.

   Now, we need to compute the smallest ball intersecting every $Q \in U'$. We can do this by considering every set S consisting of one feature from each polytope in U'. Now the smallest ball intersecting every $\phi \in S$ can be computed in time polynomial in d. For each U', there are $O(2^k)$ sets S to consider (where k is a constant bounding the complexity of the polytopes in $\mathcal{R}$). So, since there are $O(2^d)$ sets U' to consider, the basis computation requires $2^{O(d)}$ time.

Theorem 3.5.4. *The smallest ball intersecting each of a set $\mathcal{R}$ of n convex polytopes in $\mathbb{R}^d$ can be computed in $2^{O(d)}O(n)$ expected time.*

3.5.2 Largest Minimum Enclosing Ball

The smallest ball containing all $R \in \mathcal{R}$ may be determined by more than one point from some $R \in \mathcal{R}$. Thus, to compute the largest minimum enclosing ball for a set of imprecise points modelled as a set of regions $\mathcal{R}$, it is not sufficient to simply compute the smallest ball containing all $R \in \mathcal{R}$.

Löffler and van Kreveld [21] describe algorithms to compute the largest minimum enclosing disc for imprecise points modelled as discs or as squares in $\mathbb{R}^2$. The algorithms I describe here for points modelled as balls or polytopes in $\mathbb{R}^d$ are simple extensions of those algorithms to higher dimensions.

Balls

There are two cases to consider when computing the largest minimum enclosing ball for a set of imprecise points modelled as a set $\mathcal{R} = \{b_1, \ldots, b_n\}$ of a balls in $\mathbb{R}^d$

1. If $\exists b \in \mathcal{R}$ such that $\forall b' \in \mathcal{R} \setminus \{b\},\ b' \subset b$, then the largest minimum enclosing ball for $\mathcal{R}$ is the smallest ball containing a pair of points p and q where q lies on the boundary of b and p lies on the boundary of some $b' \in \mathcal{R} \setminus \{b\}$ such that the distance between p and q is as large as possible. Let $U = \bigcup_{b' \in \mathcal{R} \setminus \{b\}} \partial b'$, then

$$p = \arg\min_{p' \in U} \left(\min_{p'' \in \partial R} \|p', p''\| \right)$$

   where ∂b denotes the boundary of b and

$$q = \arg\max_{q' \in \partial b} \|p, q'\|.$$

   Clearly p, q and the smallest ball containing p and q can be computed in $O(dn)$ time.

2. If $\nexists b \in \mathcal{R}$ such that $\forall b' \in \mathcal{R} \setminus \{b\},\ b' \subset b$, then largest minimum enclosing ball for P is simply the smallest enclosing ball of all the regions. Fischer and Gärtner [12] developed an LP-type algorithm which solves this problem in $2^{O(d)}O(n)$ expected time.

It is easy to distinguish between the above two cases in $O(dn)$ time. Clearly, if b_1 and b_2 are balls in $\mathbb{R}^d$, with centres c_1 and c_2 and radii ρ_1 and ρ_2 respectively, then

$$b_1 \subset b_2 \Leftrightarrow \rho_1 < \rho_2 \text{ and } \|c_1, c_2\| \leq \rho_2 - \rho_1.$$

So, to test whether $\exists b \in \mathcal{R}$ such that $\forall b' \in \mathcal{R} \setminus \{b\},\ b' \subset b$, we first find the ball $b \in \mathcal{R}$ with maximum radius (which can be done in $O(n)$ time) and test whether every ball $b' \in \mathcal{R} \setminus \{b\}$ is contained in R (which can be done in $O(dn)$ time).

Theorem 3.5.5. *The largest minimum enclosing ball for P can be computed in $2^{O(d)}O(n)$ expected time, or $O(dn)$ time if some ball $b \in \mathcal{R}$ contains all other balls in $\mathcal{R}$.*

Convex Polytopes

To compute the largest minimum enclosing ball of a set of imprecise points modelled as a set $\mathcal{R} = \{P_1, \ldots, P_n\}$ of polytopes in $\mathbb{R}^d$, it is sufficient to only consider placements of each p_i at a vertex of P_i.

Lemma 3.5.6. *If we have $f \in \mathcal{F}_{\mathcal{R}}$ and $U \subseteq \mathcal{R}$ such that the minimum enclosing ball B of $f(\mathcal{R})$ is the largest minimum enclosing ball for $\mathcal{R}$ and $f(U)$ is the basis that determines B, then $\forall P \in U,\ f(P)$ is a vertex of P.*

Proof. Note that $\forall P \in \mathcal{R} \backslash U,\ P$ lies entirely in the interior of B. Assume that for some $P \in U,\ f(P)$ is not a vertex of P. Let h be a hyperplane such that $f(U \setminus P) \subset h$ and let s be the line segment orthogonal to h, connecting h to $f(P)$. Now $\exists v \in \mathbb{R}^d$ with $\|v\| = 1$ and an interval $I = (0, k]$ such that for any $\lambda \in I,\ q \in P$ and $q \notin B$ (where $q = f(P) + \lambda v$) and such that the minimum enclosing ball B' of $f(U \setminus P) \cup q$ is a minimum enclosing ball for $\mathcal{R}$ and B' is larger than B. This is a contradiction since B is the largest minimum enclosing ball of $\mathcal{R}$, so $\forall P \in U,\ f(P)$ is a vertex of P. □

Now, let V be the set of all vertices of polytopes in $\mathcal{R}$ and let $U_V \subset V$ be the basis that determines the minimum enclosing ball B_V of V. If each $P \in \mathcal{R}$ contributes at most one vertex to U_V then, by Lemma 3.5.6, B_V will be the largest minimum enclosing ball for $\mathcal{R}$. If, on the other hand, some $P_i \in \mathcal{R}$ contributes two or more vertices to U_V then B_V can not be the largest minimum enclosing ball for $\mathcal{R}$, since that would require placing p_i at two or more distinct positions. However, if P_i contributes more than one vertex to U_V, then it can be shown that exactly one vertex of P_i must be in the basis U that determines the largest minimum enclosing ball B of $\mathcal{R}$.

Lemma 3.5.7. *Let V_i be the set of vertices of polytope P_i and let $V = \bigcup_{P_i \in \mathcal{R}} V_i$. Also, let U_V be the basis determining B_V, the minimum enclosing ball of V and let B be the largest minimum enclosing ball of $\mathcal{R}$, determined by basis $U \subset V$. Now, if $|V_i \cap U_V| > 1$ then $|V_i \cap U| = 1$.*

Proof. Clearly, $|V_i \cap U| \leq 1$. So assume that $V_i \cap U = \emptyset$. Now, since $V_i \cap U_V \neq \emptyset$ and $V_i \cap U = \emptyset,\ \exists p \in P_i$ such that $p \notin B$. Also, since $V_i \cap U = \emptyset$ we are free to place p_i anywhere in P_i, so let $p_i = p$. Now $p_i \notin B$ which is a contradiction, so $|V_i \cap U| = 1$. □

Now, the largest minimum enclosing ball of $\mathcal{R}$ can be computed using Algorithm 3.3 (with Q initially set to $\emptyset$). In line 3, the routine MEB$(V \cup Q)$ computes the minimum enclosing ball of $V \cup Q$; this can be done in $O(d^2 n) + e^{O(\sqrt{d \ln d})}$ time [22]. In line 10, it is possible that there are up to $\lfloor \frac{d+1}{2} \rfloor$ polytopes P_i that satisfy $|V_i \cap U_V| > 1$; in such cases any one of these polytopes may be chosen, the rest will be dealt with at subsequent levels of recursion. Clearly, in line 11 at most a constant number of recursive calls are made and the recursion depth will never be greater than $d + 1$.

Theorem 3.5.8. *Algorithm 3.3 will compute the largest minimum enclosing ball for $\mathcal{R}$ in $O(d^3 n) + e^{O(\sqrt{d \ln d})}$ expected time.*

Algorithm 3.3 LargeMEBP($\mathcal{R}, Q$)

Input: $\mathcal{R}$ a set of polytopes in $\mathbb{R}^d$ and a set of points Q ($|Q| \leq d+1$).
Output: The largest minimum enclosing ball B for $\mathcal{R}$ with all points in Q on the boundary of B.

Let V_i be the set of all vertices of polytope $P_i \in \mathcal{R}$
$V \leftarrow \bigcup_{i \in \{1,\ldots,n\}} V_i$
$B_V \leftarrow \text{MEB}(V \cup Q)$
Let $U_V \subset V \cup Q$ be the basis that determines B_V
if $Q \not\subseteq U_V$ **then**
  return $\emptyset$
else if $\nexists P_i \in \mathcal{R}$ such that $|V_i \cap U_v| > 1$ **then**
  return B_V
else
  Suppose that for $P_i \in \mathcal{R},\ |V_i \cap U_V| > 1$
  return $\max_{v \in V_i}$ LargeMEBP($\mathcal{R} \setminus \{P_i\}, Q \cup \{v\}$)
end if

3.6 Width

The width of a precise point set $P \subset \mathbb{R}^d$ is defined as the smallest distance between two parallel hyperplanes, such that the entire set P is contained between the two hyperplanes.

3.6.1 Minimum Width

For a set of regions $\mathcal{R}$ in $\mathbb{R}^d$, the minimum width of $\mathcal{R}$ can be defined as the smallest $W \geq 0$ such that there exists a pair of parallel hyperplanes

$$h_1 = \{x \in \mathbb{R}^d \mid f(x) = v \cdot x + \delta = 0\} \tag{3.5}$$
$$h_2 = \{x \in \mathbb{R}^d \mid f(x) = W\} \tag{3.6}$$

where $\|v\| = 1$ and

$$\forall R \in \mathcal{R}, \exists x \in R \text{ such that } 0 \leq f(x) \leq W. \tag{3.7}$$

Convex Polytopes

The problem of computing the minimum width of an imprecise point set is closely related to the problem of computing a hyperplane transversal of a set of objects in $\mathbb{R}^d$. It is easy to see that the minimum width of $\mathcal{R}$ is zero if and only if there exists a hyperplane that intersects all regions $R \in \mathcal{R}$.

Avis and Doskas [3] give an algorithm for computing a hyperplane transversal of a set of polytopes in $\mathbb{R}^d$ in $O(n^d)$ time. For each P_i in a set $\mathcal{R} = \{P_1, \ldots, P_n\}$ of polytopes in $\mathbb{R}^d$, they compute a subdivision Γ_i of the space of normal vectors to supporting hyperplanes of P_i. For each d-dimensional cell σ of Γ_i there exists a unique ordered pair $(p_{i\sigma}, q_{i\sigma})$ where $p_{i\sigma}$ and $q_{i\sigma}$ are vertices of P_i, such that for any vector $v \in \sigma$ there exists a pair of parallel hyperplanes

$$h_{iv} = \{x \in \mathbb{R}^d \mid f_{iv}(x) = v \cdot x - v \cdot p_{i\sigma} = 0\}$$
$$h'_{iv} = \{x \in \mathbb{R}^d \mid f'_{iv}(x) = v \cdot q_{i\sigma} - v \cdot x = 0\}$$

such that $\forall x \in P_i,\ f_{iv}(x) \leq 0$ and $f'_{iv}(x) \leq 0$. Clearly, a hyperplane with normal $v \in \sigma$ intersects P_i, if and only if, it intersects the line segment $\overline{p_{i\sigma} q_{i\sigma}}$.

Avis and Doskas compute a hyperplane transversal of $\mathcal{R}$ by computing the subdivision $\Gamma^*(\mathcal{R})$ obtained by overlaying all the subdivisions $\Gamma_i, \ldots, \Gamma_n$ and then using linear programming to search for a hyperplane transversal of $\mathcal{R}$ in each cell of $\Gamma^*(\mathcal{R})$.

A similar approach can be used to compute the minimum width of a set of imprecise points modelled as polytopes in $\mathbb{R}^d$. Consider the subdivision $\Gamma^*(\mathcal{R})$: for each d-dimensional cell σ of $\Gamma^*(\mathcal{R})$, let

$$\begin{aligned} P_\sigma &= \text{CH}(\{p_{1\sigma}, \ldots, p_{n\sigma}\}) \\ Q_\sigma &= \text{CH}(\{q_{1\sigma}, \ldots, q_{n\sigma}\}) \end{aligned}$$

where $\text{CH}(S)$ is the convex hull of S. Now, for any unit vector $v \in \sigma$, let h_v and h'_v be supporting hyperplanes of P_σ and Q_σ, respectively, where

$$\begin{aligned} h_v &= \{x \in \mathbb{R}^d \mid f_v(x) = v \cdot x + \delta_v = 0\} \\ h'_v &= \{x \in \mathbb{R}^d \mid f_v(x) = W_v\} \end{aligned}$$

such that $\forall x \in P_\sigma,\ f_v(x) \geq 0$ and $\forall x \in Q_\sigma,\ f_v(x) \leq W_v$. Clearly, $\forall P \in \mathcal{R},\ \exists x \in P$ such that $0 \leq f_v(x) \leq W_v$. So, if $W_\sigma = \min_{v \in \sigma} W_v$, then the minimum width of $\mathcal{R}$ is given by

$$W = \max\big\{0, \min\{W_\sigma \mid \sigma \text{ is a } d\text{-dimensional cell of } \Gamma^*(\mathcal{R})\}\big\}.$$

Houle and Toussaint [16], showed that to compute the width of a convex polyhedron in $\mathbb{R}^3$, it is sufficient to consider all antipodal vertex-face and edge-edge pairs of features. By a similar argument to theirs, to compute W_σ for a cell σ of $\Gamma^*(\mathcal{R})$, it is sufficient to consider all antipodal feature pairs (ψ, ϕ) where ψ is a k-dimensional feature of P_σ and ϕ is a $(d-1-k)$-dimensional feature of Q_σ, such that a pair of parallel hyperplanes with normal vector $v \in \sigma$ are tangent to P_σ and Q_σ at ψ and ϕ respectively.

Figure 3.7 illustrates the computation of the minimum width of a set of triangles in $\mathbb{R}^2$. $\Gamma^*(\mathcal{R})$ is shown in the top left corner with an arrow indicating the cell σ such that W_σ gives the minimum width of $\mathcal{R}$. The convex hulls P_σ and Q_σ and the pair of parallel lines determining W_σ are also shown.

Theorem 3.6.1. *The minimum width of an imprecise point set modelled as a set $\mathcal{R}$ of polytopes in $\mathbb{R}^d$ can be computed in $O(n^{2d-2})$ time if d is odd or $O(n^{2d-1})$ time if d is even.*

Proof. $\Gamma^*(\mathcal{R})$ is a subdivision induced by an arrangement of $O(n)$ hyperplanes through the origin [3]. Using the algorithm of Edelsbrunner et al. [10], $\Gamma^*(\mathcal{R})$ can be computed in $O(n^d)$ time. Winder [26] showed that an arrangement of $O(n)$ hyperplanes through the origin subdivides $\mathbb{R}^d$ into $O(n^{d-1})$ d-dimensional cells.

Chazelle [7] gives an algorithm to compute the convex hull of n points in $\mathbb{R}^d$ in $O(n \log n + n^{\lfloor d/2 \rfloor})$ time. It is well known that the complexity of the convex hull of n points in $\mathbb{R}^d$ is bounded by $O(n^{\lfloor d/2 \rfloor})$, so by simply considering all possible antipodal pairs, W_σ can be computed in $O(n^d)$ time for even d and $O(n^{d-1})$ time for odd d.

Thus, we need to consider $O(n^{d-1})$ cells σ of $\Gamma^*(\mathcal{R})$. For each cell we spend $O(n^{d-1})$ time (for odd d) or $O(n^d)$ time (for even d) to compute W_σ. □

Balls

If $\mathcal{R}$ is a set of balls, then it is clearly not possible to construct a subdivision of the space of normal vectors as for polytopes. Thus a completely different approach is needed to compute the minimum width of a set of imprecise points modelled as balls.

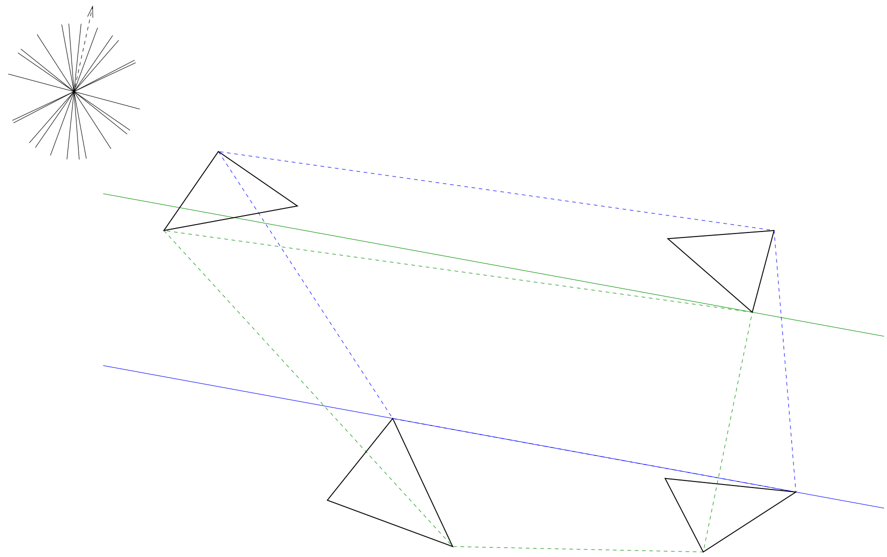

Figure 3.7: Computing the minimum width of a set of polytopes.

I shall show that if the minimum width of $\mathcal{R}$ is zero, then there exists a pair of disjoint sets $S_1, S_2 \subseteq \mathcal{R}$ with $1 \leq |S_1| + |S_2| \leq d$ such that a plane h tangent to each ball in $S_1 \cup S_2$ and separating S_1 from S_2 intersects each ball $b \in \mathcal{R}$. Also, if the minimum width is greater than zero, then there exists a pair of disjoint, nonempty sets $S_1, S_2 \subseteq \mathcal{R}$ with $|S_1| + |S_2| = d + 1$ and a pair of parallel hyperplanes h_1, h_2 each tangent to all the balls in S_1 or S_2 respectively and both separating S_1 from S_2, such that for each $b \in \mathcal{R}$, $\exists p \in b$ such that p lies between h_1 and h_2, and the minimum width of $\mathcal{R}$ is given by the distance between h_1 and h_2.

Zero width: Let $\mathcal{R} = \{b_1, \ldots, b_n\}$ be a set of balls in $\mathbb{R}^d$. For a pair of disjoint subsets $S_1, S_2 \subseteq \mathcal{R}$ with $|S_1| + |S_2| \leq d$ and $S_1 \neq \emptyset$, define H_{S_1,S_2} to be the set of all hyperplanes tangent to every ball in $S_1 \cup S_2$ such that for any $h \in H_{S_1,S_2}$ all balls in S_1 are on the same side of h and all balls in S_2 are on the other side of h.

Lemma 3.6.2. *For any S_1 and S_2 as described above with $|S_1| + |S_2| = k \leq d$, such that H_{S_1,S_2} is not finite, there exists an m-dimensional flat l ($0 \leq m \leq k-1$), passing through the centers of all balls in $S_1 \cup S_2$, such that if $h \in H_{S_1,S_2}$, then any $h' \in H_{S_1,S_2}$ can be obtained by a rotation of h about l and any rotation of h about l yields a hyperplane $h' \in H_{S_1,S_2}$.*

Theorem 3.6.3. *Suppose h is a hyperplane transversal of $\mathcal{R}$. $\exists S_1, S_2 \subseteq \mathcal{R}$ with $S_1 \cap S_2 = \emptyset$, $S_1 \neq \emptyset$, $|S_1| + |S_2| \leq d$ such that either H_{S_1,S_2} is finite and some $h' \in H_{S_1,S_2}$ intersects each ball in $\mathcal{R}$ or H_{S_1,S_2} is not finite and every hyperplane in H_{S_1,S_2} intersects every ball in $\mathcal{R}$.*

Proof. Suppose h is not tangent to any ball in $\mathcal{R}$. Clearly it is possible to translate h to obtain a hyperplane h' tangent to some $b \in \mathcal{R}$ such that h' intersects every ball in $\mathcal{R}$.

Now suppose that for some $1 \leq k < d$ there exists a pair of disjoint sets $S_1, S_2 \subseteq \mathcal{R}$ with $S_1 \neq \emptyset$ and $|S_1| + |S_2| = k$ such that some $h \in H_{S_1,S_2}$ intersects every ball in $\mathcal{R}$. Then, by Lemma 3.6.2, we can rotate h to obtain a hyperplane $h' \in H_{S_1,S_2}$ tangent to some ball $b \in \mathcal{R} \setminus (S_1 \cup S_2)$ such that h' intersects each ball in $\mathcal{R}$, or if no such rotation exists, then every hyperplane in H_{S_1,S_2} intersects every ball in $\mathcal{R}$. The same argument holds in the case where $|S_1| + |S_2| = d$ and H_{S_1,S_2} is not finite. $\square$

By Theorem 3.6.3, we can compute a hyperplane transversal of $\mathcal{R}$ by considering every pair of disjoint sets $S_1, S_2 \subseteq \mathcal{R}$ with $S_1 \neq \emptyset$ and $|S_1| + |S_2| \leq d$. For each such pair, we can determine whether H_{S_1,S_2} is finite, and if it is test each $h \in H_{S_1,S_2}$ to see whether it intersects every ball in $\mathcal{R}$, otherwise if H_{S_1,S_2} is not finite, then test any $h \in H_{S_1,S_2}$ to see whether it intersects every ball in $\mathcal{R}$.

Computing H_{S_1,S_2}: For a pair of balls b_1, b_2, the set of unit normal vectors to hyperplanes in $H_{\{b_1,b_2\},\emptyset}$ is given by the intersection of

$$\{x \in \mathbb{R}^d \mid (c_{b_2} - c_{b_1}) \cdot x - \rho_{b_1} + \rho_{b_2} = 0\}$$

with the unit sphere centred at the origin, where c_b is the centre of b and ρ_b is the radius of b. Similarly, the set of unit normal vectors to all hyperplanes $h \in H_{\{b_1\},\{b_2\}}$ is given by the intersection of

$$\{x \in \mathbb{R}^d \mid (c_{b_2} - c_{b_1}) \cdot x - \rho_{b_1} - \rho_{b_2} = 0\}$$

with the unit sphere centred at the origin in $\mathbb{R}^d$. Thus, for a pair of disjoint sets of balls $S_1, S_2,\ S_1 \neq \emptyset, |S_1| + |S_2| = k \leq d$, we can select any ball $b \in S_1$ and compute the sets

$$\begin{aligned} H_{S_1} &= \big\{\{x \in \mathbb{R}^d \mid (c_{b'} - c_b) \cdot x - \rho_b + \rho_{b'} = 0\} \bigm| b' \in S_1 \setminus \{b\}\big\} \\ H_{S_2} &= \big\{\{x \in \mathbb{R}^d \mid (c_{b'} - c_b) \cdot x - \rho_b - \rho_{b'} = 0\} \bigm| b' \in S_2\big\} \end{aligned}$$

Now, if we let $G = \bigcap_{h \in H_{S_1} \cup H_{S_2}} h$, then the set of unit normal vectors to hyperplanes in H_{S_1,S_2} is given by the intersection of G with the unit sphere centred at the origin in $\mathbb{R}^d$. Using Gaussian Elimination, we can easily find a point $p \in G$ and a set of vectors spanning G in $O(k^3 + dk^2)$ time. From this, we can compute an orthonormal basis for G in $O(d^3)$ time. Now the point $q \in G$ such that $\forall q' \in G, \|q'\| \geq \|q\|$ can be computed in $O(d^2)$ time as $q = p - AA^{\mathrm{T}}p$, where the rows of matrix A form an orthonormal basis for G. Now, H_{S_1,S_2} is finite if and only if $A \in \mathbb{R}^{1\times d}$ (i.e. G is a line), in which case $|H_{S_1,S_2}| \leq 2$ or $\|q\| = 1$ (G is tangent to the unit sphere centred at the origin) in which case $|H_{S_1,S_2}| = 1$. Thus, if H_{S_1,S_2} is finite each $h \in H_{S_1,S_2}$ can be computed in $O(d)$ time or if H_{S_1,S_2} is infinite, an arbitrary $h \in H_{S_1,S_2}$ can also be computed in $O(d)$ time.

Note that this method of computing H_{S_1,S_2} assumes that no two balls have the same centre. This assumption does not cause any problems, since clearly if $\exists b \in \mathcal{R}$ such that for some $b' \in \mathcal{R},\ b' \subseteq b$ then the minimum width of $\mathcal{R}$ is equal to the minimum width of $\mathcal{R} \setminus \{b\}$, thus before computing the minimum width, we can preprocess $\mathcal{R}$ to remove any ball b such that $\exists b' \in \mathcal{R} \setminus \{b\}$ with $b' \subseteq b$.

Positive Width: For a pair of non-empty disjoint subsets $S_1, S_2 \subseteq \mathcal{R}$, with $|S_1| + |S_2| > d$, define H_{S_1,S_2} to be the set of all pairs of parallel hyperplanes (h_1, h_2) tangent to each ball in S_1 and S_2 respectively such that both h_1 and h_2 separate S_1 from S_2.

Theorem 3.6.4. *If $\mathcal{R}$ is a set of balls in $\mathbb{R}^d$ with minimum width $W > 0$, then there exists a pair of non-empty disjoint sets $S_1, S_2 \subseteq \mathcal{R}$ with $|S_1| + |S_2| > d$ such that H_{S_1,S_2} is finite and the minimum width of P is given by the distance between a pair of hyperplanes $(h_1, h_2) \in H_{S_1,S_2}$.*

Proof. Let h_1 and h_2 be a pair of parallel hyperplanes as in (3.5) and (3.6) above. Also, let

$$\begin{aligned} S_1 &= \{b \in \mathcal{R} \mid \forall x \in b,\ f(x) \leq 0\} \\ S_2 &= \{b \in \mathcal{R} \mid \forall x \in b,\ f(x) \geq W\} \end{aligned}$$

Clearly $S_1 \cap S_2 = \emptyset$. Now assume that $S_2 = \emptyset$, then $\forall b \in \mathcal{R}$, $\exists p \in b$ such that $f(p) < W$. This contradicts the fact that the minimum width of $\mathcal{R}$ is W, so $S_2 \neq \emptyset$. By a similar argument, it can be shown that $S_1 \neq \emptyset$.

Now, let

$$\begin{aligned} S'_1 &= \{b \cap h_1 \mid b \in S_1\} \\ S'_2 &= \{b \cap h_2 \mid b \in S_2\}. \end{aligned}$$

Assume there exists a hyperplane h such that $S'_1 \cup S'_2 \subset h$. Let $l_1 = h_1 \cap h$ and $l_2 = h_2 \cap h$. Now, $\forall b \in \mathcal{R} \setminus S_1$, $\exists p \in b$ such that $f(p) > 0$ and $\forall b \in \mathcal{R} \setminus S_2$, $\exists p \in b$ such that $f(p) < W$. Thus, $\exists \theta$ such that rotating h_1 and h_2 by an angle of θ about l_1 and l_2 respectively, gives a new pair of parallel hyperplanes

$$\begin{aligned} h_3 &= \{x \in \mathbb{R}^d \mid g(x) = u \cdot x + \gamma = 0\} \\ h_4 &= \{x \in \mathbb{R}^d \mid g(x) = V\} \end{aligned}$$

where $\|u\| = 1, V < W$ and $\forall b \in \mathcal{R}, \exists p \in b$ such that $0 \leq g(p) \leq V$. This is a contradiction, since W is the minimum width of $\mathcal{R}$. Thus, for any hyperplane h, $\exists p \in S'_1 \cup S'_2$ such that $p \notin h$. This implies that $|S_1| + |S_2| > d$ and H_{S_1,S_2} is finite. Also, clearly $(h_1, h_2) \in H_{S_1,S_2}$. □

Theorem 3.6.5. *Let $\mathcal{R} = \{b_1, \ldots, b_n\}$ be a set of balls in $\mathbb{R}^d$, with minimum width $W > 0$. $\exists S_1, S_2$ satisfying Theorem 3.6.4, such that $|S_1| + |S_2| = d + 1$.*

Proof. Let $\mathcal{R}_1, \mathcal{R}_2 \subseteq \mathcal{R}$ be a pair of sets satisfying Theorem 3.6.4 and let h_1 and h_2 be a pair of parallel hyperplanes as in (3.5) and (3.6), with $(h_1, h_2) \in H_{\mathcal{R}_1,\mathcal{R}_2}$. Now, let $S'_1 \subseteq \mathcal{R}_1$, $S'_2 \subseteq \mathcal{R}_2$ such that $|S'_1| + |S'_2| = d$ and let $Q_1 = \{b \cap h_1 \mid b \in S_1\}$ and $Q_2 = \{b \cap h_2 \mid b \in S_2\}$ such that there exists a unique hyperplane h with $Q_1 \cup Q_2 \subset h$. Now $\exists b \in (\mathcal{R}_1 \cup \mathcal{R}_2) \setminus (S_1 \cup S_2)$ such that $(b \cap h_1) \cup (b \cap h_2) \not\subseteq h$. Without loss of generality, assume $b \in \mathcal{R}_1$ and let $S_1 = S'_1 \cup \{b\}$ and $S_2 = S'_2$. Clearly S_1 and S_2 satisfy Theorem 3.6.4. □

Figure 3.8 shows a set of balls in $\mathbb{R}^3$ and the pair of parallel planes determining the minimum width. Note that there are two balls tangent to and above the upper plane and two balls tangent to and below the lower plane.

Now, if the minimum width of $\mathcal{R}$ is known to be greater than zero, it can be computed by searching for a pair of sets S_1 and S_2 satisfying Theorem 3.6.5. Clearly if S_1 and S_2 satisfy Theorem 3.6.5, then $|H_{S_1,S_2}| \leq 2$. Thus, to compute the minimum width, we can consider all pairs of non-empty disjoint $S_1, S_2 \subseteq \mathcal{R}$ with $|S_1| + |S_2| = d + 1$, determine whether H_{S_1,S_2} is finite, and if it is, test whether $\exists (h_1, h_2) \in H_{S_1,S_2}$ such that each $b \in \mathcal{R}$ has at least one point lying between h_1 and h_2. The minimum distance between all such h_1 and h_2 over all pairs S_1, S_2 will give the minimum width of $\mathcal{R}$.

Computing H_{S_1,S_2}: For $|S_1| + |S_2| = d + 1$, H_{S_1,S_2} can be computed in a manner very similar to when $|S_1| + |S_2| \leq d$, except that now we choose two balls $b_1 \in S_1$ and $b_2 \in S_2$ and let

$$\begin{aligned} H_{S_1} &= \big\{\{x \in \mathbb{R}^d \mid (c_{b'} - c_{b_1}) \cdot x - \rho_{b_1} + \rho_{b'} = 0\} \bigm| b' \in S_1 \setminus \{b_1\}\big\} \\ H_{S_2} &= \big\{\{x \in \mathbb{R}^d \mid (c_{b_2} - c_{b'}) \cdot x - \rho_{b_2} - \rho_{b'} = 0\} \bigm| b' \in S_2 \setminus \{b_2\}\big\}. \end{aligned}$$

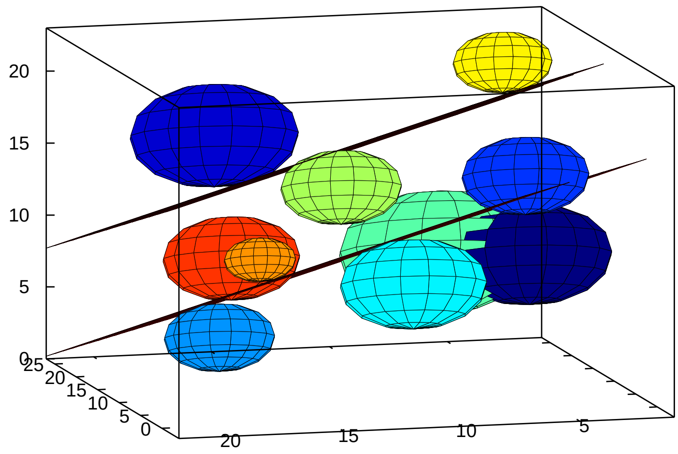

Figure 3.8: The minimum width of a set of balls in $\mathbb{R}^3$.

Algorithm for Minimum Width: We can now combine the results from this section to obtain an algorithm (Algorithm 3.4) for computing the minimum width of a set of imprecise points modelled as a set $\mathcal{R}$ of balls in $\mathbb{R}^d$.

Preprocessing $\mathcal{R}$ requires $O(n^2 d)$ time. Then, there are $O(n^i)$ subsets S_1 of size i, $\forall 1 \leq i \leq d$, for each of which there are $O(n^j)$ subsets $S_2 \subseteq \mathcal{R} \setminus S_1$, $|S_2| = j$, $\forall j \leq d + 1 - i$. This gives $O(dn^{d+1})$ pairs S_1, S_2. For each such pair, we can compute H in $O(d^3)$ time. Then, for each of the, at most two pairs $(h_1, h_2) \in H$, we can test whether each $b \in \mathcal{R}$ has at least one point between h_1 and h_2 in $O(nd)$ time.

Theorem 3.6.6. *The minimum width of a set of imprecise points modelled as a set $\mathcal{R}$ of balls in $\mathbb{R}^d$ can be computed in $O(d^2 n^{d+2})$ time.*

3.6.2 Maximum Width

The maximum width of an imprecise point set modelled as a set of regions $\mathcal{R} = \{R_1, \ldots, R_n\}$ in $\mathbb{R}^d$ is the maximum width over all point sets $P = \{p_1, \ldots, p_n\}$ such that $p_i \in R_i$. That is, the maximum width W_{max} of $\mathcal{R}$ is given by:

$$W_{\text{max}} = \max_{f \in \mathcal{F}_{\mathcal{R}}} \text{Width}(f(\mathcal{R})) \tag{3.8}$$

Note that the maximum width is not given by the smallest distance between a pair of parallel hyperplanes h_1 and h_2 such that $\mathcal{R}$ is contained between h_1 and h_2.

Computing the maximum width of $\mathcal{R}$ is made difficult by the fact that it is not sufficient to only consider subsets of $\mathcal{R}$: there do exist sets of points such that removing any one point reduces the width. Furthermore, it is possible that the maximum width is simultaneously determined by several pairs of parallel hyperplanes [21].

Algorithm 3.4 Minimum Width for balls.

Input: A set $\mathcal{R}$ of balls in $\mathbb{R}^d$.
Output: The minimum width W of $\mathcal{R}$.

Preprocess $\mathcal{R}$ to remove any $b \in \mathcal{R}$ such that $\exists b' \in \mathcal{R} \setminus \{b\},\ b' \subseteq b$
$W \leftarrow \infty$
for each $S_1 \subseteq \mathcal{R},\ 1 \le |S_1| \le d$ **do**
  for each $S_2 \subseteq \mathcal{R} \setminus \{S_1\},\ |S_2| \le d + 1 - |S_1|$ **do**
    $k \leftarrow |S_1| + |S_2|$
    if $H_{S_1,S_2} = \emptyset$ **then**
      $H' \leftarrow \emptyset$
    else if H_{S_1,S_2} is finite **then**
      $H' \leftarrow H_{S_1,S_2}$
    else if $k > d$ **then**
      $H' \leftarrow \emptyset$
    else
      Let h be any hyperplane in H_{S_1,S_2}
      $H' \leftarrow \{h\}$
    end if
    if $k \le d$ **then**
      $H \leftarrow \{(h,h) \mid h \in H'\}$
    else
      $H \leftarrow H'$
    end if
    for each $(h_1, h_2) \in H$ **do**
      Let W' be the distance between h_1 and h_2
      if $W' < W$ **then**
        if $\forall b \in \mathcal{R} \setminus (S_1 \cup S_2), \exists p \in b$ such that p lies between h_1 and h_2 **then**
          $W \leftarrow W'$
        end if
      end if
      if $W = 0$ **then**
        return W
      end if
    end for
  end for
end for
return W

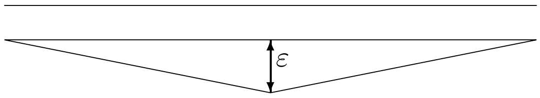

Figure 3.9: Replacing a line segment with a triangle.

Convex Polytopes

Löffler and van Kreveld [21] proved that computing the maximum width of a set of imprecise points modelled as line segments in $\mathbb{R}^2$ is NP-Hard. They showed that for any instance of SAT and some $W > 0$, it is possible to construct a set L of line segments such that the maximum width of L is W if and only if the SAT instance can be satisfied.

Theorem 3.6.7. *For any instance of SAT and any $W > 0$, it is possible to construct a set $\mathcal{R}$ of convex polytopes in $\mathbb{R}^d$ such that the maximum width of $\mathcal{R}$ is W if and only if the SAT instance can be satisfied.*

Proof. In $\mathbb{R}^2$ we can use the same construction Löffler and van Kreveld used for line segments, and replace each line segment with a triangle of height ε (for some very small ε), as shown in Figure 3.9. The resulting set of convex polygons has maximum width W if and only if the SAT instance can be satisfied.

Now, assume that for each $2 \leq i \leq d$ we can construct a set $\mathcal{R}_i$ of convex polytopes in $\mathbb{R}^i$ that has maximum width W if and only if the SAT instance can be satisfied. Now we can construct a set $\mathcal{R}_{d+1}$ of convex polytopes in $\mathbb{R}^{d+1}$ as follows:

$$\mathcal{R}_{d+1} = \{P \times [-2W, 2W] \mid P \in \mathcal{R}_d\}$$

The resulting $\mathcal{R}_{d+1}$ will have maximum width W if and only if $\mathcal{R}_d$ has maximum width W. □

Balls

Löffler and van Kreveld [21] were unable to obtain any result for computing the maximum width of a set of imprecise points modelled as discs in $\mathbb{R}^2$. I was also not able to obtain any result for points modelled as balls in higher dimensions. Thus, this problem remains open.

Chapter 4

Discussion and Conclusion

4.1 Discussion

For the most part, I focused only on establishing the existence of polynomial time algorithms for computing upper and lower bounds on measures of imprecise point sets, rather than attempting to find algorithms with optimal running times. In particular, it may be possible to obtain faster algorithms for computing the largest diameter and minimum width. Note however that the algorithms for solving these problems in $\mathbb{R}^2$ rely on using the rotating callipers method, which does not generalise to higher dimensions. Thus, it is unlikely that running times similar to those obtained by Löffler and van Kreveld [21] can be obtained in higher dimensions.

In the case of the smallest closest pair, my algorithms require quadratic time, while Löffler and van Kreveld obtained $O(n \log n)$ time algorithms for imprecise points in $\mathbb{R}^2$. They achieved this result through the use of Voronoi diagrams. It is however well known that Voronoi diagrams in $\mathbb{R}^3$ can have complexity $\Omega(n^2)$. Also, the divide and conquer approach used by Bentley [5] relies on knowing the exact location of every point, and is thus not applicable to imprecise point sets (except when the points are modelled as unit balls).

My algorithms for the minimum enclosing ball problem as well as the largest minimum volume axis aligned bounding box, are simple extensions of those of Löffler and van Kreveld to higher dimensions. As in the case of precise points, these problems do get slightly harder as the dimension increases but can still be solved in time linear in the number of points.

The smallest minimum volume axis aligned bounding box however, becomes significantly harder as the dimension increases. In $\mathbb{R}^2$, the minimum area rectangle intersecting a set of convex polygons can be computed in $O(n \log n)$ time [9] and the minimum area rectangle intersecting a set of discs can be computed in $O(n^2)$ time [21]. For balls in $\mathbb{R}^3$ however, the running time increases to $O(n^{7+\varepsilon})$, due to the fact that it is necessary to compute boxes with three corners lying on surfaces of complexity $O(n^{2+\varepsilon})$ while in $\mathbb{R}^2$ it is only necessary to consider two corners lying on curves of linear complexity. Somewhat disappointing however, is that while in $\mathbb{R}^2$, the problem is easier to solve for convex polygons than for discs, I was not able to obtain a polynomial algorithm to compute the smallest box intersecting a set of convex polytopes in higher dimensions.

4.2 Open Problems

Although I successfully solved most of the problems I studied, I was not able to obtain any results on computing the minimum diameter of a set of imprecise points in $\mathbb{R}^d$. I was also unable to obtain any results on computing the maximum width of a set of imprecise points modelled as balls. Given that the maximum width problem is NP-Hard for points modelled as convex polytopes, it seems reasonable to expect that the problem would also be hard for other models. There are many similarities between the minimum diameter, maximum width and largest closest pair problems, thus, it would not be surprising if the minimum diameter problem also turns out to be NP-Hard. I was also unable to solve the problem of computing the smallest minimum volume axis aligned bounding box for points modelled as convex polytopes — it would be interesting to see whether a polynomial time algorithm can be developed for this problem.

Apart from the few problems I was unable to solve, there are many other open problems that I did not consider. It would be interesting to see whether the work of Löffler [18] on convex hulls for imprecise points in $\mathbb{R}^2$ can be extended to higher dimensions. Other interesting problems include the computation of a minimum width annulus enclosing a set of points as well as the minimum bounding box of arbitrary orientation.

It would also be interesting to consider models of imprecision other than balls or convex polytopes. For example, if imprecise points modelled as discs in $\mathbb{R}^2$ are combined with imprecise elevation data modelled as intervals, one obtains points modelled as axis parallel cylinders in $\mathbb{R}^3$. GPS devices, compute the coordinates of a point in $\mathbb{R}^3$ by computing the distance to at least three satellites and computing a point common to three (or more) spheres. Thus imprecision of GPS coordinates may be modelled by the intersection of spherical shells.

References

[1] M. Abellanas, F. Hurtado, and P.A. Ramos. Structural tolerance and Delaunay triangulation. *Information Processing Letters*, 71:221–227, 1999.

[2] I. Averbakh and S. Bereg. Facility location problems with uncertainty on the plane. *Discrete Optimization*, 2:3–34, 2005.

[3] D. Avis and M. Doskas. Algorithms for high dimensional stabbing problems. *Discrete Applied Mathematics*, 27:39–48, 1990.

[4] D. Bandyopadhyay and J. Snoeyink. Almost-Delaunay Simplices: Nearest Neighbor Relations for Imprecise Points. In *PROC. 15th ACM-SIAM Symposium on Discrete Algorithms*, pages 410–419, 2004.

[5] J.L. Bentley. Multidimensional divide-and-conquer. *Communications of the ACM*, 23(4):214–229, 1980.

[6] S. Cabello. Approximation algorithms for spreading points. *Journal of Algorithms*, 62:49–73, 2007.

[7] B. Chazelle. An Optimal Convex Hull Algorithm in Any Fixed Dimension. *Discrete and Computational Geometry*, 10:377–409, 1993.

[8] B. Chazelle, H. Edelsbrunner, L. Guibas, and M. Sharir. Diameter, Width, Closest Line Pair, and Parametric Searching. *Discrete and Computational Geometry*, 10:183–196, 1993.

[9] P. Colley, H. Meijer, and D. Rappaport. Optimal nearly-similar polygon stabbers of convex polygons. In *Proceedings of the 6th Canadian Conference on Computational Geometry*, pages 269–274, 1994.

[10] H. Edelsbrunner, J. O'Rourke, and R. Seidel. Constructing Arrangements of Lines and Hyperplanes with Applications. *SIAM Journal on Computing*, 15(2):341–363, 1986.

[11] J. Fiala, J. Kratochvíl, and A. Proskurowski. Systems of distant representatives. *Discrete Applied Mathematics*, 145:306–316, 2005.

[12] K. Fischer and B. Gärtner. The Smallest Enclosing Ball of Balls: Combinatorial Structure and Algorithms. In *SCG '03: Proceedings of the nineteenth annual symposium on Computational Geometry*, pages 292–301, New York, USA, 2003. ACM Press.

[13] B. Gärtner. A subexponential algorithm for abstract optimization problems. *SIAM Journal on Computing Science*, 24:1018–1035, 1995.

[14] L. Guibas, D. Salesin, and Stolfi J. Epsilon Geometry: Building Robust Algorithms from Imprecise Computations. In *Proceedings of the 5th Annual ACM Symposium on Computational Geometry*, pages 208–217, 1989.

[15] L. Guibas, D. Salesin, and Stolfi J. Constructing Strongly Convex Approximate Hulls with Inaccurate Primitives. *Algorithmica*, 9:534–560, 1993.

[16] M.E. Houle and G.T. Toussaint. Computing the Width of a Set. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(5):761–765, 1988.

[17] A. A. Khanban and A. Edalat. Computing Delaunay Triangulation with Imprecise Input Data. In *Proc. 15th Canadian Conference on Computational Geometry*, pages 94–97, 2003.

[18] M. Löffler. Smallest and Largest Convex Hulls for Imprecise Points. Master's thesis, Department of Computer Science, Utrecht University, Utrecht, The Netherlands, 2005.

[19] M. Löffler and J. Snoeyink. Delaunay Triangulations of Imprecise Points in Linear Time after Preprocessing. In *Annual Symposium on Computational Geometry*, pages 298–304. ACM, 2008.

[20] M. Löffler and M. van Kreveld. Largest and Smallest Convex Hulls for Imprecise Points. *Algorithmica*. To Appear.

[21] M. Löffler and M. van Kreveld. Largest Bounding Box, Smallest Diameter, and Related Problems on Imprecise Points. Technical Report UU-CS-2007-025, Department of Information and Computing Sciences, Utrecht University, 2007.

[22] J. Matoušek, M. Sharir, and E. Welzl. A subexponential bound for linear programming. *Algorithmica*, 16:498–516, 1996.

[23] T. Nagai and N. Tokura. Tight Error Bounds of Geometric Problems on Convex Objects with Imprecise Coordinates. In *Japanese Conference on Discrete and Computational Geometry*, pages 252–263, 2000.

[24] J. Schwerdt, M. Smid, J. Majhi, and R. Janardan. Computing the width of a three-dimensional point set: an experimental study. In *Proceedings WAE'98*, pages 62–73, 1998.

[25] M. Sharir. Almost Tight Upper Bounds for Lower Envelopes in Higher Dimensions. *Discrete and Computational Geometry*, 12:327–345, 1994.

[26] R.O. Winder. Partitions of N space by hyperplanes. *SIAM Journal on Applied Mathematics*, 14: 811–818, 1966.