Probabilistic methods for the analysis of algorithms on random tessellations

Ross Hemsley

Université de Nice – Sophia Antipolis

École Doctorale STIC

SCIENCES ET TECHNOLOGIES DE L'INFORMATION ET DE LA COMMUNICATION

Thèse

Pour l'obtention du grade de

Docteur en Sciences

de l'Université de Nice–Sophia Antipolis

Mention : informatique

présentée et soutenue par

Ross Hemsley

Méthodes probabilistes pour l'analyse des algorithmes sur les tesselations aléatoires

directeur de these : Olivier Devillers

soutenue le 16 décembre 2014

Jury :	Luc Devroye	–	McGill University
	Jean-François Marckert	–	Université de Bordeaux
Examinateurs :	Nicolas Broutin	–	Inria Paris – Rocquencourt
	Kevin Buchin	–	Technische Universiteit Eindhoven
	Pierre Calka	–	Université de Rouen
	Olivier Devillers	–	Inria Nancy – Grand est

Université de Nice – Sophia Antipolis

École Doctorale STIC

SCIENCES ET TECHNOLOGIES DE L'INFORMATION ET DE LA COMMUNICATION

PhD Thesis

to obtain the title of

Doctor of Science

of the University of Nice – Sophia Antipolis

Specialty: Computer Science

prepared and defended by

Ross Hemsley

Probabilistic methods for the analysis of algorithms on random tessellations

advisor: Olivier Devillers

defended on Dec. 16, 2014

Reviewers:	Luc Devroye	–	McGill University
	Jean-François Marckert	–	Université de Bordeaux
Examiners:	Nicolas Broutin	–	Inria Paris – Rocquencourt
	Kevin Buchin	–	Technische Universiteit Eindhoven
	Pierre Calka	–	Université de Rouen
	Olivier Devillers	–	Inria Nancy – Grand est

Acknowledgements

This work wouldn't be here without the work of many people. I must in particular thank my supervisor, Olivier for his sage-like intuition for all things geometric, his deep knowledge of Computer Science, and of course, his ability to politely tolerate the dubious approximation to *français* which I have come to employ whilst communicating with him. I wish also to thank my excellent collaborators, Nicolas Broutin and Nicolas Chenavier for their constant attention to detail and for introducing me to the techniques of stochastic geometry. Lastly, but surely not least, I wish to thank my various French *colocataires* Clément, Simon and Florian who have contributed enourmously to my learning of the French language[1], and of course, Hélène for her consistant encouragement and support.

[1] Albeit somewhat restricted to a subset of the language of questionable utility in polite company.

Résumé

Dans cette thèse, nous exploitons les outils de la théorie des probabilités et de la géométrie stochastique pour analyser des algorithmes opérant sur les tessellations. Ce travail est divisé entre deux thèmes principaux, le premier traite de la navigation dans une tessellation de Delaunay et dans son dual, le diagramme de Voronoï avec des implications pour les algorithmes de localisation spatiales et de routage dans les réseaux en ligne. Nous proposons deux nouveaux algorithmes de navigation dans la triangulation de Delaunay, que nous appelons Pivot Walk et Cone Walk. Pour Cone Walk, nous fournissons une analyse en moyenne détaillée avec des bornes explicites sur les propriétés de la pire marche possible effectué par l'algorithme sur une triangulation de Delaunay aléatoire d'une région convexe bornée. C'est un progrès significatif car dans l'algorithme Cone Walk, les probabilités d'utiliser un triangle ou un autre au cours de la marche présentent des dépendances complexes, dépendances inexistantes dans d'autres marches.

La deuxième partie de ce travail concerne l'étude des propriétés extrémales de tessellations aléatoires. En particulier, nous dérivons les premiers et derniers statistiques d'ordre pour les boules inscrites dans les cellules d'un arrangement de droites Poissonnien ; ce résultat a des implications par exemple pour le hachage respectant la localité. Comme corollaire, nous montrons que les cellules minimisant l'aire sont des triangles.

Abstract

In this thesis, we leverage the tools of probability theory and stochastic geometry to investigate the behaviour of algorithms on geometric tessellations of space. This work is split between two main themes, the first of which is focused on the problem of navigating the Delaunay tessellation and its geometric dual, the Voronoi diagram. We explore the applications of this problem to point location using walking algorithms and the study of online routing in networks. We then propose and investigate two new algorithms which navigate the Delaunay triangulation, which we call Pivot Walk and Cone Walk. For Cone Walk, we provide a detailed average-case analysis, giving explicit bounds on the properties of the the worst possible path taken by the algorithm on a random Delaunay triangulation in a bounded convex region. This analysis is a significant departure from similar results that have been obtained, due to the difficulty of dealing with the complex dependence structure of localised navigation algorithms on the Delaunay triangulation.

The second part of this work is concerned with the study of extremal properties of random tessellations. In particular, we derive the first and last order-statistics for the inballs of the cells in a Poisson line tessellation. This result has implications for algorithms involving line tessellations, such as locality sensitive hashing. As a corollary, we show that the cells minimising the area are triangles.

Contents

Chapter 1

Introduction

Computational geometry is a field in computer science of ever-increasing importance, with many practical problems having natural geometric interpretations, and a continually growing repository of spatial data being used in areas as diverse as gaming and scene reconstruction. With this continual increase in data and applications comes an increased demand for efficient algorithms and more precise analysis of existing algorithms. In an era where a 1% efficiency gain in parts of Facebook's computing infrastructure can easily pay the salary of a developer over the course of a year in power savings, and extremal events at the tail end of performance can come to dominate the waiting time for queries at Google [42], it is increasingly clear that the detailed analysis of algorithms and data structures has an important part to play in a modern data-driven world. Therefore, whilst the problems that we shall consider have clear theoretical interest, their solutions can often provide important technical insights for implementers and engineers developing solutions to problems at scale, aiding design decisions and increasing efficiency.

Tessellations of space In this dissertation we shall focus on results applied to *tessellations*. We define a tessellation of space to be a partition of that space into a collection of convex regions or *cells* whose interiors do not intersect. Tessellations are a core concept in computational geometry, since many different geometric constructions may be modelled using them. As an example, data structures representing spatial tessellations have many practical uses, in which case it can be important to understand the behaviour of algorithms applied to those tessellations so that the data structures can be used efficiently. Other uses of tessellations may be more theoretical. For example, we can sometimes use results on tessellations to prove bounds for algorithms. In the literature, a small number of tessellations have emerged as being particularly important. Our work will focus on three of these, which we introduce now.

The Delaunay and Voronoi tessellations The Delaunay tessellation, originally proposed by the Russian physicist (and mountain climber) Delaunay [43] is defined in the Euclidean plane, $\mathbf{R}^2$ to be a partition of space into triangles such that no circumdisc of any

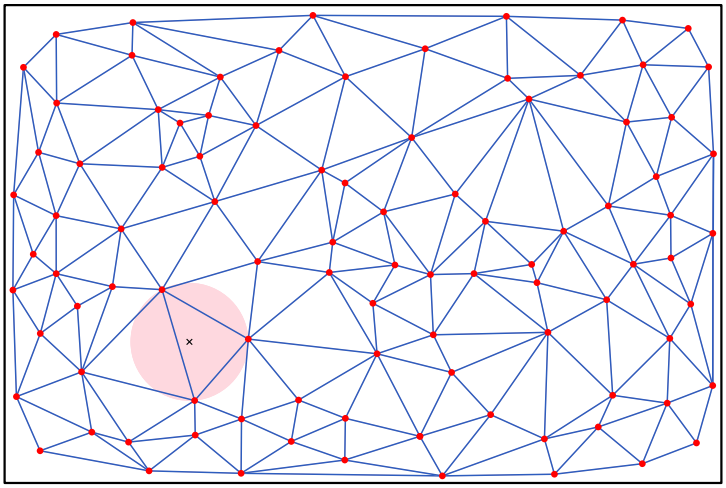


Figure 1.1 – A Delaunay tessellation of points in a rectangle in $\mathbf{R}^2$. The interior of the circumdisc for every triangle contains no points in the process.

triangle contains any vertices in its interior (see Figure 1.1.) Higher dimensional Delaunay tessellations may be constructed by replacing the triangles with their d dimensional counterparts, known as *d-simplices*. We will often refer to the Delaunay tessellation in any dimension as a *triangulation* due to this property. The Delaunay triangulation may be generalised to arbitrary metric spaces, though the cases of $\mathbf{R}^2$ and $\mathbf{R}^3$ with the Euclidean distance are the most used in practical applications. The Delaunay triangulation and its geometric dual, the *Voronoi Diagram*, have found uses in hundreds of practical problems including, but not limited to; routing, surface reconstruction, interpolation, meshing, the placement of telecommunications antennae, minimum spanning tree computation and many, many more. A more complete survey (circa 1992) may be found in Okabe et al. [94].

The Hyperplane tessellation The next tessellation that we consider is the *hyperplane tessellation*. This tessellation is constructed in d-dimensional Euclidean space, $\mathbf{R}^d$ by partitioning the space with a collection of affine hyperplanes. The empty regions left between the hyperplanes are taken to be the *cells* of the tessellation. Practical applications of this tessellation include locality consistent hashing, dimension reduction and the analysis of physical phenomena.

1.1 Graph navigation and the Delaunay tessellation

Many of our results will relate to algorithms and properties of the Delaunay tessellation, in particular we focus on the problem of *point location*. A problem which is also highly related to *geographic routing* in wireless networks. We give a brief overview of both of these problems in this section

A Delaunay tessellation may be represented by a data structure, with a number of fundamental operations. The most basic of these include Insert, to insert a new point,

DELETE, to remove a point, and LOCATE, to return a reference to the simplex containing a given point of the underlying space. The last of these operations is known as *point location*, and its efficient implementation has received extensive treatment in the literature. Despite this fact, a number of the most commonly used algorithms to solve this problem are only bounded by pessimistic worst-case analyses. It is these algorithms that we are interested in investigating.

Assuming that the Delaunay tessellation data structure is stored as a collection of simplices with neighbour pointers, the simplest way to implement LOCATE is to iterate over every simplex in the tessellation until the first one is found which contains the query point. This amounts to computing the sign of the determinant of a $d \times d$ matrix for each $(d-1)$-face of each simplex in the tessellation. In $\mathbf{R}^2$, this implies that the algorithm has a worst-case complexity of $\Theta(n)$, if the input triangulation contains n vertices.[1]

More sophisticated methods for performing point location include optimal $\Theta(\log n)$ time algorithms which require that the Delaunay data structure be *augmented* with an auxiliary structure which can be used to speed up queries. Whilst these can lead to good asymptotic performance, they are generally quite complicated from an implementation perspective, and require non-trivial amounts of extra space (on the order of $\Omega(n)$.) It is for these reasons that algorithms which do not require any (or very little) modification of the Delaunay data structure may often be preferred. One such option is to use *walking-algorithms*, which are algorithms that work by traversing the tessellation using the incidences between faces (or vertices) and using local geometric information. Whilst it is easy to construct 'bad' tessellations such that any walking algorithm will need to visit all of the simplices ($\Theta(n)$ in $\mathbf{R}^2$) in practice, walking algorithms are often very efficient. This is because for many practical applications, the point sets that are treated tend to have some spatial uniformity. To translate this intuition into practical bounds, we can consider replacing the worst-case analysis bounds with *average-case* analysis. This essentially means averaging the complexity of an algorithm over the space of inputs. Average-case analysis tends to result in pathological cases which make the worst-case analysis overly pessimistic become negligible, with the aim of achieving a more realistic reflection of reality.

Whilst our primary interest for studying walking algorithms is in the context of point location in Delaunay triangulations, this is not the only domain where these algorithms are used. For example, another fruitful application of these techniques is in the area of *geometric routing* in networks. Many walking algorithms that can be used for point location also transfer well to graph routing, and *vice versa*. In Chapter 3 we provide a detailed comparison of both of these applications.

Despite the obvious interest in performing average-case analysis for walking algorithms, the dependence structure of Delaunay triangulations makes it very complicated to perform the probabilistic analysis in practice. The only previous analysis that has been completed in this area is that for *Straight Walk*, which is a walking algorithm that follows a straight line between an arbitrary chosen start vertex in the tessellation and the query point. In this case, Devroye et al. [50] proved that for a Delaunay triangulation of n uniformly random points in a square in $\mathbf{R}^2$, the expected number of triangles visited for a random query will be $O(\sqrt{n})$. The analysis in this case is simplified because it reduces to counting the

[1] We assume all complexities are given in the RAM model of computation throughout.

expected number of triangles intersecting a fixed line. In particular, the probability of a random triangle being visited by the walk depends only on the start and query points. This is not the case in most other walking strategies, including most notably those algorithms which tend to perform the best in practical implementations.

In this dissertation, we give the average case analysis for a new algorithm that traverses the Delaunay triangulation which we call *Cone Walk*. Our analysis provides detailed probabilistic bounds on the cost of the worst-case point location query for a fixed number of uniformly distributed points in a closed convex region. We are careful to ensure that our bounds work even when approaching the border of the triangulation without degradation. We additionally show that our algorithm has implications to the field of geometric routing in networks, by showing that our algorithm has constant *competitiveness* and satisfies some locality properties. To the best of our knowledge, this is the first average-case analysis that has been given for an algorithm that performs geometric navigation in an 'interesting' planar graph with bounds on the memory of the process. We additionally propose another new algorithm which we call *Pivot Walk*. We use simulations to show that Pivot Walk tends to require significantly fewer orientation tests on average than other walking algorithms when it is run on randomly generated data.

1.2 Extrema in geometric structures

The second part of this dissertation will be focused on the study of *geometric characteristics* of random tessellations. By geometric characteristics, we mean properties such as the areas and the circumradii of the cells, the degrees of the vertices, the lengths of the edges, and others. Clearly, the behaviour of these properties has an important influence on algorithms operating on tessellations, and so understanding the average-case behaviour of these properties is important to investigating the expected performance of algorithms.

For random Delaunay, Voronoi and hyperplane tessellations, many results have been obtained for the properties of the *typical* cell, which may be understood as the result of choosing a cell at random from the tessellation. The typical cell admits an exact description using the tools of stochastic geometry, and is accordingly well understood [25, 94, 104]. In order to obtain more detailed understanding of these tessellations, we can consider studying the *global* behaviour of cells of the tessellation. This may be achieved by the use of *extreme value theory*, which is concerned with the upper and lower extremal values of geometric characteristics for all cells in the process. In contrast to more simple asymptotic estimates for the expected minima and maxima, the use of extreme value theory permits a very precise understanding of the behaviour of geometric characterises through seeking to explicitly provide the limiting distributions.

Whilst extreme value theory already has a great deal of applications in the case of iid (independent and identically distributed) and stationary sequences, only a small amount of results have been given for random tessellations. This is mainly due to the difficulties in dealing with the complicated ways in which cells in random tessellations can interact, and is further compounded by difficulties in dealing with boundary effects. Recently, results have been obtained by Calka and Chenavier [26] and Chenavier [30], in which a number of properties of Delaunay and Voronoi tessellations are investigated, including

the introduction of a generalised theorem for obtaining the limiting distributions for the order statistics of properties of tessellations.

In our work, we consider properties for random hyperplane tessellations in $\mathbf{R}^2$. We are able to obtain the exact limiting distribution for the order-statistics of the largest and smallest inradii for a stationary hyperplane tessellation generated by a Poisson process in the plane. Despite the similarity in the results we obtain, our methods are rather different from those used by Calka and Chenavier [26] and Chenavier [30]. In particular, the general theorem provided by Chenavier [30] can not be applied in our context, since the required conditions are not satisfied.

1.3 Outline

We begin by introducing some of the required background in geometry and probability in Chapter 2. After this, the remainder of the work is split into two main parts. The first of these focuses on navigating tessellations, with applications to geometric routing and point location, beginning with a review of relevant results and continuing into more detailed analysis of walking algorithms. The majority of this part is concerned with the introduction and investigation of the algorithm Cone Walk.

The second part is concerned with extremes of random tessellations. An overview of the related literature is first provided, and the rest of this part is dedicated to proving a theorem relating to the largest and smallest inscribed radii of cells in the Poisson line tessellation.

Finally, the appendix contains information on software that was written during this work to aid the study of random tessellations.

Chapter 2

Geometric and Probabilistic Preliminaries

In the following sections we give a brief tour of the background required throughout the rest of this dissertation. For more details, the reader may consult the following works. For stochastic geometry and point processes, we refer to Schneider and Weil [104] and Baddeley [3]. Particular applications of stochastic geometry to wireless networks is specifically dealt with in books by Haenggi et al. [65] and Haenggi [64]. For Voronoi and Delaunay tessellations, Okabe et al. [94] is an invaluable resource. For probability applied to the analysis of algorithms we refer to Penrose [96], Szpankowski [108] and Cormen et al. [34]. A less formal, but succinct and refreshingly entertaining, introduction to Measure Theory may be found in Cheng [31]. For practical purposes such as technical implementation details, the CGAL manual is also an excellent resource [109].

2.1 Notational and stylistic conventions

We typeset spaces using bold face, so that $\mathbf{R}$, $\mathbf{Z}$, $\mathbf{N}$ represent the reals, integers and non-negative integers respectively. To represent the unit d-sphere, we write $\mathbf{S}_d$, with $\mathbf{S}$ as shorthand for $\mathbf{S}_2$. For a space $\mathbf{E}$, we write the associated Borel sets as $\mathcal{B}(\mathbf{E})$. We reserve d to denote the dimension of the ambient space throughout and we usually assume $d \geq 2$ to avoid trivialities. The Lebesgue measure over $\mathbf{R}^d$ will be written as λ_d, and the volume of the ball of radius 1 in $\mathbf{R}^d$ is denoted κ_d. The uniform measure on the d-sphere is written $\sigma_d(\cdot)$. We shall also write $B(z,r)$ to be the closed ball in $\mathbf{R}^d$ with radius r centred at z. For a set $\mathbf{X}$, we write $\mathbf{X}^k_{\neq}$ to denote the set of tuples in $\mathbf{X}^k$ in which no two elements are equal. To save space, we occasionally use $E_{1:k}$ to mean $E_1, E_2, \ldots E_k$. For a tessellation in the plane, we use the word *face* to refer to the empty regions bordered by the *edges* of the tessellation. In higher dimensions, we call the empty regions *cells*, which are separated by *faces*. Finally, all symbols are listed with the page on which they are first defined at the back of the manuscript for easy reference.

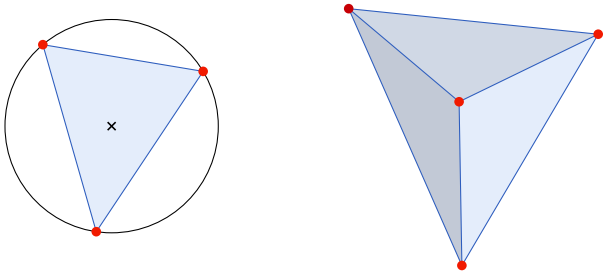


Figure 2.1 – A 2-simplex and a 3-simplex. The circumsphere (circumcircle) is given for the 2-simplex.

2.2 Results from geometry

We shall use a number of well-known results in geometry throughout this work. A brisk *whistle-stop* tour of the main concepts required is thus provided below for reference. Further details can be found in works such as Okabe et al. [94]. We begin by defining the *convex hull*, which in two and three dimensions may intuitively be thought of as the surface obtained by 'stretching' a rubber sheet over a set of fixed points.

Definition 2.1 (Convex hull)**.** The *convex hull* of a compact set $\mathbf{X} \subset \mathbf{R}^d$, written $\mathrm{Conv}(\mathbf{X})$ is defined to be the surface of the convex polytope formed by the set of points in $\mathbf{R}^d$ that can be written as strictly positive affine combinations of points in $\mathbf{X}$.

Given the above definition, we may give an appropriate generalisation of the concept of a *triangle* into higher dimensions, which we in turn use to define a triangulation dimension greater than two.

Definition 2.2 (Simplex)**.** A *d-simplex* in $\mathbf{R}^d$ is defined to be the convex hull of $d+1$ affinely independent points. Thus a triangle is a *2-simplex* and a tetrahedron is a *3-simplex*.

For any given d-simplex, we may associate a unique d-dimensional sphere which intersects all of the vertices of the simplex. We refer to this sphere as the *circumsphere* or the *circumcircle* in the plane. We also define the *circumball* and *cicumdisc* in the same spirit. In each case, we refer to the point at the centre of this sphere as the *circumcentre*. An example is given in Figure 2.1.

Definition 2.3 (Triangulation)**.** Given a set of locally finite points, $\mathbf{X}$ in $\mathbf{R}^d$, we define a *triangulation* of $\mathbf{X}$, to be a maximal collection of d-simplicies with vertices given by points in $\mathbf{X}$. We sometimes write $\mathrm{T}(\mathbf{X})$ to refer to a general triangulation.

Remark 2.4. It is not difficult to see that the surface of the union of all simplicies in the triangulation is exactly the convex hull.

A triangulation of particular importance to us is the *Delaunay triangulation*. The Delaunay triangulation satisfies a number of useful properties which make it very practical in both algorithm design and in probabilistic analysis.

Definition 2.5 (Delaunay Triangulation)**.** A *Delaunay triangulation* of a set of points $\mathbf{X}$, written $\mathrm{Del}(\mathbf{X})$ is a triangulation in which the interior of the circumball of every simplex does not contain any points in $\mathbf{X}$. We use $\mathrm{Del}(\mathbf{X})$ explicitly to represent the collection of $(d+1)$-tuples of vertices forming the simplices in the triangulation.

Remark 2.6. The definition given here may be fruitfully generalised by exchanging the empty circumsphere with other convex shapes. For example, using axis-aligned equilateral triangles gives a planar graph which is equivalent to the so called *half-θ_6* graph [12, 20]. Further generalisations are possible by replacing the space $\mathbf{R}^d$ with more exotic alternatives such as periodic [27] and hyperbolic spaces [7].

Another important tessellation which we consider is the *Voronoi diagram.*[1] This tessellation is obtained for a locally finite set $X \subset \mathbf{R}^d$ by associating with every point $x \in \mathbf{X}$ a *neighbourhood* of points whose nearest neighbour in $\mathbf{X}$ is x.

Definition 2.7 (Voronoi Diagram)**.** Given a locally finite set, $\mathbf{X} \subset \mathbf{R}^d$ with a distance metric $d(\cdot,\cdot)$ (which we assume is the Euclidean distance from now on), we associate with each point a Voronoi *cell*,

$$C_{\mathrm{X}}(x) \coloneqq \bigcap_{y \in \mathbf{X} \setminus x} R(x,y), \quad R(x,y) \coloneqq \left\{ z \in \mathbf{R}^d : d(x,z) \leq d(y,z) \right\}.$$

The Voronoi tessellation is then defined to be the set of all such cells,

$$\mathrm{Vor}(\mathbf{X}) \coloneqq \left\{ C_{\mathrm{X}}(x) : x \in \mathbf{X} \right\}.$$

An example of the Voronoi tessellation is given in Figure 2.2.

Duality A beautiful result in geometry is that the Voronoi tessellation and the Delaunay triangulation are *geometric duals*. This means that the Delaunay triangulation may be transformed into the Voronoi tessellation by associating a Voronoi vertex with the circumcentre of each simplex in the Delaunay triangulation. Adding Voronoi edges between all pairs of these vertices which were previously neighbours in the Delaunay tessellation gives exactly the Voronoi tessellation. A formal proof of this fact is given in Okabe et al. [94].

Degenerecy If we consider a finite set of points on a sphere containing at least $d+2$ points, $\mathbf{X} \subset \mathrm{S}_{d-1}$, $|\mathbf{X}| \geq d+2$, then it's easy to see that the Delaunay triangulation is not unique, and that every simplex shares the same circumball. To ensure that the Delaunay tessellation is unique, it is sufficient to assume that the points are in *general position*, defined below.

Definition 2.8 (General Position)**.** We say that a locally finite set of points, $\mathbf{X} \subset \mathbf{R}^d$ ($d \geq 2$) is in *general position* if no $d+1$ points lie on the same hyperplane and no $d+2$ lie on a hypersphere.

[1] The Voronoi tessellation is sometimes referred to *Thiessen* or *Dirichlet* tessellation, though these names seem to be falling out of fashion.

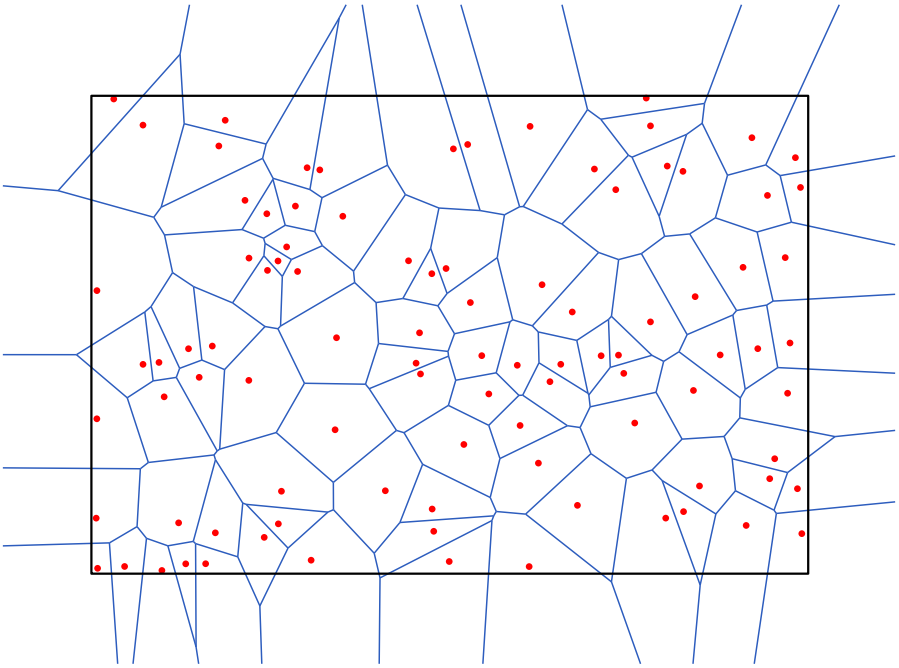


Figure 2.2 – A Voronoi diagram for a set of uniformly random points in a rectangle in $\mathbf{R}^2$.

Remark 2.9. When a set of points $\mathbf{X}$ is in general position, then the Delaunay triangulation is unique.

Remark 2.10. In most contexts throughout this work, all sets of points that are generated by random processes will be in general position almost surely, since these *degenerate* configurations are measure-zero events.

Properties of the Delaunay triangulation Given a set of points $\mathbf{X} \subset \mathbf{R}^d$ in general position, The Delaunay triangulation $\mathrm{Del}(\mathbf{X})$ satisfies a number of useful properties.

i) Let $(x, y) \in \mathbf{X}^2_{\neq}$. If the ball with diameter xy contains no points in $\mathbf{X}$, then the edge (x, y) is in the Delaunay triangulation.

ii) For every $x \in \mathbf{X}$, having a nearest neighbour $y \in \mathbf{X}$, (x, y) is an edge in the Delaunay triangulation.

iii) The number of simplices in a Delaunay triangulation in $\mathbf{R}^d$ of n points is $O(n^{\lceil n/2 \rceil})$ and $\Omega(n)$ [106]. In $\mathbf{R}^2$, we can count the number of triangles exactly using Remark 2.12.

Definition 2.11 (Euler's characteristic)**.** For a connected planar graph with n vertices, n_f faces and n_e edges, The *Euler characteristic* states that

$$n - n_e + n_f = 2.$$

Remark 2.12. In the case of planar triangulations, we observe that every face has three edges, each of which is counted twice unless it is on the convex hull. Suppose there are c edges on the convex hull, then $3n_f = 2n_e - c$, Which gives the well known result for planar triangulations,

$$n_f = 2n - 2 - c,$$
$$n_e = 3n - 3 - c.$$

Data structures for triangulations

When discussing algorithms implemented on triangulations, we need a conceptual model for operations that may be done on the basic *triangulation data structure*. To this end, we systematically assume that the Delaunay triangulation is stored using the following *triangle* representation. This is the representation used for triangulations in CGAL [109], and has also been used in higher dimensions [10]. Alternatives such as the *quad-edge* representation by Guibas and Stolfi [62] may also be considered without changing the complexities of any algorithms we consider. In our model, only the full-dimensional simplices and the vertices are stored. Each stored simplex has access to a list of its $d + 1$ neighbouring simplices, which may be accessed as $v.\text{neighbour}(i)$ for $i = 0, \ldots, d$. The $d + 1$ vertices of a d-simplex, σ are accessible by $\sigma.\text{vertex}(i)$ for $i = 0, \ldots, d$. We shall assume that every simplex is consistently oriented, which is for example guaranteed in CGAL [115]. The vertices are stored separately in a list, and it is assumed that each vertex has a pointer to one of the simplices adjacent to it. Finally, each memory access is assumed to take constant time.

2.3 Analysis of algorithms

Perhaps the most common way to compare algorithms is using *worst-case* analysis, which amounts to bounding the largest number of *steps* (in a given model of computation) that an algorithm can do given an input of a given size, n. This metric is very convenient for algorithms operating on lower-dimensional spaces, such as algorithms operating on sequences of numbers or characters of text. However, for higher dimensional inputs, computing complexity just in terms of the input size cannot necessarily give a realistic understanding for the performance of the algorithm. For example, we may consider the DELETE operation for a triangulation in $\mathbf{R}^2$, for which there exists an algorithm that requires a number of steps proportional to the degree of the vertex to be deleted [1]. The worst-case complexity for this algorithm given a triangulation of n points is thus $\Theta(n)$. This is because a triangulation can be constructed with one vertex of degree $n - 1$. In practice however, DELETE usually only needs to access a very local neighbourhood of vertices around the vertex to be removed. Formalising this intuition is the purpose of average and smoothed case analysis.

Average-case analysis Average-case analysis still measures the performance of an algorithm in terms of the input size, but instead of taking the largest number of steps for any input of a given size, we apply a probability distribution to the space of inputs of the

algorithm and then average over the number of computational steps required when the input is sampled from that distribution. As an example, Delete can be analysed using average-case analysis by assuming that the vertex to be deleted is chosen uniformly at random from the input triangulation. In this case, we have that the expected number of computational steps required is $\Theta(1)$ in $\mathbf{R}^2$ (see de Berg et al. [36, Theorem 9.1]), even without any assumptions on the triangulation. Since the number of steps required by the algorithm on a fixed input size becomes a random variable in some probability space in this model, we are able to employ the whole toolbox of probability theory to analyse its behaviour.

Smoothed analysis A weakness of average-case analysis is that it assumes that realistic use-cases for algorithms resemble random inputs. This is not always a reasonable assumption. *Smoothed analysis* was therefore proposed as way to 'interpolate between' worst-case and average-case analysis, thereby reducing the assumptions on the configuration of the input[107]. The idea is to consider the expected worst-case behaviour given any input of fixed size n in the input space with region of 'noise' added to each point. As the distribution for the noise becomes less concentrated, the results become closer to the results given by standard average-case analysis. This method of analysis similarly can be used to make the effects of pathological cases negligible. Famous results in this area have already been achieved for the Simplex Algorithm [107]. Whilst smoothed complexity clearly provides detailed information about the algorithmic complexity, it can be difficult to achieve bounds in practice.

Randomised algorithms We briefly outline the important distinction between average-case analysis of algorithms and *randomised* algorithms. We shall define a randomised algorithm to be an algorithm that takes a deterministic input, and has access to random *bits* which may be used during computation. Thus the worst-case complexity of such an algorithm for a fixed input size is also a random variable, however it does not necessarily coincide with the average-case analysis of the same algorithm.

2.4 Probability theory and point processes

In the chapters that follow, we make extensive use of techniques from probability theory and stochastic geometry. We give an overview of the most important tools in this section, with a particular focus on those results required for the analysis of algorithms. We begin by introducing some notation. [Define palm measure.]

Definition 2.13 (Probability space)**.** Given a space of outcomes, Ω, with associated σ-algebra $\mathcal{F}$ (which is a collection of sets closed under complement and countable union) and a sigma-additive measure $\mathbb{P}\colon \mathcal{F} \to [0,1]$ on the *measurable space* $(\Omega, \mathcal{F})$ satisfying $\mathbb{P}(\Omega) = 1$, we call the triple $(\Omega, \mathcal{F}, \mathbb{P})$ a *probability space.*

Point processes

There are multiple ways that point processes may be defined, and here we shall treat them as random measures. For a detailed and rigorous treatment of point processes we refer the reader to Schneider and Weil [104, Chapter 3]. Our treatment in what follows is given for an arbitrary locally compact space $\mathbf{E}$.

Definition 2.14 (Counting measure)**.** A *counting measure*, is a measure η such that $\eta(A) \in \mathbf{N} \cup \{\infty\}$ for every $A \in \mathcal{B}(\mathbf{E})$. We call a counting measure *simple* if $\eta(\{x\}) \in \{0, 1\}$ for every $\{x\} \in \mathcal{B}(\mathbf{E})$. We use $\mathbf{M}$ to represent the space of counting measures, and $\mathbf{M}_s$ to denote the space of simple counting measures on the space $\mathbf{E}$. We respectively associate the σ-algebras $\mathcal{M}$ and $\mathcal{M}_s$.

Definition 2.15 (Point process)**.** We shall define a *point process* to be a map from a probability space $(\Omega, \mathcal{F}, \mathbb{P})$, to the space of counting measures, $\mathbf{X}\colon (\Omega, \mathcal{F}, \mathbb{P}) \to (\mathbf{M}, \mathcal{M})$ such that the events

$$\left\{ \omega \in \Omega : \mathbf{X}(\omega)(A) = k \right\}$$

, for all $A \in \mathcal{B}(\mathbf{E})$, $k \in \mathbf{N}$ are all measurable. We shall call a point process *simple* if $\mathbf{X}(\omega) \in \mathbf{M}_s$ almost surely.

Definition 2.16 (Support)**.** For a counting measure, η we define its support as,

$$\operatorname{supp}(\eta) := \left\{ x \in \mathbf{E} : \eta(\{x\}) > 0 \right\}.$$

Remark 2.17. We adhere to the popular abuse of notation of associating a simple point process $\mathbf{X} := \mathbf{X}(\omega)$ with its support, $\operatorname{supp}(\mathbf{X})$. This means that we can use the notation $\mathbf{X}(\{x\}) = 1$ and $\{x\} \in \mathbf{X}$ interchangeably.

Binomial point process

One point process of great importance for us will be the *Binomial process*. For $n \in \mathbf{N}$, let $X_1, X_2, \ldots, X_n$ be independent and distributed in $\mathbf{E}$ with density f. We construct the process

$$\mathbf{X} := \sum_{i \leq n} \delta_{X_i}.$$

Where δ is the Dirac measure. It's easy to see that this point process satisfies our definition, since

$$\mathbb{P}(\mathbf{X}(A) = k) = \binom{n}{k} \mathbb{P}(X_1 \in A)^k \, \mathbb{P}(X_1 \in A^c)^{(n-k)}. \tag{2.1}$$

We note that the distribution in (2.1) is Binomial, to which this process owes its name. The Binomial process is convenient for performing average-case analysis for algorithms, since for a given input size n, we may consider the expected complexity given by running an

algorithm on the Binomial process $\mathbf{X}$ with n points in the input space of the algorithm. A downside of the Binomial process, however, is that the joint distributions of independent regions, $A, B \in \mathbf{E}$, $A \cap B = \varnothing$ are not independent. This is illustrated by considering

$$\mathbb{P}(\mathbf{X}(A) = k \mid \mathbf{X}(B) = n) = 0.$$

It is for this reason that other point processes may be preferred, namely the Poisson process.

Definition 2.18 (Poisson Process)**.** Let μ be a locally finite measure on the space $\mathbf{E}$ with no atoms. Then for $A \in \mathcal{B}(\mathbf{E})$, the point process satisfying

$$\mathbb{P}(\mathbf{X}(A) = k) = \frac{\mu(A)^k \exp(-\mu(A))}{k!}$$

exists and is simple. We call this process the *Poisson process* since the number of points falling within each region is Poisson distributed.

Definition 2.19 (Invariance)**.** When the intensity measure for a Poisson process, $\mathbf{X}$ is invariant under translations, we say that the process is *stationary*. If it is also invariant under rotations, it is *isotropic*.

The Poisson process has a number of properties that make it very convenient for performing analysis of algorithms. We use the following results extensively.

Theorem 2.20 (Properties of the Poisson process)**.** *Let $\mathbf{X}$ be a Poisson process on $\mathbf{E}$ with intensity μ.*

(a) Let $A \in \mathcal{B}(\mathbf{E})$, then

$$\mathbb{E}[\mathbf{X}(A)] = \mu(A).$$

(b) Let $A, B \in \mathcal{B}(\mathbf{E})$ with $A \cap B = \varnothing$, then

$$\mathbb{P}(\mathbf{X}(A) = k \mid \mathbf{X}(B) = t) = \mathbb{P}(\mathbf{X}(A) = k).$$

This is known as the independence property *of the Poisson process.*

(c) Suppose that $\mathbf{X}$ is stationary, then for $A \in \mathcal{B}(\mathbf{E})$,

$$\mathbf{X} \cap A\big|_{\mathbf{X}(A)=n} \overset{d}{=} \mathbf{X}'$$

Where $\mathbf{X}'$ is the Binomial point process on A with n points and uniform density, $\mathbf{X} \cap A$ is used to mean the restriction of $\mathbf{X}(\omega)$ to A and $\overset{d}{=}$ is used to denote equality in distribution.

(d) Let $\mathbb{P}_x(\cdot)$ denote the Palm measure*, which may be loosely interpreted as the distribution of $\mathbf{X}$ conditional on there being a point at x. In which case, for an event $A \in \mathcal{B}(\mathcal{F})$*

$$\mathbb{P}_x(A) = \mathbb{P}(X \cup \{x\} \in A). \tag{2.2}$$

Proof omitted. □

Remark 2.21. Theorem 2.20 Part (d) is known as *Slivnyak's theorem*, and roughly means that assuming a fixed point is at a given location in the process does not change its distribution. In fact, the reverse implication is also true, so this property uniquely characterises the Poisson processes. More details can be found in Schneider and Weil [104, Theorem 3.3.5].

Tools from stochastic geometry

A number of tools in stochastic geometry prove extremely useful in the analysis of random algorithms. We provide the following results for use later, their proofs may be found in Schneider and Weil [104]. The first of these results in the Slivnyak-Mecke formula, which allows us to compute the measure over configurations over the underlying space.

Theorem 2.22 (Slivnyak-Mecke)**.** *Let* $\mathbf{X}$ *be a Poisson point process on* $\mathbf{E}$ *with intensity measure* μ *(with no atoms). Then for fixed* $k \in \mathbf{N}_+$ *and non-negative* $f\colon (\mathbf{M}_s, \mathcal{M}_s) \times \mathbf{E}^k \to \mathbf{R}$*,*

$$\mathbb{E}\left[\sum_{(x_1,\ldots,x_k)\in \mathbf{X}^k_{\neq}} f(\mathbf{X}, x_1, \ldots x_k)\right] = \int_{\mathbf{E}} \cdots \int_{\mathbf{E}} \mathbb{E}\left[f\left(\mathbf{X} + \sum_{i\leq k} \delta_{x_i}, x_1, \ldots x_k\right)\right] \mu(dx_1)\cdots\mu(dx_k).$$

Proof omitted. □

Integral transforms The following two Blaschke-Petkantschin type integral transforms, when coupled with the Slivnyak-Mecke formula (Theorem 2.22) can greatly simplify calculations. We state them here for completeness, though their proofs may be found in Schneider and Weil [104, p.287]. The theorem allows us to transform the integral over $d+1$ points into the integral of the vertices of a simplex having a circumball of known centre and radius.

Theorem 2.23 (Blaschke-Petkantschin)**.** *Let* $f\colon (\mathbf{R}^d)^{d+1} \to \mathbf{R}$ *be a non-negative measurable function, then*

$$\int_{(\mathbf{R}^d)^{d+1}} f \, d\lambda_d^{d+1} = d! \int_{\mathbf{R}^d} \int_0^\infty \int_{\mathrm{S}_{d-1}} \cdots \int_{\mathrm{S}_{d-1}} f\Big(z + r(u_0,\ldots,u_d)\Big) \times r^{d^2-1} \lambda_d(\mathrm{Conv}(u_0,\ldots,u_d))\sigma(du_0)\cdots\sigma(du_d)dr\lambda(dz)$$

Proof omitted. □

The next theorem allows us to transform tuples of $d+1$ hyperplanes in the space of affine d-hyperplanes, $\mathcal{A}^d$ to a ball of known centre and radius with $d+1$ hyperplanes to its surface. We delay the formal definition of this space to Section 8.1, where this result is used for the study of the Poisson line tessellation. We also take, for $u \in \mathrm{S}_d$ and $t \in \mathbf{R}$, $H(u,t)$ to be the affine hyperplane through the point ut and normal to ut.

Theorem 2.24 (Blaschke-Petkantschin). *$f\colon (\mathcal{A}^d)^{d+1} \to \mathbf{R}$ be a non-negative measurable function and let μ_d be the uniform (Haar) measure on $\mathcal{A}_d$. Then*

$$\int_{(\mathcal{A}^d)^{d+1}} f \, d\mu_d^{d+1} = d! \int_{\mathbf{R}^d} \int_0^\infty \int_{S_{d-1}} \cdots \int_{S_{d-1}} f\Big(H(u_0, \langle z, u_0\rangle + r), \ldots, H(u_0, \langle z, u_d\rangle + r)\Big) \times \mathbb{1}_{P(u_0,\ldots u_d)} \, \lambda_d(\mathrm{Conv}(u_0, \ldots, u_d))\sigma(du_0)\cdots\sigma(du_d)dr\lambda(dz).$$

Where $P(u_0, \ldots, u_d)$ is the event that $u_0, \ldots, u_d$ do not lie in a closed hemisphere.

Proof omitted. □

2.5 Models for average-case analysis

When modelling random spatial tessellations for the average-case analysis of algorithms, there are a small number of possible processes we may consider. We consider some of the benefits and pitfalls of each approach below. We also discuss methods to deal with boundary effects.

Comparing point processes

Binomial process The process which is perhaps the easiest to interpret and understand when doing average-case analysis is the *Binomial process* with a uniform density function. This process neatly models our intuitive understanding of *n random points in a region.* In addition, since the number of points is fixed, we have a direct estimation of the expected number of steps an algorithm will require for a particular input size. Asymptotic bounds are achieved by considering the limit as the number of points in a fixed region tends to infinity. A potential 'catch' is that the number of points in independent regions of the process are *not* independent. This can mean that some ingenuity may be required to compute the probabilities of certain events.

Poisson process A suitably re-normalised Poisson process can also be used to model *n random points in a region*, however in this case we can only estimate the actual number of points that will be sampled. Whilst the expected case is exactly the same as the Binomial process (see Theorem 2.20, Part (b)), we cannot *a priori* guarantee a given number of points. This inconvenience may be contrasted against the availability of powerful tools such as the Slivnyak-Mecke formula (Theorem 2.22), which do not transfer to the Binomial case. A common convention is to refer to the Delaunay and Voronoi tessellations of a set of points generated by a Poisson process as the *Poisson Delaunay* and *Poisson Voronoi* tessellations.

(α, β)-process Another model which has received some attention is the (α, β) *process.* The first reference to this model in the literature which is known to the authors appears to be as a remark at the end of a paper by Dwyer [52], though the first actual application seems to be by Bose and Devroye [14]. This model is intended to generalise the idea of *uniformly random points* to include more interesting distributions. The idea is to replace the uniform density in the Binomial process by a density bounded between α and β. The

resulting bounds are then given in terms of these limits, and the analysis may more realistically model certain inputs as a result. It seems that in principle the same technique could be applied to Poisson processes without much difficulty.

Depoissonisation

Whilst we stated that results on Binomial point processes can be more desirable for average-case analysis, we also suggested that it can be more difficult to prove results on Binomial point processes when compared to Poisson processes. It is thus common practice to apply the following technique, which can allow one to achieve the best of both worlds. The idea is to analyse an algorithm using a Poisson process and to then extract bounds for the Binomial case using *depoissonisation*. We give a brief outline of this technique, though more details may be found in Szpankowski [108] and Penrose [96].

Let $g_n := \mathbb{E}[f(\mathbf{X}_n)]$ be the expectation for the function f applied to $\mathbf{X}_n$, a set of n Binomial random points. It may be difficult to directly compute g_n for a fixed n (for example due to dealing with dependence) but easier for g_N, where $N \sim \mathrm{Po}(n)$, so that $\mathbf{X}_N$ is a Poisson point process process with intensity n (due to Theorem 2.20, Part (b)). Since N is Poisson distributed, we can write g_N as an infinite sum over fixed n,

$$\widetilde{G}(z) := g_N = \sum_{n \geq 0} g_n \cdot \frac{z^n \cdot \exp(-z)}{n!} \tag{2.3}$$

Retrieving the asymptotics for the Binomial case then reduces to *extracting the coefficient* of z^n in Equation 2.3.[2] Formally,

$$g_n = [z^n]\Big(n! \exp(z)\, \widetilde{G}(z)\Big)$$

Where $[z^n](\cdot)$ is the *coefficient extraction* operator for formal power series. This process is sometimes known as *algebraic depoissonisation*. In many simple cases, extracting asymptotic bounds for this coefficient is as simple as applying the appropriate theorem. For example, Szpankowski [108][Theorem 10.3] may be used to achieve asymptotic bounds when $g_N = O(n^\beta)$, for $\beta \in \mathbf{R}_+$, in which case we also have $g_n = O(n^\beta)$.

Boundary effects

Dealing with boundary effects in geometric algorithms on tessellations can often require a significant amount of work. For example, non-negligible effects occur near the boundary of Delaunay triangulations in the plane, since edges tend to be asymptotically longer [78]. Analyses which correctly deal with boundary effects are thus desirable, to guarantee good behaviour of algorithms in all situations.

Various techniques to deal with boundary effects can be found in the literature, with the easiest being to explicitly assume that an algorithm is limited to parts of the tessellation sufficiently far from the boundary. Another similar option is to consider the average-case complexity of an algorithm run on an *infinite* tessellation observed in a window which is scaled to infinity. This is the model we shall use in Chapter 8.

It may also be possible to explicitly deal with the boundary effects, for example by proving that they do not asymptotically impact the algorithm run-time.

[2] Equation 2.3 is known as the *Poisson transform* of g.

2.6 General results

Poisson tail bounds

It is often convenient to give simple bounds to the number of elements that may be found in a region of a Poisson process, the following Lemma is reproduced from Penrose [96], with a small modification.

Lemma 2.25 (Poisson bounds)**.** *Given a Poisson random variable of rate λ_n and the positive increasing sequence, k_n such that $\lambda_n \in o(k_n)$ for $n \to \infty$, then there exists an n_0 such that for all $n > n_0$,*

$$\mathbb{P}\left(\mathrm{Po}(\lambda_n) \geq k_n\right) \leq \exp(-k_n).$$

Proof. We prove a more general version and then specialise to our case. Let $X \sim \mathrm{Po}(\lambda)$. Applying Markov's inequality, we have

$$\begin{aligned}
\mathbb{P}(X \geq k) = \mathbb{P}\left(z^X \geq z^k\right) &\leq \frac{\mathbb{E}\left[z^X\right]}{z^k} \qquad \text{let } z := k/\lambda, \\
&= \exp\left(k\left(1 - \log\frac{k}{\lambda}\right) - \lambda\right) \\
&\leq \exp\left(-\frac{k}{2}\log\left(\frac{k}{\lambda}\right)\right).
\end{aligned}$$

The last line follows by noting that $1 \leq 1/2\log(k/\lambda)$ when $k \geq e^2\lambda$. The stated result then follows since this is satisfied for n large enough. □

Concentration of measure

Many of our computations will follow by computing expectation and then using concentration arguments and dependency graphs to show that the result can be extended to all possible configurations. Since the number of configurations that we consider is often polynomial, it will not suffice to use standard arguments such as Markov's inequality or Chebyshev's inequality. In addition, we often do not have sequences of independent random variables, but rather sequences of random variables with *bounded* dependence. The following theorem is taken from Janson [70] and is an extension of previous Hoeffding-type inequalities applicable in the context of sums of independent random variables [68]. This subject is treated in more detail in [51].

Definition 2.26 (Dependency graph)**.** Given a sequence of random variables, $\{\xi\}_{i\geq 0}$, we define the *dependency graph* for those random variables to be a graph whose vertices represent random variables in the sequence, with an edge between two vertices if the random variables are dependent.

Theorem 2.27 (Janson)**.** *Let $\{\xi\}_{i\leq n}$ be a sequence of non negative real-valued random variables almost surely bounded by $a_i \leq \xi_i \leq b_i$. On this collection we construct a dependency*

graph, $\mathbf{G}$ *whose maximum degree is denoted* χ*. Then,*

$$\mathbb{P}\left(\sum_{i=1}^{n} \xi_i \geq \sum_{i=1}^{n} \mathbb{E}[\xi_i] + t\right) \leq \exp\left(-\frac{t^2}{\chi \cdot \sum_{i=1}^{n}(a_i - b_i)^2}\right)$$

Proof omitted. □

Remark 2.28. The statement given in Theorem 2.27 can easily be reversed to give lower bounds.

Part I

Graph navigation and the Delaunay Triangulation

Chapter 3

Graph Navigation in the Delaunay tessellation

The problem which we focus on these chapters was motivated by the study of point location in the Delaunay triangulation using walking algorithms. We recall that walking algorithms are simple procedures for performing point location which take as input a query point, and then navigate through a tessellation data structure using neighbour pointers until the face of the tessellation containing the query point is found. The problem of point location using walking algorithms also shares many similarities with *geometric online routing in networks*, which is the problem of finding a path for a data packet to a given destination in a connected network of nodes knowing their approximate locations. To avoid focusing too much on any given application, we shall frame our problem as a generalised problem on graphs, which we call *graph navigation*. Our use of terminology will reflect this, and we shall mix terms from graph routing and point location when it is practical or when it suits our taste. A number of algorithms and results have been proposed for both of these two domains, with a large amount overlap. We shall thus begin by introducing the two domains, and then focus on the algorithms and how they apply to each respective domain in the sections that follow.

We shall try to remain dimension-agnostic in our definitions where possible, however most of the results we give will be focused on the case of $\mathbf{R}^2$. For the case of graph routing, this is entirely natural, since the majority of geometric routing algorithms are applied to planar graphs. In the context of point location, the most important cases are $\mathbf{R}^2$ and $\mathbf{R}^3$, since the majority of spatial data is used to simulate real-world objects. In addition, for results that hold for random planar Delaunay triangulations, extensions to Delaunay triangulations in $\mathbf{R}^d$ are often possible by using similar arguments to those in the two-dimensional case.

Finally, we limit ourselves to studying only classes of graphs which can be treated as embeddings of tessellations with convex cells. This is true by definition for all planar graphs, but does restrict the graphs we consider in higher dimensions. We will consider some of the consequences of this restriction in Section 3.7.

3.1 Point location in triangulations

Point location in spatial tessellations is a fundamental problem in Computational Geometry, and has accordingly received extensive study [39, 93, 99]. In its most full generality, the problem of point location applies to any arbitrary tessellation, although it is often more practical to focus on the case of triangulations and their higher-dimensional analogues. This is because arbitrary tessellations can often be efficiently triangulated in order to apply techniques developed for triangulations. Triangulations are also particularly important, because many algorithms for their construction depend upon efficient point location routines [22, 61].

Point location in planar triangulations was solved in optimal $\Theta(\log n)$ time and $\Theta(n)$ space (where n represents the number of points) by Kirkpatrick's hierarchy in the 1980's [72], however the proposed structure was too complicated to be useful for practical implementations. A number of simpler methods which are optimal only in a randomised sense have been proposed since [2, 8, 9, 32, 33, 47, 63], culminating in Devillers' *Delaunay Hierarchy* [45], which is relatively simple, requires only around 3% space overhead, and is very efficient in practice. Finding worst-case optimal point location strategies in higher dimensions, including the important case of $\mathbf{R}^3$ remains an open problem [44].

Despite the existence of a number of advanced and efficient techniques for performing point location in triangulations, simple walking algorithms prevail as the method of choice in many implementations. This is because they are simple to implement, require no space overhead or maintenance of auxiliary structures, and are amply fast in the majority of practical situations, despite their worst-case complexity being linear in the number of cells in most cases [48].

Walking algorithms were first introduced by Lawson [82] and Green and Sibson [61] to speed up implementations to construct the Voronoi diagram. In these early papers, it was suggested that the algorithms should intuitively require $\Theta(n^{1/d})$ steps (for n points in d dimensions), for most practical cases. Proving this formally with reasonable assumptions remains an open problem for the large majority of walking algorithms that have been proposed, excluding the case of *Straight Walk*, which admits a relatively simple average-case analysis, detailed in Section 3.4.

In some cases, walking algorithms are combined with other data structures of varying complexity to give a better choice of 'initialisation point'. The Delaunay Hierarchy may be considered as one such example [45], albeit a complicated one. Simpler methods such as *Jump and Walk* [37, 49, 92], *Keep, Jump and Walk* [39] have been proposed which work by storing a small subset of the input, and then using the nearest neighbour to the destination in that set as the initialisation point for each query.

3.2 Online routing in networks

Given a network of inter-communicating autonomous nodes, the problem of finding a path for a data packet from one node to a destination node is known as *routing*. Routing may be solved by techniques such as *flooding* or by constructing routing tables [58, 87]. However, there exist contexts where the graph of nodes that communicate with each other, or *network topology*, is subject to frequent changes. In these cases, these 'global' routing

methods can become unnecessarily wasteful of network resources. One particular example where this can occur is that of *Mobile Ad-hoc NETworks* or 'MANETS', which are collections of wireless nodes that communicate with each other but which do not have any fixed infrastructure. In these cases, a popular solution is to use *geometric routing*,[1] which was first introduced by Ko and Vaidya [76] and Kranakis et al. [79]. In this context, the nodes are assumed to know their approximate location. This permits the construction of algorithms that perform routing using only limited local information to find a path, reducing the dependence of routing on a complex global state [58, 87, 91].

The network topology in geometric routing may be chosen according to the distribution of the nodes, or the properties desired of the routing algorithm. Due to their ability to admit simpler routing strategies, planar graphs were suggested as good candidates by Bose et al. [18] and Karp and Kung [71], in particular because they can be considered as tessellations with two-dimensional faces and edges. Specific classes of planar graphs include the Relative Neighbourhood Graph (RNG) and the Gabriel Graph (GG), which first appeared in the context of geometric routing by Karp and Kung [71] and Bose et al. [18]. Later work suggested the use of restrictions of the Delaunay tessellation [56], and the complete Delaunay triangulation [16, 78]. The Delaunay tessellation has been considered desirable due to it being a *spanner*, and because of the number of routing algorithms which may be proved to always succeed on it.

Most of the theoretical work in this area has been concentrated on existance proofs, and proofs of correctness for particular routing 'primitives' on certain network topologies, [16, 17, 19, 20, 80]. Complexity analysis for geometric routing algorithms is somewhat 'thin on the ground', with simulation studies historically being the method of choice to compare routing strategies in networks. Kuhn et al. [81] seek to improve this by giving more careful simulation studies, stating in their own work that *"[analysing] the complexity of mobile ad-hoc routing algorithms appears to be not only a demanding but an almost impossible mission"*. Kuhn et al. [81] also suggest that whilst simulations are popular for performing comparisons, they are not very well suited to analysing walking algorithms due to the large degrees of freedom of these processes. One result relating to the expected complexity of geometric routing algorithms is given by Chen et al. [28], who prove that for an convex subdivision in $\mathbf{R}^2$, no memoryless routing algorithm is more efficient than a walking algorithm which makes choices completely at random (a random walk), when no information about the topology of the underlying graph is known.

3.3 Graph navigation

Given an embedding of a graph $G := G(V, E)$ into a metric space $(\mathbf{E}, d)$, (which we assume to be $\mathbf{R}^d$ with the Euclidean distance from now on), a source node $z \in V$ and a destination point $q \in \mathbf{R}^d$, we consider the *graph navigation* problem of navigating G from z to the nearest neighbour of q in V by *exploring* the graph using only local information. In particular, we assume that a graph navigation algorithm initially only has access to the start vertex z along with its neighbours, neighbours(z) and the destination, q. Each time the algorithm moves to a new vertex $v \in V$, it learns the neighbours of v. The algorithm

[1] Some authors refer to this as 'position-based', 'location-based' or 'geographic' routing.

may maintain a memory of the steps it has visited previously. We also require that the algorithm be able to output a *path* in the graph between z and final point found in q. Finally, in the case that the embedding of G is planar, we shall call this problem *planar* graph navigation.

Given a well-defined metric for the termination of the navigation, we may use graph navigation to model the problem of point location in a tessellation. The graph to be navigated may be either the graph of vertex-adjacencies or face adjacencies, which is the geometric dual graph. To model the case of geometric online routing, we may simply restrict the domain of the destination to V instead of $\mathbf{E}$ and the problems are identical.

Properties of graph navigation algorithms

To compare graph navigation algorithms, we define the following terms. We shall denote the set of all neighbours within graph distance k of some vertex v as the *k-neighbourhood* of v, denoted $\text{neighbours}_k(v)$.

k-memoryless We call a graph navigation algorithm *k-memoryless* if the algorithm only accesses graph vertices within a k-neighbourhood of a vertex and the destination, q when choosing the next vertex to take. When an algorithm is 1-memoryless, we simply say it is *memoryless*.

deterministic An algorithm is *deterministic* if it always follows the same sequence of vertices when run with the same input.

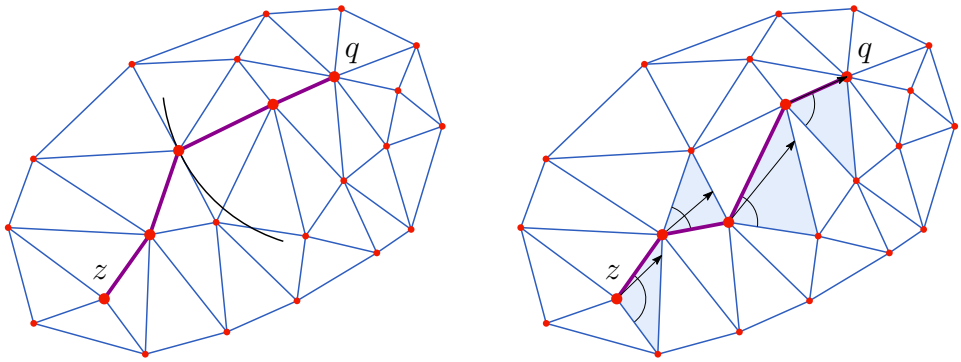
randomised An algorithm is called randomised it has access to a source of randomness during execution.

online We call an algorithm *online* if it requires $O(1)$ memory about the history of the process to complete.

c-competitive We call a graph navigation algorithm *competitive* if the *cost* of every possible path between z and q outputted by the algorithm is within a factor c of the Euclidean distance $|zq|$; for a given cost function assigning weights to the vertices and edges visited by the path. When not otherwised stated, we assume the cost is the sum of the edge lengths in the path.

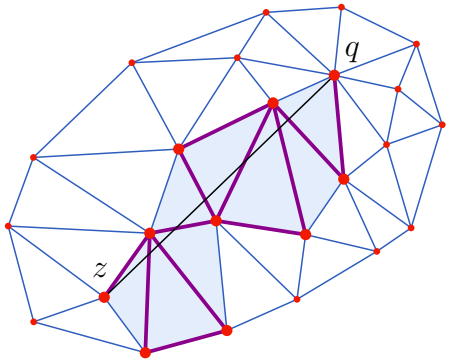
3.4 Graph navigation algorithms

In the following we detail the main 'primitives' which have historically been used to construct graph navigation algorithms. For now we simply treat the algorithms as sets of rules for traversing the process, and we deal with implementation details later in Section 3.5. We will adopt the convention of referring to the algorithms as 'walks', for example, 'Greedy Walk' however depending on the literature they may be better known by other names, such as 'Greedy Routing'. Some examples of the key algorithms we discuss applied to the Delaunay triangulation of a set of points in $\mathbf{R}^2$ are given in Figure 3.1. In the following, we assume that n represents the number of points, and d is dimension of the graph embedding

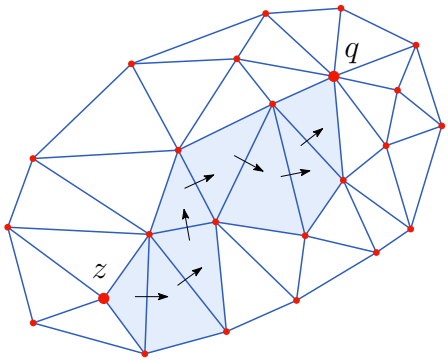


(a) Greedy Walk

(b) Compass Walk



(c) Straight Walk



(d) Visibility Walk

Figure 3.1 – Illustrations for different graph navigation algorithms applied to the Delaunay triangulation of a set of points in $\mathbf{R}^2$. We note that Greedy Walk and Compass Walk form a path in the edges, whereas visibility walk forms a path on the (triangular) faces. Straight Walk technically forms a path on the faces, though in geometric routing a path is always extracted on the edges. An example path is given which might be obtained by *right-hand routing*[79].

being considered. We also assume that n_f gives the number of full-dimensional faces in the tessellation, noting that $n_f = \Theta(n)$ in $\mathbf{R}^2$ due to Euler's formula (Definition (5.28).)

Greedy Walk

Greedy Walk is perhaps the simplest graph navigation algorithm. The algorithm is initialised by some vertex $z \in V$, and then iteratively chooses the vertex which is closest to the destination among the neighbours of the current vertex at each step (see Figure 3.1a.) It is easy to see that Greedy Walk is memoryless and deterministic. Greedy Walk always succeeds in the Delaunay triangulation Bose and Morin [16], a result that extends without modification to $\mathbf{R}^d$. For non-Delaunay triangulations and other general more graph embeddings, Greedy Walk may get stuck in local minima [16]. It is known that Greedy Walk is not competitive for any class of graph which we have considered [16]. We also remark briefly that whilst Greedy Walk may get 'stuck' in local minima in many classes of graph,

work has been done to endow graphs with 'virtual coordinates' so that Greedy Walk can be made to succeed in more general network topologies [5].

Compass Walk

Given a current vertex u and a destination vertex q, *Compass Walk* is a deterministic memoryless graph navigation algorithm that works by choosing the vertex v neighbouring u such that the angle $\angle vuz$ is minimised (see Figure 3.1b.) Compass Walk always succeeds in the Delaunay triangulation, though it can fail in more general triangulations. This is demonstrated by counter-example by Kranakis et al. [79]. The algorithm first appeared in a paper by Kranakis et al. [79]. It has also been shown that no version of Compass Walk is competitive in any of the classes of graph we have considered [16].

Random Compass

If Compass Walk is randomised to choose uniformly between the two neighbours of u closest to the line uq at each step, then the algorithm succeeds almost surely on any convex planar tessellation [19].

Greedy-Compass Walk

Modifying Compass Walk by combining it with Greedy Walk results in a deterministic memoryless deterministic randomised algorithm that succeeds on any planar triangulation [19]. In this case, the algorithm considers the two neighbours of the current vertex u closest to the line uq, and then chooses the vertex minimising the distance to q.

Straight Walk

Straight Walk is a deterministic online graph navigation algorithm that succeeds in any tessellation in $\mathbf{R}^d$ (see Figure 3.1c.) The algorithm works by visiting all cells of the tessellation intersecting the line zq, for z the initial point and q the destination. It was first introduced by by [61] for point location, though it has also been used extensively in the context of geometric online routing, beginning with the paper by Kranakis et al. [79]. In the context of geometric routing, it is often modified to give different properties depending on the network topology, though the idea remains essentially the same. In the geometric routing literature, this strategy is often referred to as *face routing* or *right-hand routing*. It is easy to see that Straight Walk can only visit each face once, giving a $\Theta(f_n)$ worst-case bound for the number of faces visited. Straight Walk has also a detailed average-case analysis for the case of DelDny triangulations, which we deal with in Section 3.8. Finally, some paths generated using algorithms similar to Straight Walk for geometric routing may be competitive for planar Delaunay triangulations [16]; though standard variants such as *Face Routing* are not [16].

Visibility Walk

Visibility walk is perhaps the best known walking algorithm that has historically been used for point location. It was first introduced by Lawson [82], and quickly adopted by

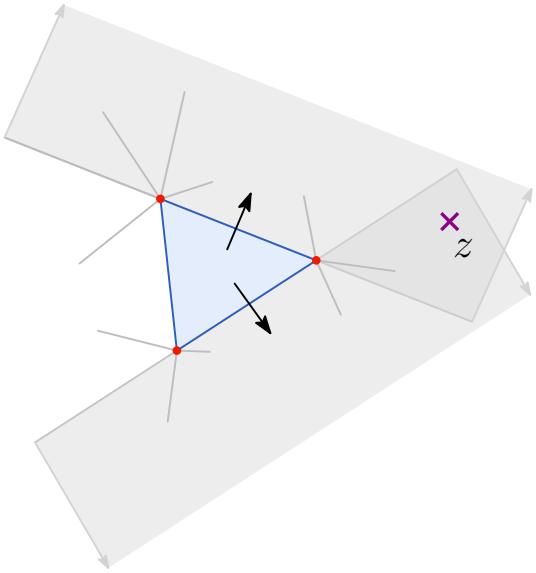


Figure 3.2 – *Orientation* or *visibility* based walks use orientation predicates to find edges which can be walked through. This reduces to checking to see which side of an edge the desination point, z falls on.

others[22, 61] to speed-up the construction of Delaunay triangulations. The basic idea of the algorithm is to navigate a triangulation by moving between *faces* until the destination is reached, so that the walk may be seen as a graph navigation in the dual graph of the triangulation. To select the next face at each step, Visibility Walk chooses the first edge of the current triangle such that the point lines on the halfspace supported by that edge not containing any other point of the current triangle (see Figure 3.2.) Visibility walk is memoryless and deterministic, and succeeds in all Delaunay triangulations, since no triangle is visited more than once [40, 53]. In more general triangulations the algorithm may visit triangles multiple times [48]. The algorithm extends easily to triangulations in higher dimensions. The worst-case analysis for Visibility Walk in Delaunay triangulations is $\Theta(n_f)$, though it has long been conjectured that the average-case complexity for visibility walk for n uniformly random points in a unit d-ball is $\Theta(|zq|n^{1/d})$ in expectation [38, 39, 48, 49]. Zhu [116] gave a tentative proof for this with a polylogarhimic factor, unfortunately the proof given assumes that each successive step in the walk process may be treated independently from the history of the process, which may be shown to be false by counter-example. Thus the proof must be considered false until this problem can be repaired. Ultimately, it is this dependence property that makes the average-case analysis of visibility-type walking algorithms particularly challenging.

Stochastic Visibility Walk

Similarly to Compass Walk, Visibility Walk can be made to succeed *almost surely* (with probability 1) on any triangulation by choosing amongst the possible 'ways on' at each step at random, a fact which is demonstrated by Devillers et al. [48]. The authors also show that

the worst-case complexity for stochastic visibility walk on arbitrary planar triangulations has expectation of at *least* $\Omega(2^{\sqrt[3]{n}})$. For Delaunay triangulations, the same bounds and conjectures follow as in the case of standard deterministic Visibility Walk given above.

3.5 Orientation-based walks

In the context of point location, most walking algorithms are implemented using *orientation predicates*. These predicates take as input the face of a simplex in a triangulation and a point, and then return Visible, Not-Visible or Colinear, depending on the configuration of the points (see Figure 3.2.) In practice, this is implemented by the computation of the sign of the determinant for a $d \times d$ matrix, for the case in $\mathbf{R}^d$ [48].[2] It is possible to construct a directed graph so that every orientation-based walk gives a path in that graph. This was first introduced by Devillers et al. [48], as the *visibility graph*. We shall refer to the same concept as the *walk graph* in order to avoid name clashes with other similarly named concepts.

Definition 3.1 (Walk Graph)**.** Let $\mathbf{X} \subset \mathbf{R}^d$ be a set of points in general position with a triangulation $\mathrm{T}(\mathbf{X})$, and choose a fixed *destination* point $q \in \mathrm{Conv}(\mathbf{X})$. We then use $\mathbf{W}(\mathbf{X}, q)$ to denote the directed graph with *nodes* given by the simplices of $\mathrm{T}(\mathbf{X})$ and directed *arcs*,

$$\sigma \longrightarrow \sigma.\,\mathrm{neighbour}(i) \iff \textsc{Orientation}(\sigma.\,\mathrm{face}(i), q)) = \textsc{Visible}.$$

It will be convenient to ensure that every node of the walk graph (contained in the convex hull of $\mathbf{X}$) has the same degree. To do this, we augment $\mathbf{W}(\mathbf{X}, q)$ with an 'infinite node' σ_∞ which has a directed arc connecting σ_∞ to every simplex touching the convex hull of $\mathrm{Del}(\mathbf{X})$.

Remark 3.2. Definition 3.1 is more restrictive than strictly necessary, though this definition will allow us to spend less time on small technicalities which do not significantly impact our results.

Since it is known that every path that follows strictly visible edges will always terminate with the triangle containing the destination point [40, 53], the walk graph must be acyclic. A simple way to prove this in $\mathbf{R}^2$ is to use the *circle power* and some simple geometry. Whilst this trick is colloquially known, it does not seem to appear in the literature. We thus reproduce a possible proof along these lines below.

Definition 3.3 (Circle power)**.** Given a circle centred at z with radius r and a point q; the *circle power* of q relative to the circle is given by $|zq|^2 - r^2$. Simple geometry may then be used to show that the circle power can also be computed by tracing an arbitrary line from q, to intersect the circle at two points p, p' (which may be the identical). The circle power is then given by $|zq| \cdot |qp'|$ (with negative sign if the point q is contained within the circle) a simple geometric argument can be used to show that this is independent of the choice of the line. An example is given in Figure 3.4.

[2] Libraries such as Cgal provide functions to compute these exactly, ensuring robustness.

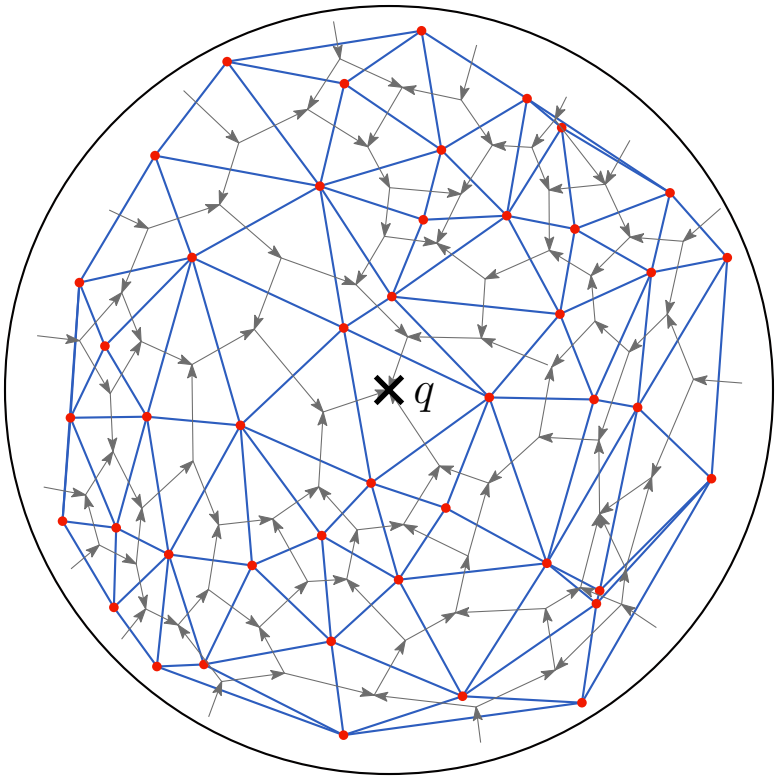


Figure 3.3 – An illustration of the *Walk Graph* for the Delaunay triangulation of a set of points in $\mathbf{R}^2$. The arrows represent the orientations through each of the edges of the triangulation and together form a graph which is *combinatorially* equivalent to the Voronoi tessellation. The points in each triangle are not chosen to be the Voronoi centres since it makes the illustration more clear.

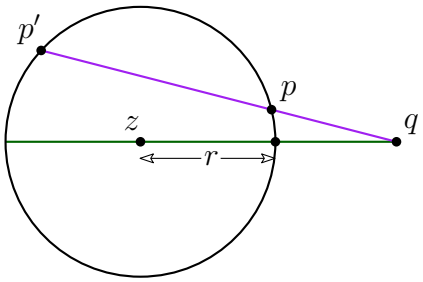


Figure 3.4 – The circle power of a point q relative to the circle centred at z of radius r is $|qz|^2 - r^2 \equiv |qp| \cdot |qp'|$ (where the right side is given negative sign if q is contained within the circle).

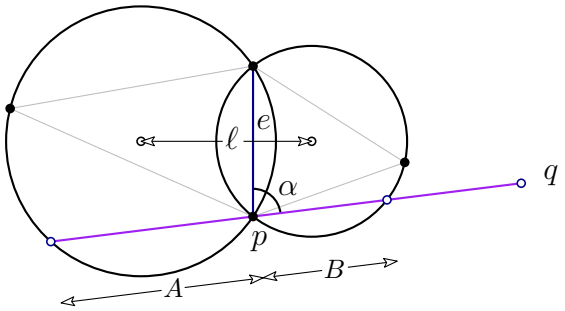


Figure 3.5 – The change in circle power relative to q when moving between two overlapping circles is $2\ell \sin\alpha|pq|$, and is thus independent of the radii.

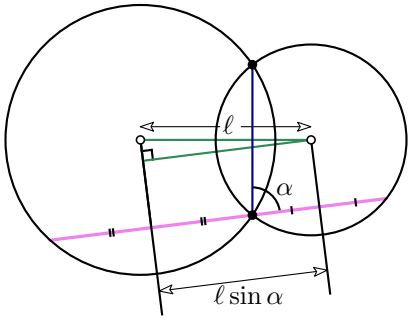


Figure 3.6 – The length of the line intersecting the interiors of both circles is $2\ell \sin\alpha$.

Lemma 3.4. *Consider two overlapping circles arranged as in Figure 3.5, and let e be the line segment between their two points of intersection. Let p be one of the end points of e and let α be the angle between e and the line pq, for q an arbitrary point. Then the change in circle power between the two circles relative to q (when moving* towards *q) is $2\ell \sin\alpha|pq|$.*

Proof. Let A be the length of the line segment which overlaps the interior of the first circle, and B be the length of the line segment overlapping the interior of the second circle (see Figure 3.5). We assume that the segments represented by A and B are non-overlapping, if this is not the case the computations must be modified accordingly. Pleasingly, it turns out that the stated result holds verbatim, independently of the arrangement of the circles and the choice of q; including when q is on the interior of one of the circles. We shall omit the proofs for these cases since they use the same ideas. Observe that the change in circle power when moving across the edge is now given by

$$|pq|(|pq| + A) - |pq|(|pq| - B) = |pq|(A + B). \tag{3.1}$$

To calculate $A+B$, we refer to Figure 3.6. Basic trigonometry gives us that $A+B = 2\ell \sin\alpha$ (for the case when A and B are overlapping, this becomes $A - B = 2\ell \sin\alpha$ by similar arguments). □

Theorem 3.5. *Suppose that* $\mathrm{T}(\mathbf{X})$ *is the Delaunay triangulation, then:*

1. *The graph* $\mathbf{W}(\mathrm{T}(\mathbf{X}), \cdot)$ *is acyclic.*

2. *Any navigation in the graph* $\mathbf{W}(T(\mathbf{X}), q)$ *must eventually terminate with the face containing the destination point* q*.*

Proof. Note that we may only walk across edges of the triangulation satisfying the visibility property. If the triangulation is Delaunay, then moving across an edge means the circumcentre of the new face is on the same side of the edge as the destination. Referring to Lemma 3.4, it is then not hard to see that this implies that $\alpha \in (0, \pi)$, so the circle power when traversing an edge is always strictly decreasing and $\mathbf{W}$ has no loops. To see that the walk never gets stuck before reaching q, observe that by Lemma 6.2, only one node in $\mathbf{W}$ has out-degree 0, and this is the face containing q. □

Walking the walk graph

Given the definition of the walk graph, Stochastic Visibility Walk may now be defined simply as a random walk on the walk graph which chooses between with multiple outgoing arcs by selecting an arc uniformly at random. Straight Walk may also be implemented in terms of orientation predicates. Devillers et al. [48] explicitly give an algorithm for implemented Straight Walk which requires exactly two orientations per triangle intersecting the line segment between start and end points.

Often, orientation-based algorithms can be very simple to implement. Notably, the pseudo-code listed below only requires a small number of modifications before it could be used for point location in a computational library such as CGAL. We discuss these details later in Section 3.6.

```
DETERMINISTIC-VISIBILITY-WALK(σ, q)
1  while true
2      for i = 0 to d
3          if orientation(σ.face[i], q) = VISIBLE
4              σ = σ.neighbour[i]
5              break
6      return σ
```

An implementation of Visibility Walk such as this has been observed to exhibit very good performance in practice. In simulation studies, it has repeatedly been observed to be faster than other walking strategies for point location in triangulations such as Straight Walk [38, 39, 48]. The performance of these orientation-based walks may even be improved even further using tricks such as 'remembering' the previous face visited in order to potentially avoid performing an orientation that has already been computed (we call this *Remembering Visibility Walk*) and 'termination guessing', which reduces the number of orientations by assuming the last orientation to be performed in a simplex always succeeds for a certain number of steps (this works since the last orientation only 'fails' once the destination has been reached.) Examples of these techniques given in [77] and [38] report up to 40% speed improvements for planar Delaunay point location using these techniques. Straight Walk, meanwhile, is tricky to implement correctly due to difficulties dealing with degeneracies [48]. It is for these reasons that a disparity has arisen between the

algorithms used in theoretical papers involving walking algorithms and algorithms used in simulation studies [38, 39]. Since Straight Walk has strong average-case guarantees, it is very often used to gain bounds on algorithms using walking algorithms. For simulations though, authors tend to use Visibility Walk since the paths taken tend to be quite similar, but the timings tend to be better. Resolving this question would thus have practical implications to research in this area. Unfortunately though, the dependence properties induced by the graphs makes probabilistic analysis difficult. Correctly dealing with dependence between steps requires some care. We make the following conjecture.

Conjecture 3.6. *Define* $\mathbf{P}(\sigma, q)$ *to be the set of all paths in the walk graph* $\mathbf{W}(\mathrm{Del}(\mathbf{X}), q)$ *leaving a simplex* $\sigma \in \mathrm{Del}(\mathbf{X})$. *Then for a collection* $\mathbf{X} \subset \mathbf{R}^d$ *of n points generated by a uniform Binomial process in a compact convex region D, and a constant* $C \in \mathbf{R}_+$ *(which may depend on d),*

$$\mathbb{P}\left(\sup_{\sigma \in \mathrm{Del}(\mathbf{X}),\, q \in D} \sup \left\{ \frac{|w|}{|z(\sigma) - q| \cdot n^{1/d} + \log^d n} : w \in \mathbf{P}(\sigma, q) \right\} \geq C \right) \underset{n \to \infty}{\longrightarrow} 0$$

In particular, we conjecture that in the limit as n tends to infinity, there is no path in the walk graph that visits a number of simplices which is more than a constant times the distance between the centre of the first simplex and the destination (as long as the walk is longer than some minimum length). Since this conjecture involves all possible orientation-based walks, an easy corollary of this result would be to give similar average-case bounds for all variants of Compass Walk, Straight Walk, all variants of Visibility Walk and Pivot Walk (which we introduce in Chapter 6) when applied to the Delaunay triangulation. This follows since a unique path in the walk graph can be associated with each of these algorithms (for compass walking strategies, this is a consequence of the same techniques used in Chapter 6.) We are as yet unable to prove this specific result, however in Chapter 5, we prove the same kind of bound for a walking strategy which shares many similarities with Greedy Walk.

3.6 Robustness

When implementing walking strategies, as with many geometric algorithms, it is important to consider the possibility that algorithms will fail due to rounding errors because of inexact arithmetic. This problem is most apparent when the locations of the vertices are deterministically fixed, such as in point location. It is known that robustness problems can lead to even provably correct algorithms failing due to rounding errors [114]. Various techniques exist to combat this problem, including lazy evaluation and filtered arithmetic kernels [46, 115]. Clearly these techniques can result in non-negligible overheads, and so a good way to improve software performance is to reduce the number of computations requiring exact arithmetic. This is an option we consider in Chapter 6.

3.7 Other graph navigation problems

In limiting ourselves to navigations of graphs which may be associated with a tessellation, we have implicitly ruled out a number of related results. In particular, for the related prob-

lem of navigation in the plane, several probabilistic results exist; for example Bonichon and Marckert [11] and Bordenave [13]. In this context, the input is a set of vertices in the plane along with an oracle that can compute the next step given the current step and the destination in $O(1)$ time. Although the steps are also dependent in these cases, the case of Delaunay triangulations we treat here is more delicate because of the geometry of the region of dependence implied by the Delaunay property. Other related problems which we do not consider include the famous results by Kleinberg [73], [74] involving decentralised routing and the *Small world phenomenon*. These results are usually based on the *Watts-Strogatz* model, which is a grid of nodes which are aware of their positions augmented with a number of random 'shortcut' paths. Notably, the shortcut edges make the graph non-planar, so this graph doesn't fall into our study of graphs which may be embedded as tessellations. In addition, The very regular nature of the graph removes the complicated dependence properties that may be found in the case of random points connected by geometric graphs which we consider. More general results in a similar vein have been given in the context of a Poisson point process based model by Franceschetti and Meester [55]. Finally, we remark that research has also been done in order to use *virtual coordinates* to enable geometric routing in more broad classes of network topology, by choosing coordinates for nodes specifically to allow routing using certain algorithms [5, 75, 91, 95, 100].

3.8 Analysis of Straight Walk

Straight Walk is unusual amongst other navigation algorithms in that has been formally analysed in the average-case setting for the case of random Delaunay triangulations. The average-case analysis of Straight Walk is somewhat simplified because the number of steps required reduces to counting the number of cells intersecting a fixed line. The first average-case result related to this algorithm was proved by Devroye et al. [50], in which they demonstrate that Straight Walk requires $O(\sqrt{n})$ steps in expectation when applied to a random pair of query and initial point amongst n uniform points in a square. This work extended a previous result by Bose and Devroye [14]. In this paper the authors also provide analysis for different ways to choose the first face for point location strategies. The analysis was later extended by Mücke et al. [92] to give a bound in $\mathbf{R}^3$, although with a polylogarithmic factor. Bose and Devroye [15] later wrote a paper bounding the *stabbing number* of the Delaunay triangulation, bounding the maximum number of edges in a random planar Delaunay triangulation intersecting a line crossing a bounded domain, which they call the *Stabbing Number*. The results given in this paper can be used to obtain a bound on the maximum number of steps needed by any run of Straight Walk of $O(\sqrt{n})$, for a particular realisation of a random Delaunay triangulation in the plane. More recent work by Rossignol and Pimentel [103] then allows one to obtain the same maximum for higher dimensions, by giving a bound on the stabbing number in $\mathbf{R}^d$, $d \geq 2$. We remark that the original paper addressing the complexity of Straight Walk given by Devroye et al. was achieved directly in the Binomial setting, using a bound on the Vapnik–Chervonenkis (VC) dimension to deal with difficulties arising due to dependence. Later results achieved by Rossignol et al. were obtained by applying the tools of percolation theory, in particular generalised the study of *Greedy Lattice Animals* first studied by Cox et al. [35].

In the following, we provide a simple alternative proof to bound the expected number of simplices visited by Straight Walk between two fixed points, away from the boundary. With a small argument to deal with the boundary and by applying the concentration inequality by Janson introduced in Theorem 2.27, it is possible to extend this bound to cover all possible walks. We use this technique in Chapter 5 for the bounds on Cone Walk.

Theorem 3.7. *Let* $\mathbf{X}$ *be a Binomial process with intensity 1 in* $\mathbf{R}^d$*. Then the number of simplices,* N *visited by the Straight Walk algorithm to travel between the fixed points* 0 *and* $(\ell, 0, \ldots, 0)$ *is*

$$\mathbb{E}[N] \leq \ell \cdot \kappa_d \cdot d! \cdot n^{1/d}. \tag{3.2}$$

Proof. Consider the space $\mathbf{R}^d$ with a Poisson point process, $\mathbf{X}$ having intensity measure λ_d, the d-dimensional Lebesgue measure. We now upper bound the number of $(d+1)$-simplices that intersect a fixed line segment, L with end points at the origin and $(\ell, 0, \cdots, 0)$. We observe that if a simplex intersects L, then its circumball must also intersect L. This gives us an upper bound as follows. Let $B(x_0, x_1, \ldots, x_d)$ denote the circumball of the $d+1$ points $x_0, x_1, \ldots, x_d$, and let N be the number of simplices intersecting the line. Applying the Slivnyak-Mecke formula (Theorem 2.22) and a Blaschke-Petkantschin type transformation (Theorem 2.24), we obtain

$$\begin{aligned}
\mathbb{E}[N] &\leq \frac{1}{(d+1)!}\,\mathbb{E}\left[\sum_{x_{0:d} \in \mathbf{X}^{d+1}_{\neq}} \mathbb{1}_{(x_0,\ldots,x_d)\in \mathrm{Del}(\mathbf{X})}\, \mathbb{1}_{B(x_0,\ldots,x_d)\cap L \neq \varnothing}\right] \\
&= \frac{1}{(d+1)!} \int_{(\mathbf{R}^d)^{d+1}} \mathbb{P}\left((x_0,\ldots,x_d) \in \mathrm{Del}(\mathbf{X} \cup \{x_0,\ldots,x_d\})\right) \mathbb{1}_{B(x_0,\ldots,x_d)\cap L\neq\varnothing}\, dx_{0:d}
\end{aligned}$$

applying the transformation and integrating out constants gives

$$\leq \frac{\kappa_d^{d+2}}{(d+1)} \int_{\mathbf{R}^d} \int_0^\infty \exp(-\kappa_d \cdot r^d) r^{d^2-1} \mathbb{1}_{|zL|\leq r}\; dr dz$$

by Fubini's theorem

$$\begin{aligned}
&\leq \ell \frac{\kappa_d^{d+3}}{(d+1)} \int_0^\infty r^{d^2+d-1} \exp(-\kappa_d \cdot r^d) dr \\
&\leq \ell \cdot \frac{\kappa_d}{(d+1)} \Gamma(d+1).
\end{aligned}$$

Assuming $d \geq 2$. □

Remark 3.8. The same result in the Binomial setting then follows directly from the depoissonisation result in Section 2.5.

Remark 3.9. The constants we give in Theorem 3.7 may be tightened up significantly by taking a little more care over the upper bounds used in the proof. These constants may be computed explicitly using the results in Schneider and Weil [104][Chapter 3].

Chapter 4

Maximum degree for Poisson Delaunay in a smooth convex

In this chapter we give an approximate bound on the maximum degree of a Delaunay triangulation in the plane. This bound is required later in Chapter 5 to bound the complexity of Cone Walk when approaching the boundary.[1] Let $\mathbf{X}$ be a homogeneous Poisson process on the entire Euclidean plane with intensity 1. In this case, Bern et al. [6] give a proof that the expected maximum degree of any vertex of the Delaunay triangulation $\mathrm{Del}(\mathbf{X})$ falling within the window $[0, \sqrt{n}\,]^2$ is $\Theta(\log n / \log\log n)$. Whilst such a bound is very useful in the analysis of geometric algorithms, it has a shortcoming in that it implicitly avoids dealing with the boundary effects that occur when considering points distributed within bounded regions. When considering such bounded regions, it can be observed that the degree distribution is significantly skewed near the boundary, with the majority of the vertices on or near the convex hull having a much higher degree than the global average. It is therefore not altogether trivial that the maximum degree should still be bounded polylogarithmicly when the considering a triangulation of random points in a bounded convex. In this chapter, we show that this indeed the case for the specific case of a smooth compact convex, $D \subset \mathbf{R}^2$. The strong bounds we derive here will be important later when analysing the complexity of the Cone Walk algorithm in Chapter 5. In the following we assume without loss of generality that $\lambda_2(D) = 1$. Let $D_n := \sqrt{n}\,D$; then D_n has area n and diameter at most $c_2 \cdot \sqrt{n}$, for a constant $c_2 \in \mathbf{R}_+$. Let $\mathbf{X}$ now be a planar Poisson point process of intensity 1 restricted to D_n, so that in expectation we have $\mathbb{E}[\mathbf{X}(D)] = n$. Let $\delta_{\mathbf{X}}(x)$ denote the degree of $x \in \mathbf{X}$ in $\mathrm{Del}(\mathbf{X})$.

Theorem 4.1. *Let D be a smooth compact convex in $\mathbf{R}^2$ of unit area and let $\mathbf{X}$ be a Poisson point process of intensity 1 in $\sqrt{n}\,D$. Then for any $\varepsilon > 0$, we have for n sufficiently large,*

$$\mathbb{P}\left(\max_{x \in \mathbf{X}} \delta_{\mathbf{X}}(x) \geq \log^{2+\varepsilon} n \right) \leq \exp\left(-\log^{1+\varepsilon/4} n \right). \tag{4.1}$$

[1] This chapter forms part of a paper written in collaboration with Nicolas Broutin and Olivier Devillers, and is currently out for review [24]

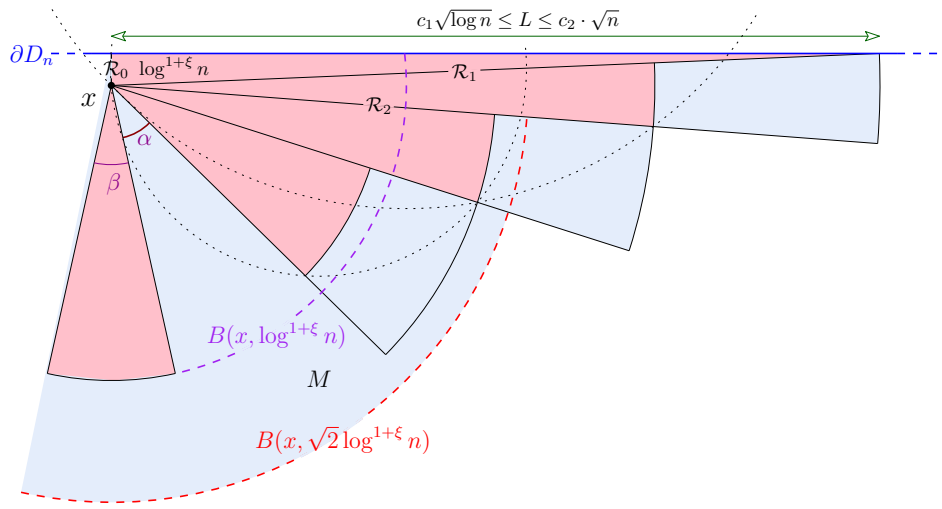


Figure 4.1 – The construction to bound the maximum degree of a vertex near the boundary. The pink shaded circular sectors are each conditioned to contain at least one point. In this case, no Delaunay neighbour to the right of x can lie outside of the outer shaded region.

Define $\|Ax\| := \inf\{\|xy\| : y \in A\}$ and let ∂D denote the boundary of D. Note that since our proof is concerned with asymptotic limits as n tends to infinity, the smoothness of D ensures that the boundary ∂D appears locally 'flat' relative to any collection of $o(\sqrt{n})$ points. Our proof will follow by considering two cases. For the first case, we consider all points $x \in \mathbf{X}$ satisfying $\|\partial D_n x\| \le \sqrt{\log n}$ and bound the number of neighbours of x in $\mathrm{Del}(\mathbf{X})$ to one *side* of x; doubling the result and the end for the final bound (See Figure 4.1). To begin, we trace a ray from x to the point $y \in \partial D_n$ minimising $\|xy\|$; we refer to this as $\mathcal{R}_0$. Next, we construct a new ray, $\mathcal{R}_1$ exiting x such that the area enclosed by $\mathcal{R}_0$, $\mathcal{R}_1$ and ∂D_n is $\log^{1+\varepsilon} n$; we let S_0 denote this region. (See Figure 4.1.) For n large enough, the angle between the rays $\mathcal{R}_0$ and $\mathcal{R}_1$ is smaller than $\frac{\pi}{2} + \frac{\pi}{12}$. In addition, the length of $\mathcal{R}_1$ is upper bounded by the diameter of $D_n \le c_2\sqrt{n}$ and lower bounded by $c_1\sqrt{\log n}$ for $c_1 = \frac{1}{2}$ since $\mathcal{R}_0$ is not longer than $\sqrt{\log n}$ by assumption.

We now define a determinstic process that finishes as soon as one of the regions (which we define shortly) is totally contained within a disc of radius $\log^{1+\varepsilon} n$ centred at x. At each step i, we look at the ray $\mathcal{R}_{i-1}$ and then grow a sector of a circle with area $\log^{1+\varepsilon} n$, and of radius $1/\sqrt{2} \cdot |\mathcal{R}_{i-1}|$, where $|\mathcal{R}_i|$ is the length of the ith ray. The sector of radius $|\mathcal{R}_{i+1}|$ defined by the rays $\mathcal{R}_i$ and $\mathcal{R}_{i+1}$ is denoted by S_i. Thus the internal angle of each new sector is exactly twice that of the previous one. For each sector apart from the first, we add a 'boundary' (shaded blue in Figure 4.1) which extends the each sector to the length of the sector proceeding it: let Q_i be the cone delimited by the rays $\mathcal{R}_i$ and $\mathcal{R}_{i+1}$, and of radius $|\mathcal{R}_i|$. Let I be the index of the first ray $\mathcal{R}_i$ for which $|\mathcal{R}_i| \le \sqrt{2}\log^{1+\varepsilon} n$, so that the last of this decreasing sequence of sectors is S_{I-1}. Finally, we add a sector S_I directly opposite $\mathcal{R}_0$ within a ball of radius $\log^{1+\varepsilon} n$. We choose its internal angle β so that its area is $\log^{1+\varepsilon} n$.

We now proceed to showing that each circular sector enclosed by two adjacent rays

contains a point with high probability and that when this event occurs, we have a bound on the region of points that may be a neighbour to x in $\mathrm{Del}(\mathbf{X})$.

Lemma 4.2. *Let α be the angle between the final ray $\mathcal{R}_I$ and the edge of the sector S_{I+1} opposite $\mathcal{R}_0$. Then, for n large enough, α is positive. This implies that the sectors S_i, $0 \le i \le I$ are disjoint.*

Proof. No angle between any two rays may exceed $2\beta = 4\pi \log^{-(1+\varepsilon)} n$ since the area of a every sector is $\log^{1+\varepsilon} n$, and the minimum circular radius of a sector is $1/\sqrt{2} \cdot \log^{1+\varepsilon} n$. Also, the angle between the last ray $\mathcal{R}_I$ and $\mathcal{R}_1$ is smaller than

$$\sum_{j=0}^{\infty} \frac{2\beta}{2^j} \le 4\beta.$$

Since for n large enough, the angle $\mathcal{R}_0\mathcal{R}_1$ is smaller than $\frac{\pi}{2} + \frac{\pi}{12}$ we obtain

$$\alpha \ge \frac{\pi}{2} - \frac{\pi}{12} - \frac{\beta}{2} - 4\beta$$

which is positive for n large enough. □

The construction given will now define a deterministic region containing all neighbours of a point given conditions that hold with high probability. Let $M \subset \mathbf{R}^2$ be defined by

$$M := B\Big(x, \sqrt{2}\log^{1+\varepsilon} n\Big) \cup \bigcup_{i=1}^{I} Q_i. \tag{4.2}$$

Thus, M is the union of all the (pink or blue) shaded regions in Figure 4.1.

Lemma 4.3. *Suppose that every for every $0 \le i \le I+1$, we have $|\mathbf{X} \cap S_i| > 0$, and that the axes are rotated so that $\mathcal{R}_0$ is in exactly in the direction of the y-axis. Then every neighbour of x in $\mathrm{Del}(\mathbf{X})$ having positive x-coordinate of must lie in M.*

Proof. Any neighbour y of x in $\mathrm{Del}(\mathbf{X})$ must have a circle not containing any point of $\mathbf{X}$ that touches both x and y. Let $y \notin M$. Since y has positive x-coordinate, it is easy to see that any circle touching both x and y must also fully contain one of the sectors S_i, $0 \le i \le I$, in pink in Figure 4.1 (see dotted circles in Figure 4.1). By assumption, each S_i, $0 \le i \le I$ contains at least one site of $\mathbf{X}$, so that no circle touching both x and y can be empty, and y cannot be a neighbour of x in $\mathrm{Del}(\mathbf{X})$. □

Lemma 4.4. *For any $\varepsilon > 0$ and n large enough,*

$$\mathbb{P}\left(\max\left\{ \delta_{\mathbf{X}}(x) \,:\, x \in \mathbf{X},\ \|\partial D_n\, x\| \le \sqrt{\log n} \right\} \ge \log^{2+3\varepsilon} n \right) \le 2\exp\left(-\log^{1+\varepsilon} n\right).$$

Proof. Let x_0 be chosen uniformly at random among the points of $\mathbf{X}$ within distance $\sqrt{\log n}$ of ∂D_n. Note that if there is no such point, then we may define

$$\max\Big\{ \delta_{\mathbf{X}}(x) : x \in \mathbf{X},\ \|\partial D_n\ x\| \le \sqrt{\log n} \Big\} := 0.$$

Let A_{x_0} be the event that no (pink) sector S_i, $0 \le i \le I$ about x_0 is empty and let B_{x_0} be the event that no sector S_i, $0 \le i \le I$ about x_0 contains more than $\log^{1+2\varepsilon} n$ points (see Figure 4.1). Given that the number of sectors $I+2$ about x_0 is deterministically bounded by $\frac{\pi}{6}\log^{1+\varepsilon} n \le n$ (for n large enough), we have by the union bound that the probability that

$$\mathbb{P}(A_{x_0}^c) \le n \cdot \exp\Big(-\log^{1+\varepsilon} \Big) \quad \text{and} \quad \mathbb{P}(B_{x_0}^c) \le n \cdot \exp\Big(-\log^{1+2\varepsilon} n\Big).$$

Conditional on A_{x_0} and B_{x_0} occurring, we may count the number of points that could possibly be Delaunay neighbours of x_0. This includes all points in the at most $\frac{\pi}{6}\log^{1+\varepsilon} n$ sectors, each containing at most $\log^{1+2\varepsilon} n$ points (conditional on B_{x_0}). We also add all the points not contained within any sector, but lying within the circle of radius $\sqrt{2}\log^{1+\varepsilon}$ about x_0 (shaded blue in Figure 4.1). Lemma 2.25 may now be applied to show that this region contains no more than $2\pi\log^{2+3\varepsilon} n$ points with probability bounded at most $\exp(-\log^{2+3\varepsilon})$. Putting these together and applying the union bound we have

$$\begin{aligned}
&\mathbb{P}\left(\max\Big\{ \delta_{\mathbf{X}}(x) : x \in \mathbf{X},\ \|\partial D_n\ x\| \le \sqrt{\log n} \Big\} > \log^{2+3\varepsilon} n\right) \\
&\le 2n \cdot \mathbb{P}\left(\delta_{\mathbf{X}}(x_0) > \log^{2+3\varepsilon} n\right) + \mathbb{P}\left(|\mathbf{X}| > 2n\right) \\
&\le 2\exp\Big(-\log^{1+\varepsilon} n\Big),
\end{aligned}$$

for n sufficiently large. □

The second case of our proof is much simpler, since it suffices to bound the maximum distance between any two Delaunay neighbours, and then count the maximum number of points falling within this region.

Lemma 4.5. *For any $\varepsilon > 0$ and n large enough,*

$$\mathbb{P}\left(\max\Big\{ \delta_{\mathbf{X}}(x)\ :\ x \in \mathbf{X},\ \|\partial D_n\ x\| >\ \sqrt{\log n} \Big\} \ge \log^{2+\varepsilon} n\right) \le \exp\Big(-\log^{1+\varepsilon/3} n\Big).$$

Proof. Let $x_0 \in \mathbf{X}$ be chosen uniformly at random among the points of $\mathbf{X}$ at distance more than $\sqrt{\log n}$ from ∂D_n. Again, note that if there is no such point, we will take

$$\max\Big\{ \delta_{\mathbf{X}}(x)\ :\ x \in \mathbf{X},\ \|\partial D_n\ x\| >\ \sqrt{\log n} \Big\} := 0. \tag{4.3}$$

If x_0 is well-defined, any neighbour of x_0 in $\mathrm{Del}(\mathbf{X})$ outside of the ball $B(x_0, 1/2\log^{1/2+\varepsilon} n)$ implies the existence of large region of D_n that is empty of points of $\mathbf{X}$. By adapting the proof of Lemma 5.20 and using Lemma 5.11, the probability that such a neighbour exists may be bounded by $\exp(-\log^{1+\varepsilon} n)$. We omit the details.

We now upper bound the number of points that may fall within this ball. Split the domain into a regular grid with cells of side length $1/2 \cdot \log^{1/2+\varepsilon} n$. The probability that any of these grid cells contains more than $\log^{2+\varepsilon/3}$ points is bounded by $\exp(-\log^{2+2\varepsilon} n)$ for large n, and our ball may intersect at most four of these, so the degree of x_0 in $\mathrm{Del}(\mathbf{X})$ is bounded by $4\log^{2+2\varepsilon} n$ with probability $2\exp(-\log^{1+\varepsilon} n)$ in this case. The result follows from the union bound, just as in Lemma 4.4. □

Chapter 5

Cone Walk

In this chapter we give a new deterministic planar graph navigation algorithm which we call *Cone Walk* that succeeds on any Delaunay triangulation and produces a path which is 3.7-competitive.[1] We provide a detailed average-case analysis for Cone Walk, demonstrating that it is efficient even for queries near the boundary of the domain. We briefly underline the fact that our algorithm has been designed for theoretical demonstration, and we do not claim that it would be faster in a practical sense than, for example, *Greedy Walk* or *Visibility Walk*. On the other hand, direct comparisons would perhaps be unfair, since *Greedy Walk* is not $O(1)$-competitive on the Delaunay triangulation [16] whereas we prove that Cone Walk is; and *Straight Walk* is not memoryless in any sense, whilst Cone Walk is localised in the sense given by Theorem 5.1. The theorems that follow provide a detailed understanding for the average-case analysis of Cone Walk.

Let D be a smooth convex domain of the plane with area 1, and write $D_n := \sqrt{n}D$ for its scaling to area n. For $x, y \in D$, let $\|xy\|$ denote the Euclidean distance between x and y. Under the hypothesis that the input is the Delaunay triangulation of n points uniformly distributed in a convex domain of unit area, we prove that, for any $\varepsilon > 0$, our algorithm is $O(\log^{1+\varepsilon} n)$-memoryless with probability tending to one. In the case of Cone Walk, this is equivalent to bounding the number of *neighbourhoods* that might be accessed during a step, which we deal with in the following theorem.

Theorem 5.1. *Let $\mathbf{X}_n := \{X_1, X_2, \ldots, X_n\}$ be a collection of n independent uniformly random points in D_n. For $z \in \mathbf{X}_n$ and $q \in D_n$, let $M(z,q)$ be the maximum number of neighbourhoods needed to compute every step of the walk. Then, for every $\epsilon > 0$,*

$$\mathbb{P}\left(\exists z \in \mathbf{X}_n, q \in D_n : M(z,q) > \log^{3+\epsilon} n\right) \leq \frac{1}{n}.$$

[1] This work was completed in collaboration with Nicolas Broutin and Olivier Devillers. An extended abstract of the work was published in the AofA conference in 2014. The full version has been submitted to Random Structures and Algorithms awaiting review [23].

In particular, as $n \to \infty$,

$$\mathbb{E}\left[\sup_{z \in \mathbf{X}_n, q \in D_n} M(z,q)\right] = O(\log^{3+\epsilon} n),$$

for every $\epsilon > 0$.

Also with probability close to one, we show that the path length, the number of edges and the number of vertices accessed are $O(\,\|zq\| + \log^6 n\,)$ for any pair of points in the domain. We formalise these properties in the following theorem.

Theorem 5.2. *Let* $\mathbf{X}_n := \{\, X_1, X_2, \ldots, X_n \,\}$ *be a collection of* n *independent uniformly random points in* D_n. *Let* $\Gamma(z,q)$ *denote either the Euclidean length of the path generated by the Cone Walk from* $z \in \mathbf{X}_n$ *to* $q \in D_n$, *its number of edges, or the number of vertices accessed by the algorithm when generating it. Then there exist constants* $C_{\Gamma,D}$ *depending only on* Γ *and on the shape of* D *such that, for all* n *large enough,*

$$\mathbb{P}\left(\exists z \in \mathbf{X}_n,\, q \in D_n \,:\, \Gamma(z,q) > C_{\Gamma,D} \cdot \|zq\| + 4(1 + \sqrt{\|zq\|}) \log^6 n\right) \leq \frac{1}{n}.$$

In particular, as $n \to \infty$,

$$\mathbb{E}\left[\sup_{z \in \mathbf{X}_n,\, q \in D_n} \Gamma(z,q)\right] = O(\,\sqrt{n}\,).$$

Finally, we bound the number of steps required by the algorithm, which we denote $T(z,q)$.

Theorem 5.3. *Let* $\mathbf{X}_n := \{X_1, X_2, \ldots, X_n\}$ *be a collection of* n *independent uniformly random points in* D_n. *Then in the RAM model of computation, there exists a constant* C *depending only on the shape of* D *and the particular implementation of the algorithm such that for all* n *large enough,*

$$\mathbb{P}\left(\exists z \in \mathbf{X}_n,\, q \in D_n \,:\, T(z,q) > C\|zq\| \log\log n + \left(1 + \sqrt{\|zq\|}\right) \log^6 n\right) \leq \frac{1}{n}.$$

In particular, as $n \to \infty$,

$$\mathbb{E}\left[\sup_{z \in \mathbf{X}_n,\, q \in D_n} T(z,q)\right] = O(\,\sqrt{n} \log\log n\,).$$

Remark 5.4. The choice of the initial vertex is never discussed. However, previous results show that choosing this point carefully can result in an expected asymptotic speed up for any graph navigation algorithm [92].

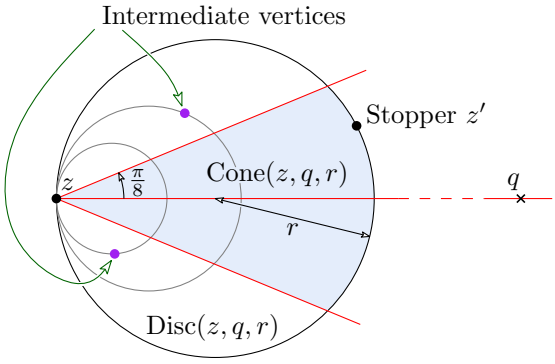


Figure 5.1 – Choosing the next vertex.

Layout of the chapter

In Section 5.1, we give a precise definition of the Cone Walk algorithm and prove some important geometric properties. In Section 5.2, we begin the analysis for the Cone Walk algorithm applied to a homogeneous planar Poisson process. To avoid problems when the walk goes close to the boundary, we provide an initial analysis which assumes that the points are sampled from a disc with the query point at its centre. This analysis is then extended to arbitrary query points in the disc and also to other convex domains in Section 5.3. In Section 5.4, we prove estimates about an auxiliary line arrangement which are crucial to proving the worst-case probabilistic bounds in Theorems 5.1 and 5.3. Finally, we compare our findings with computer simulations in Section 5.5.

5.1 Algorithm and geometric properties

We consider the finite set of sites in general position (which we recall means that no three points are co-linear, and no four points are co-circular), $\mathbf{X} \subset \mathbf{R}^2$ contained within a compact convex domain $D \subset \mathbf{R}^2$. We also recall that $\mathrm{Del}(\mathbf{X})$ is used to denote the Delaunay triangulation of $\mathbf{X}$, which is the graph in which three sites $x, y, z \in \mathbf{X}$ form a triangle if and only if the disc with x, y and z on its boundary does not contain any site in $\mathbf{X}$. Given two points $z, q \in \mathbf{R}^2$ and a number $r \in \mathbf{R}$ we define $\mathrm{Disc}(z, q, r)$ to be the closed disc whose diameter spans z and the point at a distance $2r$ from z on the ray zq. Finally, we define $\mathrm{Cone}(z, q, r)$ to be the sub-region of $\mathrm{Disc}(z, q, r)$ contained within a closed cone of apex z, axis zq and half angle $\frac{\pi}{8}$ (see Figure 5.1).

The Cone Walk algorithm

Given a site $z \in \mathbf{X}$ and a destination point $q \in D$, we define one *step* of the Cone Walk algorithm by growing the region $\mathrm{Cone}(z, q, r)$ anchored at z from $r = 0$ until the first point $z' \in \mathbf{X}$ is found such that the region is non-empty. Once z' has been determined, we refer to it as the *stopper*. We call the region $\mathrm{Cone}(z, q, r)$ for the given r a *search cone*, and we call the associated disc $\mathrm{Disc}(z, q, r)$ the *search disc* (see Figure 5.1). The point z' is then

selected as the anchor of a new search cone $\text{Cone}(z', q, \cdot)$ and the next step of the walk begins. See Figure 5.2 for an example run of the algorithm.

To find the stopper using only neighbour incidences in the Delaunay triangulation, we need only access vertices in a well-defined local neighbourhood of the search disc. Define the points $\mathbf{X} \cap \text{Disc}(z, q, r) \setminus \{z, z'\}$ to be the *intermediate vertices*. The algorithm finds the stopper at each step by gradually growing a disc anchored at z in the direction of the destination, adding the neighbours of all vertices in $\mathbf{X}$ intersected along the way. This is achieved in practice by maintaining a series of candidate vertices initialised to the neighbours of z and selecting amongst them the vertex defining the smallest search disc at each iteration. Each time we find a new vertex intersecting this disc, we check to see if it is contained within $\text{Cone}(z, q, \infty)$. If it is, this point is the next stopper and this step is finished. Otherwise the point must be an intermediate vertex and we add its neighbours to the list of candidate vertices. This procedure works because the intermediate vertex defining the next largest disc is always a neighbour of one of the intermediate vertices that we have already visited during the current step (see Lemma 5.6).

We terminate the algorithm when the destination q is contained within the current search disc for a given step. At this point we know that one of the points contained within $\text{Disc}(z, q, r)$ is a Delaunay neighbour of q in $\text{Del}(\mathbf{X} \cup \{q\})$. We can further compute the triangle of $\text{Del}(\mathbf{X})$ containing the query point q (point location) or find the nearest neighbour of q in $\text{Del}(\mathbf{X})$ by simulating the insertion of the point q into $\text{Del}(\mathbf{X})$ and performing an exhaustive search on the neighbours of q in $\text{Del}(\mathbf{X} \cup \{q\})$.

We will sometimes distinguish between the *visited* vertices, which we take to be the set of all sites contained within the search discs for every step and the *accessed* vertices, which we define to be the set of all vertices accessed by the cone-walk algorithm. Thus the accessed vertices are the visited vertices along with their 1-hop neighbourhood.

The pseudo-code below gives a detailed algorithmic description of the Cone-Walk algorithm. We take as input some $z \in \mathbf{X}$, $q \in D$ and return a Delaunay neighbour of q in $\text{Del}(\mathbf{X} \cup \{q\})$. Recalling that $\text{neighbours}_1(v)$ refers to the Delaunay neighbours of $v \in \text{Del}(\mathbf{X})$ and additionally defining Next-Vertex(S, z, q) to be the procedure that returns the vertex in S with the smallest r such that $\text{Disc}(z, q, r)$ touches a vertex in S and In-Cone(z, q, y) to be `true` when $y \in \text{Cone}(z, q, \infty)$.

```
Cone-Walk(z, q)
 1  Substeps = {z}
 2  Candidates = neighbours_1(z)
 3  while true
 4      y = Next-Vertex(Candidates ∪ {q}, z, q)
 5      if In-Cone(z, q, y)
 6          if y = q
 7              // Destination reached.
 8              return Next-Vertex(Substeps, q, z)
 9          // y is a stopper
10          z = y
11          Substeps = {z}
12          Candidates = neighbours_1(z)
13      else
14          // y is an intermediate vertex.
15          Substeps = Substeps ∪ {y}
16          Candidates = Candidates ∪ neighbours_1(y) \ Substeps
```

Path Generation

We note that the order in which the vertices are discovered during the walk does not necessarily define a path in Del($\mathbf{X}$). If we only wish to find a point of the triangulation that is close to the destination (for example, in point location), this is not a problem. However, in the case of routing, a path in the triangulation is required to provide a route for data packets. To this end, we provide two options that we shall refer to as Simple-Path and Competitive-Path. Simple-Path is a simple heuristic that can quickly generate a path that is provably short on average. We conjecture that Simple-Path is indeed competitive, however we were unable to prove this. Competitive-Path is slightly more complex from an implementation point of view, however we show that for any possible input the algorithm will always generate a path of constant competitiveness whilst still maintaining the same asymptotic behaviour under the point distribution hypotheses explored in Section 5.2.

Simple-Path A simple way to generate a valid path is to keep a predecessor table for each vertex. We start with an empty table at the beginning of each step, and then every time we access a new vertex, we store it in the table along with the vertex that we accessed it from. To trace a path back, we simply follow the predecessors.

Competitive-Path Let Z_i for $i > 0$, be the ith stopper in the walk thus, Z_i is the stopper found at step i) and $Z_0 := z$. For a path to be competitive, it should at least be *locally* competitive: for each step, there should be a bound on the length of the path generated between Z_i and Z_{i+1}, which does not depend on the points in the search disc. To construct a path verifying this property, we use the fact that the *stretch factor* of the Delaunay triangulation is bounded above by a constant, λ. This means that for any two sites x, y, there exists a path from x to y in the Delaunay triangulation for which the sum of the lengths of the edges is at most $\lambda\|xy\|$. Currently the literature gives us that the stretch factor is in $[1.5932, 1.998]$ [112, 113]. Clearly this implies that there exists a path between Z_i and Z_{i+1} with total length at most $\lambda\|Z_iZ_{i+1}\|$, and this path cannot exit the ellipse

$\text{Ell}(Z_i, Z_{i+1}) := \{x \in \mathbf{X} : \|xZ_i\| + \|xZ_{i+1}\| < \lambda\|Z_iZ_{i+1}\|\}$. We use Dijkstra's algorithm to find the shortest path between Z_i and Z_{i+1} which uses only vertices within $\text{Ell}(Z_i, Z_{i+1})$. The resulting path implicitly has stretch bounded by λ. We show in Lemma 5.5 that this algorithm results in a bound for the competitiveness for the full path.

Lemma 5.5. Cone-Walk *is* 3.7*-competitive when the* Competitive-Path *algorithm is used to generate the path in* $\text{Del}(\mathbf{X})$

Proof. Let Z_i, Z_{i+1} be the stoppers of two consecutive steps defined by the algorithm. The stretch factor bound guarantees that the path generated between Z_i and Z_{i+1} has length bounded by $\lambda\|Z_iZ_{i+1}\|$, meaning that the longest path can have stretch at most $\lambda \sum_{i=0}^{\tau-1} \|Z_iZ_{i+1}\|/\|zq\|$ where τ is the number of steps in the walk. We bound this sum by observing that $\|Z_iZ_{i+1}\| \leq 2\cos\frac{\pi}{8} \cdot (\|Z_iq\| - \|Z_{i+1}q\|)$, which follows from Figure 5.8. Finally, no path defined by the algorithm can be longer than

$$\lambda \sum_{i=0}^{\tau-1} \|Z_iZ_{i+1}\| \leq 2\lambda\cos\tfrac{\pi}{8} \sum_{i=0}^{\tau-1} \Big(\|Z_iq\| - \|Z_{i+1}q\|\Big) \leq 2\lambda\cos\tfrac{\pi}{8} \cdot \|zq\|$$

Thus the path is c-competitive for $c := 2\lambda\cos\frac{\pi}{8} \leq 4\cos\frac{\pi}{8} \leq 3.7$. □

Complexity

In this section we give deterministic bounds on the number of operations required to compute Cone-Walk(z, q) within the RAM model of computation. In this model, accessing, comparing and performing arithmetic on points is treated as atomic. We will use these deterministic bounds to extract probabilistic bounds under certain distribution assumptions in Section 5.2. For now, we focus on a single step of the walk starting from $y \in \mathbf{X}$, and resulting in a disc with radius r. Let k be the number of points intersecting the disc $\text{Disc}(y, q, r)$ and m be the number of edges in $\text{Del}(\mathbf{X})$ intersecting $\partial\,\text{Disc}(y, q, r)$ (where we use the notation ∂A to denote the boundary of A) .

We note that every intermediate vertex will add its neighbours to the list of *Candidates* when visited. Each of these insertions can be associated with a single edge of $\text{Del}(X)$ intersecting $\text{Disc}(y, q, r)$ (with multiplicity two for each 'internal' edge, since they are accessed from both sides). By the Euler relation, the total number of such insertions for one step is thus at most $3(m + 2k)$. In addition, we observe that when moving from one intermediate vertex to the next, a search in the list of *Candidates* is required. A simple linear search requires $O(m+k)$ operations for each intermediate vertex. Combining this with the above, we achieve a bound of $O(k(m + k))$ operations for one step. This bound may be improved by replacing *Candidates* with a priority queue keyed on the associated search-disc radius of each candidate, which yields a simple improvement to $O(k \log(m + k))$.

For the path generation algorithms, we observe that Simple-Path only requires a constant amount of processing per vertex accessed to generate the predecessor table and $O(k)$ time to output the path at the end of each step, so the asymptotic running time is not affected by its inclusion. Competitive-Path is slightly more complicated since it accesses all points within an ellipse enclosing each search disc. Let k' be the number of points

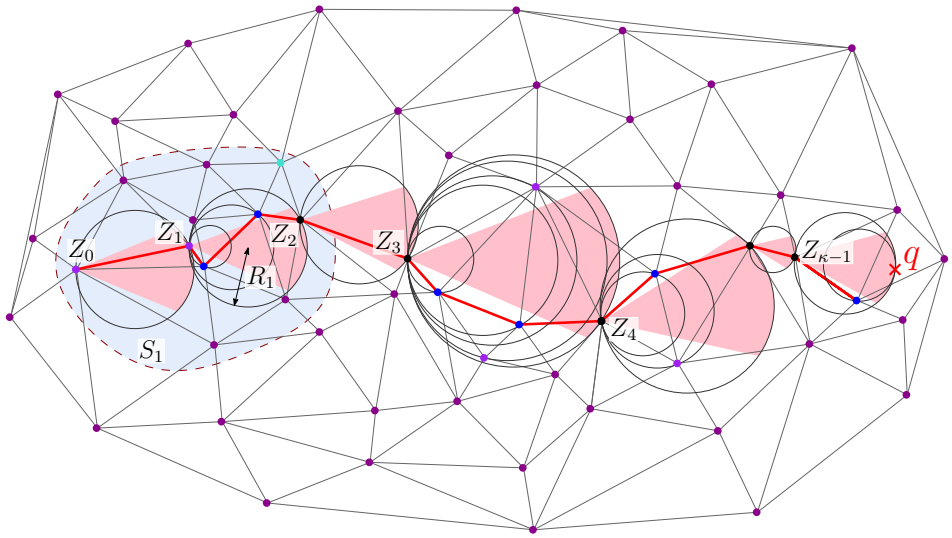


Figure 5.2 – An example of cone walk. The points $(Z_i)_{i>0}$ are the *stoppers* (or sometimes, the *steps*) and the points within each circle are the *intermediate vertices*. The shaded conic regions are the *cones*, and the set of outer circles for each of the steps in the diagram is referred to as the *discs*.

in this ellipse along with their neighbours. The path is found by applying Dijkstra's algorithm to k' points, applying Euler's relation gives us an updated bound for a single step of $O(k' \log k')$.

Geometric properties

We now prove a series of geometric lemmata giving properties of steps in the walk. We begin with a small lemma that will guarantee that we never get 'stuck' when performing a search for the next step, thus demonstrating correctness of the algorithm. The following two 'overlapping' lemmata allow us to establish which regions may be considered independent in a probabilistic sense and will be important in Section 5.2. Finally we provide a 'stability' result, which will help us to bound the region in which a destination point may be moved without changing the sequence of steps taken by the algorithm. This will be important when we enumerate the number of different walks possible for a given set of input points.

Finding a Delaunay path within the discs

Lemma 5.6 (Path finding lemma)**.** *Let $q \in D$, $z \in \mathbf{X}$ and $y' \in \mathbf{X}$ with associated disc* $\mathrm{Disc}(z,q,r')$*. Suppose there exists an $r > 0$ such that* $(\mathrm{Disc}(z,q,r') \setminus \mathrm{Disc}(z,q,r)) \cap \mathbf{X} = \{y'\}$*. Then there exists a point in* $\mathrm{Disc}(z,q,r)$ *that is a Delaunay neighbour of y'.*

Proof. Let γ' be the centre of $\mathrm{Disc}(z,q,r')$. We grow $\mathrm{Disc}(y',\gamma',\rho) \subset \mathrm{Disc}(z,q,r')$ until we hit a point w in $\mathbf{X}$. The point w is always contained within $\mathrm{Disc}(z,q,r)$ because z is on

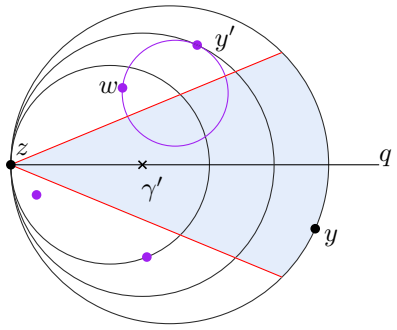


Figure 5.3 – We observe that y' has always a Delaunay neighbour in $\mathrm{Disc}(z,q,r)$, where r is the radius ensuring $y \in \partial\mathrm{Disc}(z,q,r)$.

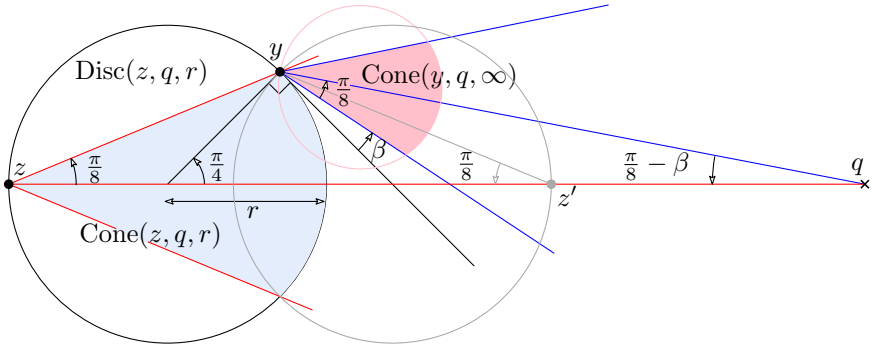


Figure 5.4 – For the proof of Lemma 5.8.

the border of $\mathrm{Disc}(z,q,r)$. Since the interior of the $\mathrm{Disc}(y',\gamma',\rho)$ is empty, w is a Delaunay neighbour of y'. See Figure 5.3. □

Corollary 5.7. *Let $q \in D$, $z \in \mathbf{X}$ with $y \in \mathbf{X}$ its associated stopper satisfying $y \in \partial\mathrm{Cone}(z,q,r)$. Then there is a path of edges of* $\mathrm{Del}(\mathbf{X})$ *between z and y contained within* $\mathrm{Disc}(z,q,r)$.

Independence of the search cones

When growing a new search cone, it is important to observe that it does not overlap any of the previous search cones, except at the very end of the walk. This is formalised by the following lemma.

Lemma 5.8 (Non-overlapping lemma)**.** *Let z and y be two points of $\mathbf{X}$ and $r > 0$ such that* $\mathrm{Cone}(z,q,r)$ *has y on its boundary. If $\|zq\| > (2+\sqrt{2})r$ then* $\mathrm{Disc}(z,q,r)$ *does not intersect the search cone* $\mathrm{Cone}(y,q,\infty)$ *issued from y nor any other search cone for any subsequent step of the walk.*

Proof. Assume without loss of generality that y lies to the left of line zq and consider the construction given in Figure 5.4. Let β denote the angle between the tangent to $\mathrm{Disc}(z,q,r)$ at y and the ray bordering $\mathrm{Cone}(y,q,\infty)$. $\mathrm{Cone}(y,q,\infty)$ and $\mathrm{Disc}(z,q,r)$ do not intersect

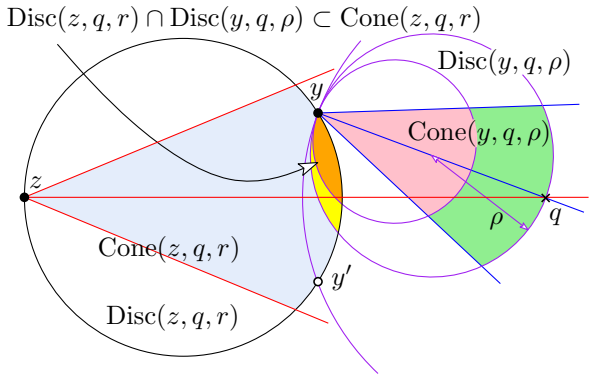


Figure 5.5 – For the proof of Lemma 5.9.

provided that $\beta \geq 0$. Placing y at the corner of $\mathrm{Cone}(z,q,r)$ maximizes β, in which case we have $\beta > 0$ if and only if q is to the right of z', the point symmetrical to z with respect to the line through y perpendicular to zq. Elementary computations then yield the result. Since the whole sequence of search cones following the one issued from y remains in $\mathrm{Cone}(y,q,\infty)$, $\mathrm{Disc}(z,q,r)$ does not intersect any of these search cones, and the result follows. □

Independence of the search discs

When growing the search disc region, the new search disc may overlap previous search discs but only in their cone parts. This is formalised by the following lemma:

Lemma 5.9 (Overlapping lemma)**.** *Let z and y be two points such that $\mathrm{Cone}(z,q,r)$ has y on its boundary. Then if the search disc $\mathrm{Disc}(y,q,\rho)$ issued from y does not contain q, it does not intersect $\mathrm{Disc}(z,q,r) \setminus \mathrm{Cone}(z,q,r)$.*

Proof. By symmetry we observe that $\mathrm{Disc}(y,q,\rho)$ only intersects the point y', the point y reflected through the line zq, when the centre of $\mathrm{Disc}(y,q,\rho)$ coincides with q (See Figure 5.5). Since the algorithm terminates as soon as the current search disc touches q, q is never contained within $\mathrm{Disc}(z,q,\rho)$ and thus this can never happen. □

Stability of the walk

In the following lemma we are interested in the stability of the sequence of steps to reach q.

Lemma 5.10 (Invariance lemma)**.** *For an n-set $\mathbf{X} \subseteq \mathbf{R}^2$, there exists an arrangement of half-lines $\Xi = \Xi(\mathbf{X})$ such that the associated subdivision of the plane has fewer than $2n^4$ cells, and such that the sequence of steps used by the Cone Walk algorithm from any vertex of $\mathbf{X}$ does not change when the aim q moves in a connected component of $\mathbf{R}^2 \setminus \Xi$.*

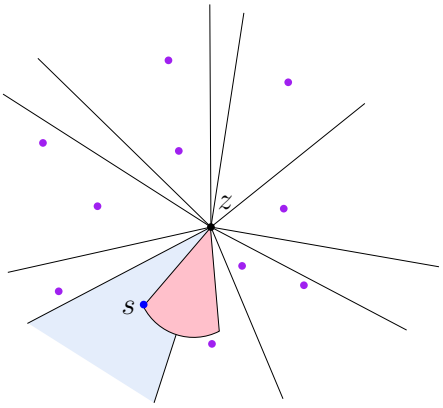


Figure 5.6 – For the proof of Lemma 5.10. For a given step z, moving the destination in the shaded sector will always result in the same stopper, s, being chosen for the next step.

Proof. Take a point $z \in \mathbf{X}$ and consider $\mathcal{S}_z$, the set of all possible stoppers defined by $\mathrm{Cone}(z, q, r)$ for some $q \in D$ and $r > 0$. Each $s \in \mathcal{S}_z$ defines a unique sector about z such that moving a point in the given sector does not change the stopper (see Figure 5.6). We then create an arrangement by adding a ray on the border of every sector for each point $z \in \mathbf{X}$. The resulting arrangement has the property that moving the destination point q within one of the cells of the arrangement does not change the stopper of any step for any possible walk. Clearly, $|\mathcal{S}_z| \leq n - 1$ for all $z \in \mathbf{X}$, and each sector is bounded by at most two rays, thus there are at most $2n(n-1)$ rays in the arrangement. Since an arrangement of m lines has at most $\frac{m^2+m+2}{2}$ cells the result follows [see, e.g., 86, p. 127]. □

5.2 Cone Walk on Poisson Delaunay in a disc

Our aim in this section is to prove the main elements towards Theorem 5.3, which we go on to complete in Section 5.3. Our ultimate goal is to prove bounds on the behaviour of the Cone Walk for the *worst* possible pair of starting point and query when the input sites are generated by a homogeneous Poisson process in a compact convex domain. Achieving this requires first strong bounds on the probability that the walk behaves badly for a fixed start point and query. One then proves the worst-case bounds by showing that to control every possible run of the algorithm, it suffices to bound the behaviour of the walk for enough pairs of starting points and query; this relies crucially on the arrangement of Lemma 5.10. The tail bounds required in the second stage of the proof may not be obtained from Markov or Chebyshev's inequalities together with mean or variance estimates only, and we thus need to resort to stronger tools.

Our techniques rely on *concentration inequalities* [21, 51, 70, 88]. Most of the bounds we obtain (for the number of steps κ and the number of visited sites) follow from a representation as a sum of random variables in which the increments can be made independent by a simple and natural conditioning. The bounds on the complexity of the algorithm Cone-Walk are slightly trickier to derive because there is no way to make the increments

independent. For the sake of presentation, we introduce two simplifications which we remove in Section 5.3. First, we start by studying the walk in the disc D_n of area n where the query is at the centre. These choices for D_n and q ensure that for any $z \in D_n$ and any $r \leq \sqrt{n/\pi}$, we have $\mathrm{Disc}(z,q,r) \subset D_n$. Note that since the distance to the aim is decreasing, the disc is precisely the *effective* domain where the walk from z and aiming at q takes place.

Then, we introduce independence between the different regions of the domain by replacing the collection of independent points $\mathbf{X}_n$ by a (homogeneous) Poisson point process $\mathbf{X}$ and consider $\mathrm{Del}(\mathbf{X})$. Recall that a Poisson point process of intensity 1 is a random collection of points $\mathbf{X} \subset D_n$ such that with probability one, all the points are distinct, for any two Borel sets $R, S \subseteq D_n$, the number of points $|\mathbf{X} \cap R|$ is distributed like a Poisson random variable whose mean is the area $\lambda_2(R)$ of R, and if $R \cap S = \varnothing$ then $|\mathbf{X} \cap R|$ and $|\mathbf{X} \cap S|$ are independent.

On many occasions, it is convenient to consider $\mathbf{X}$ conditioned to have a point located at $z \in D$ and we let $\mathbf{X}_z$ be the corresponding random point set. We recall that Slivnyak's theorem ensures that $\mathbf{X}_z \setminus \{z\}$ is distributed like $\mathbf{X}$, so that one can take $\mathbf{X}_z = \mathbf{X} \cup \{z\}$, for $\mathbf{X}$ independent of z (see Theorem 2.20 Part (d).)

Notation

We establish the following notation (see Figure 5.2). Let $\mathcal{Z} := (Z_i, i > 0)$ denote the sequence of stoppers visited during the walk with $Z_0 := z$. Let $L_i = \|Z_i q\|$ denote the distance to the destination q. The distance L_i is strictly decreasing and the point set $\mathbf{X}$ is almost surely finite, thus ensuring that the walk stops after a finite number of steps κ, at which point we have $Z_\kappa = q$. For $x > 0$, we also let $\kappa(x)$ be the number of steps required to reach a point within distance x of the query. Therefore $i < \kappa(x)$ if and only if $L_i > x$. The important parameters needed to track the location and progress of the walk are the radius R_i such that $Z_{i+1} \in \partial\mathrm{Cone}(Z_i, q, R_i)$, and the angle α_i between $Z_i q$ and $Z_i Z_{i+1}$. $\mathrm{Disc}(Z_i, q, R_i)$ may contain several points of $\mathbf{X}$, let τ_i denote $|\mathrm{Disc}(Z_i, q, R_i) \setminus \{Z_i, Z_{i+1}\} \cap \mathbf{X}|$ the number of such points and N_i the number of these points along with their Delaunay neighbours.

In order to compute the walk efficiently, the algorithm presented gathers a lot of information. In particular, we access all of the points in $\mathrm{Disc}(Z_i, q, R_i)$ and their neighbours. For the analysis, we want to keep the landscape as concise as possible, and so we define a filtration which only contains the necessary information for the walk to be a measurable process. Let $\mathcal{F}$ denote the information consisting of (the σ-algebra generated by) the locations of the points of $\mathbf{X}$ contained in $\cup_{j=0}^{i} \mathrm{Disc}(Z_j, q, R_j)$. Finally, we shall write ω_n to denote a sequence satisfying $\omega_n \geq \log n$.

We often need to condition on the size of the largest empty ball within the process $\mathbf{X}_n$. This is dealt with in the following lemma.

Lemma 5.11. *and $B(\mathbf{x}, r)$ denotes the closed ball of radius r centred at $\mathbf{x}$. Then $\forall c > 0$, $\xi > 0$,*

$$\mathbb{P}\left(\exists x \in D_n : B\left(x, c\,\omega_n^{1/2+\xi}\right) \cap \mathbf{X}_n = \varnothing\right) \leq \exp\left(-\omega_n^{1+\xi}\right)$$

for n sufficiently large.

Proof. We have

$$\mathbb{P}(\exists x \in D_n : B(x, c\,\omega_n^{1/2+\xi}) \cap \mathbf{X}_n = \varnothing) \leq \mathbb{P}(\exists B \in P : B \cap \mathbf{X}_n = \varnothing),$$

where P is any maximal packing of D with balls B of radius $\frac{1}{2}c\,\omega_n^{1/2+\xi}$ centred in D_n. If the radius of curvature of D_n is lower bounded by $c\,\omega_n^{1/2+\xi}$ (which happen for n large enough) such a ball B contains a ball of radius $\frac{1}{4}c\,\omega_n^{1/2+\xi}$ entirely inside D_n, For n large enough, any such packing contains at most n balls and we have

$$\mathbb{P}(\exists x \in D_n : B(x, c\,\omega_n^{1/2+\xi}) \cap \mathbf{X}_n = \varnothing) \leq n \exp\left(-\pi \tfrac{1}{4^2}\right) c^2 \omega_n^{1+2\xi} \leq \exp\left(-\omega_n^{1+\xi}\right).$$

□

The size of the discs

If the search cone $\mathrm{Cone}(Z_i, q, \infty)$ does not intersect any of the previous discs, the region which determines R_{i+1} is 'fresh' and R_{i+1} is independent of $\mathcal{F}_i$. Lemma 5.8 provides a condition which guarantees independence of the search cones. To take advantage of it, we write $\xi := 2 + \sqrt{2}$, and for $i \geq 0$, define the event

$$G_i := \{\forall j \leq i+1, R_j < \omega_n/\xi\}, \tag{5.1}$$

Then if the event $G_i^\star := G_i \cap \{L_i \geq \omega_n\}$ occurs; for every $j \leq i$, the search-cone $\mathrm{Cone}(Z_j, q, \infty)$ does not intersect any of the regions $\mathrm{Disc}(Z_k, q, R_k)$, $0 \leq k < j$, and the corresponding variables (R_j, α_j), $0 \leq j \leq i+1$ are independent. Although it might seem like an odd idea, $G_i^\star$ does include some condition on R_{i+1}; this ensures that on $G_i^\star$, we have $L_{i+1} > L_i - 2R_{i+1} > 0$, so that $i+1$ is not the last step. So for $x > 0$ we have

$$\begin{aligned}\mathbb{P}(R_{i+1} > x \mid \mathcal{F}_i, G_i^\star) &= \mathbb{P}(\mathbf{X} \cap \mathrm{Cone}(Z_i, q, x) \setminus \{Z_i\} = \varnothing \mid \mathcal{F}_i, G_i^\star) \mathbb{1}_{\xi x \leq \omega_n} \\ &= \exp\left(-Ax^2\right) \mathbb{1}_{\xi x \leq \omega_n}, \end{aligned} \tag{5.2}$$

where A denotes the area of $\mathrm{Cone}(z, q, 1)$ which is the shaded region in Figure 5.1. Indeed, conditional on $\mathcal{F}_i$ and $G_i^\star$, $|\mathbf{X} \cap \mathrm{Cone}(Z_i, q, x) \setminus \{Z_i\}|$ is a Poisson random variable with mean Ax^2 where

$$A := 2\left(\cos\frac{\pi}{8}\sin\frac{\pi}{8} + \frac{\pi}{8}\right) = \frac{\sqrt{2}}{2} + \frac{\pi}{4}. \tag{5.3}$$

We will repeatedly use the conditioning on G_i to introduce independence, and it is important to verify that G_i indeed occurs with high probability. For G_i to fail, there must be a first step j for which $R_j \geq \omega_n/\xi$. Writing G_i^c for the complement of G_i and defining G_{-1} to be a void conditioning: provided that $i = O(n)$ (which will always be the case in the following)

$$\begin{aligned}\mathbb{P}(G_i^c) &\leq \sum_{0 \leq j \leq i+1} \mathbb{P}(R_j \geq \omega_n/\xi \mid G_{j-1}) \\ &\leq \exp\left(\log O(n) - A\omega_n^2/\xi^2\right) \\ &\leq \exp\left(-\omega_n^{3/2}\right) \end{aligned} \tag{5.4}$$

for all n large enough since $\omega_n \geq \log n$.

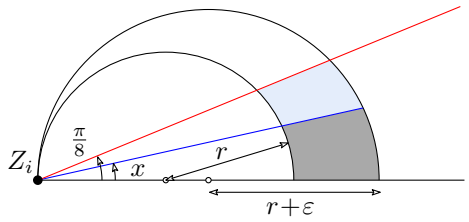


Figure 5.7 – For the angle to be smaller than x given $R_{i+1} \in [r, r+\varepsilon]$, the stopper must fall within the dark shaded region

Remark about the notation. It is convenient to work with an "ideal" random variable that is not constrained by the location of the query or artificially forced to be at most ω_n/ξ, and we define $\mathcal{R}$ by $\mathbb{P}(\mathcal{R} \geq x) = \exp(-Ax^2)$ for $x \geq 0$. In the course of the proof, we use multiple other such ideal random variables, to distinguish them from the ones arising from the actual process, we use calligraphic letters to denote them.

The progress for one step

We now focus on the distribution of the angle $\angle qZ_iZ_{i+1}$ and by extension the progress made during one step in the walk. Let $\mathrm{Cone}_\alpha(z,q,r)$ be the cone of half angle α with the same apex and axis as $\mathrm{Cone}(z,q,r)$. On the event $G_i^\star$, $Z_{i+1} \neq q$ and α_{i+1} is truly random and its distribution is symmetric and given by (see Figure 5.7)

$$\begin{aligned}
\mathbb{P}\left(|\alpha_{i+1}| < x \;\middle|\; R_{i+1} = r, \mathcal{F}_i, G_i^\star\right) &= \lim_{\varepsilon\to 0} \frac{\lambda_2\Big(\mathrm{Cone}_x(Z_i,q,r+\varepsilon) \setminus \mathrm{Cone}_x(Z_i,q,r)\Big)}{\lambda_2\Big(\mathrm{Cone}(Z_i,q,r+\varepsilon) \setminus \mathrm{Cone}(Z_i,q,r)\Big)} \\
&= \lim_{\varepsilon\to 0} \frac{\Big((r+\varepsilon)^2 - r^2\Big)\left(x + \frac{1}{2}\sin 2x\right)}{\Big((r+\varepsilon)^2 - r^2\Big)\left(\frac{\pi}{8} + \frac{\sqrt{2}}{4}\right)} \\
&= \frac{8}{\pi + 2\sqrt{2}}\left(x + \frac{\sin 2x}{2}\right). \qquad (5.5)
\end{aligned}$$

So in particular, conditional on $\mathcal{F}_i$ and $G_i^\star$, α_{i+1} is independent of R_{i+1}. We will write α for the 'ideal' angle distribution given by (5.5), and enforce that $\mathcal{R}$ and α be independent.

Geometric and combinatorial parameters

In this section we will build the elements required to bound the algorithmic complexity of the Cone-Walk algorithm. We begin by bounding the number of *steps* (or equivalently, the number of stoppers) required by the walk process to reach the destination. We will then bound the number of vertices *visited* by the walk process, recalling that this will involve bounding the number of *intermediary vertices* within the discs $\mathrm{Disc}(Z_i,q,R_i)$ at each step. The final part of the proof will be to bound the number of vertices *accessed* by the Cone-Walk algorithm when constructing the sequence of stoppers and intermediary vertices. The vertices *accessed* will include all of the vertices visited, and their 1-hop neighbourhood.

The maximum number of vertices accessed during a step

At each step during a walk, we do not a priori access a bounded number of sites when performing a search for the next stopper. Such a bound is important to limit the number of neighbourhoods that may be accessed during one step, since we note that the maximum number of vertices accessed during one step explicitly provides an upper bound on the number of neighbourhoods accessed. A easy bound of $\log^{1+\varepsilon} n$, for any $\varepsilon > 0$ may be obtained when considering pairs of start and destination points at least $\sqrt{\log n}$ away from the boundary of ∂D. However, we opt to explicitly take care of boundary effects, giving us a slightly weaker bound that can be applied everywhere.

More detail here?

Proposition 5.12. *Let M_{max} be the maximum number of vertices accessed during any step in any walk. Then*

$$\mathbb{P}\left(M_{max} \geq \omega_n^{3+\varepsilon}\right) \leq 2\exp\left(-\omega_n^{1+\varepsilon/4}\right).$$

In the following, we note that $M_{\max}$ is bounded by $\tau_{\max} \cdot \Delta_{\mathbf{X}}$, where $\Delta_{\mathbf{X}}$ gives the maximum degree of any vertex contained within $\mathrm{Del}(\mathbf{X})$ and $\tau_{\max}$ is the maximum number sites contained within any step in any instance of Cone Walk. We thus focus on bounding $\tau_{\max}$, and our result will follow directly from the proof of Theorem 4.1 in Chapter 4.

Lemma 5.13.

$$\mathbb{P}\left(\tau_{max} > \omega_n^{1+\varepsilon}\right) \leq \exp\left(-\omega_n^{1+\varepsilon/3}\right).$$

Proof. Let A be the event that the maximum disc radius for any step in any walk is bounded by $\frac{1}{2}\omega_n^{1/2+\varepsilon}$ and let B be the event that every ball $b(x, \frac{1}{2}\omega_n^{1/2+\varepsilon})$ contains fewer than $\omega_n^{1+2\varepsilon}$ points of $\mathbf{X}$, for $x \in D$. We have, for n large enough,

$$\begin{aligned}\mathbb{P}(\tau_{\max} > \omega_n^{1+2\varepsilon}) &\leq \mathbb{P}(\tau_{max} > \omega_n^{1+2\varepsilon} \mid A \cap B) + \mathbb{P}(A^c) + \mathbb{P}(B^c)\\ &\leq \exp\left(-\omega_n^{1+\varepsilon}\right) + \exp\left(-\omega_n^{1+\varepsilon}\right).\end{aligned}$$

Note that the bound on $\mathbb{P}(A^c)$ is implied by Lemma 5.11 since a large disc implicitly has a large empty cone. For the bound on $\mathbb{P}(B^c)$, we imagine splitting D into a uniform grid with squares of side $\frac{1}{2}\omega_n^{1+\varepsilon}$. The proof follows by noting that every ball of radius $\frac{1}{2}\omega_n^{1+\varepsilon}$ is contained in a group of at most four adjacent squares, each of which must contain at least $\frac{1}{4}\omega_n^{1+2\varepsilon}$ sites. We then use the fact that $n\,\mathbb{P}(\mathrm{Po}(\frac{1}{4}\omega_n^{1+\varepsilon}) \geq \frac{1}{4}\omega_n^{1+2\varepsilon}) \leq \exp\left(-\omega_n^{1+\varepsilon}\right)$ for n large enough. We omit the details. □

The number of steps in the walk

Recall that a new step is defined each time a new stopper is visited. We will start with a first crude estimate for the decrease in distance after a given number of steps. Note that $L_i = \|Z_i q\|$ and α_i denotes the angle between $Z_i Z_{i+1}$ and $Z_i q$. Simple geometry implies (see Figure 5.8):

$$\begin{aligned}L_i - R_i(1+\cos(2\alpha_i)) \leq L_{i+1} &= \sqrt{(L_i - R_i(1+\cos(2\alpha_i)))^2 + R_i^2\sin^2(2\alpha_i)}\\ &\leq L_i - R_i(1+\cos(2\alpha_i)) + 2\frac{R_i^2}{L_i},\end{aligned} \tag{5.6}$$

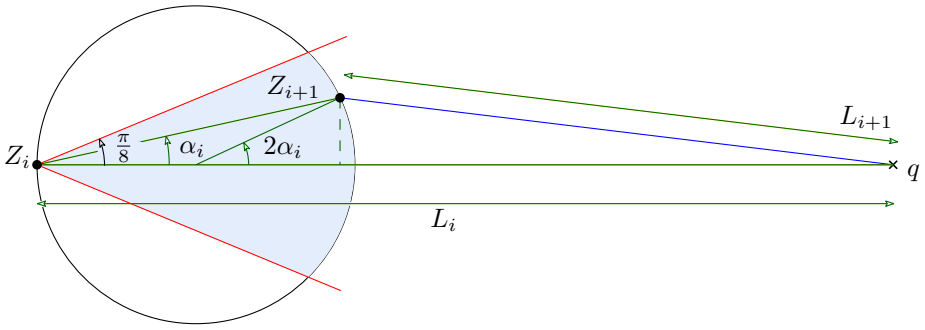


Figure 5.8 – Computing distance progress at step i.

since $\sqrt{1-x} \leq 1 - x/2$ for any $x \in [0,1]$. As a consequence

$$L_0 - \sum_{j=0}^{i-1} R_i(1+\cos(2\alpha_i)) \leq \quad L_i \quad \leq L_0 - \sum_{j=0}^{i-1} R_i(1+\cos(2\alpha_i)) + \frac{2}{\omega_n} \cdot \sum_{j=0}^{i-1} R_i^2. \tag{5.7}$$

In particular, since $\omega_n \to \infty$, after i steps, the expected distance $\mathbb{E}[L_i]$ to the aim q should not be far from $L_0 - i\,\mathbb{E}[\mathcal{R}(1+\cos(2\alpha))]$. Furthermore, conditional on G_i, and for i such that $L_i \geq \omega_n$, the summands involved in Equation (5.7) are independent, bounded by $2\omega_n$ and have bounded variance, so that the sum should be highly concentrated about its expected value [21, 51, 88]. In other words, one expects that for i much larger than $L_0/\,\mathbb{E}[\mathcal{R}(1+\cos 2\alpha)]$, it should be the case that $L_i \leq \omega_n$ with fairly high probability. Making this formal constitutes the backbone of our proof.

Lemma 5.14. *Let $z \in D$, suppose that $\ell \geq 1$ is such that $L_0 = \|zq\| \geq (\ell+1)\omega_n$. Consider* $\mathrm{Del}(\mathbf{X}_z)$. *There exists a constant $\eta > 0$ such that*

$$\mathbb{P}\left(L_0 - L_\ell \leq \ell\ \mathbb{E}[\mathcal{R}]/2\right) \leq \exp\left(-\eta\ell/\omega_n\right) + \exp\left(-\omega_n^{3/2}\right). \tag{5.8}$$

Proof. We use the crude bounds $R_i \leq L_i - L_{i+1} \leq 2R_i$ (see Figure 5.8). It follows that

$$\begin{aligned}\mathbb{P}(L_0 - L_\ell \leq \ell\ \mathbb{E}[\mathcal{R}]/2) &\leq \mathbb{P}(L_0 - L_\ell \leq \ell\ \mathbb{E}[\mathcal{R}]/2 \,|\, G_\ell) + \mathbb{P}(G_\ell^c) \\ &\leq \mathbb{P}\left(\sum_{j=0}^{\ell-1} R_j \leq \frac{\ell}{2}\,\mathbb{E}[\mathcal{R}] \,\middle|\, G_\ell\right) + \exp\left(-\omega_n^{3/2}\right),\end{aligned}$$

by (5.4), since the constraint on ℓ imposes that $\ell = O(\sqrt{n})$. Now, since $L_0 \geq (\ell+1)\omega_n$ and $\xi > 2$, on the event G_ℓ, we have $L_i \geq \omega_n$ for $0 \leq i \leq \ell$ so that $G_\ell^\star$ occurs: conditional on G_ℓ, the search cones do not intersect and the random variables R_j, $0 \leq j \leq \ell$ are independent and identically distributed (see Lemma 5.8). Furthermore, we have

$$\begin{aligned}\mathbb{E}[R_j \mid G_\ell] &= \int_0^\infty \mathbb{P}\left(R_j \geq x \mid G_\ell\right) dx \\ &\geq \int_0^{\omega_n/\xi} \exp\left(-Ax^2\right) dx \\ &\geq \mathbb{E}[\mathcal{R}] - \exp\left(-\omega_n^{3/2}\right),\end{aligned}$$

for all n large enough. It follows that for all n large enough, by Theorem 2.7 of [88, p. 203]

$$\begin{aligned}\mathbb{P}\left(\sum_{j=0}^{\ell-1} R_j \le \frac{\ell}{2}\,\mathbb{E}[\mathcal{R}]\,\middle|\,G_\ell\right) &\le \mathbb{P}\left(\sum_{j=0}^{\ell-1}(R_j - \mathbb{E}[R_j \mid G_\ell]) \le -\frac{\ell}{3}\,\mathbb{E}[R_0 \mid G_\ell]\,\middle|\,G_\ell\right)\\ &\le \exp\left(-\frac{t^2}{2\ell\ \mathrm{Var}(R_0 \mid G_\ell) + 2t\omega_n/3}\right) \qquad t = \ell\ \mathbb{E}[R_0 \mid G_\ell]/3\\ &\le \exp\left(-\eta\ell/\omega_n\right),\end{aligned}$$

for some constant $\eta > 0$ independent of ℓ and n. □

The rough estimate in Lemma 5.14 may be significantly strengthened, and the very representation in (5.7) yields a bound on the number of search cones or steps that are required to get within distance ω_n of the query point q. (If the starting site z satisfies $L_0 = \|zq\| \le \omega_n$, then this phase does not contain any step.)

Proposition 5.15. *Let $z \in D_n$, and let $\kappa(\omega_n)$ denote the number of steps of the walk to reach a site which is within distance ω_n of q in $\mathbf{X}_z \cup \{q\}$ when starting from the site $z \in \mathbf{X}_z$ at distance $L_0 = \|zq\| \ge \omega_n$. Then*

$$\mathbb{P}\left(\left|\kappa(\omega_n) - \frac{L_0}{\mathbb{E}[\mathcal{R}(1+\cos 2\alpha)]}\right| \ge 2\omega_n^2\sqrt{2L_0} + \omega_n\right) \le 4\exp\left(-\omega_n^{3/2}\right). \tag{5.9}$$

Proof. We now make formal the intuition that follow Equation (5.7). We start with the upper bound. For any integer $k \ge 0$, we have

$$\begin{aligned}\mathbb{P}\left(\kappa(\omega_n) \ge k\right) &= \mathbb{P}\left(L_k \ge \omega_n\right)\\ &\le \mathbb{P}\left(L_k \ge \omega_n \mid G_k\right) + \mathbb{P}\left(G_k^c\right),\end{aligned}$$

and since the second term is bounded in (5.4), it now suffices to bound the first one. However, given G_k and $L_k \ge \omega_n$, the random variables (R_i, α_i), $i = 1, \ldots, k$ are independent and identically distributed. The only effect of this conditioning is that R_i is distributed as $\mathcal{R}$ conditioned on $\mathcal{R} < \omega_n/\xi$.

Write $X_i = R_i(1 + \cos 2\alpha_i) - 2R_i^2/\omega_n$, and note that $X_i \ge 0$ if $R_i \le \omega/\xi$. Then, from (5.7), we have

$$\mathbb{P}\left(L_k \ge \omega_n \mid G_k\right) \le \mathbb{P}\left(\sum_{i=0}^{k-1} X_i \le L_0 - \omega_n\,\middle|\,G_k, L_k \ge \omega_n\right).$$

Conditional on $G_k^\star = G_k \cap \{L_k \ge \omega_n\}$, the random variables X_i are independent, $0 \le X_i \le 2R_i \le \omega_n$. Furthermore, since X_i has Gaussian tails, its variance (conditional on G_k) is bounded by a constant independent of i and n. Choosing $k_0 = \lceil (L_0 + t)/\,\mathbb{E}[X_0 \mid G_0^\star]\rceil$, for some $t < L_0$ to be chosen later, and using the Bernstein-type inequality in Theorem 2.7 of [88, p. 203], we obtain

$$\begin{aligned}\mathbb{P}(L_{k_0} \ge \omega_n \mid G_{k_0}) &\le \mathbb{P}\left(\sum_{i=0}^{k_0-1}(X_i - \mathbb{E}[X_i \mid G_{k_0}]) \le -t\,\middle|\,G_{k_0}^\star\right)\\ &\le \exp\left(-\frac{t^2}{2k_0\,\mathrm{Var}(X_0 \mid G_0^\star) + 2\omega_n t/3}\right).\end{aligned}$$

In particular, for $t = \omega_n^3 \sqrt{L_0}$, we have for all n large enough $\mathbb{P}(L_{k_0} \geq \omega_n \mid G_{k_0}) \leq \exp(-\omega_n^2)$, since $L_0 \geq \omega_n$.

A matching lower bound on $\kappa(\omega_n)$ may be obtained similarly, using the lower bound on L_{i+1} in Equation (5.6) and following the approach we used to devise the upper bound with $X_i' = R_i(1 + \cos 2\alpha_i)$ (we omit the details). It follows that, for $k_1 = \lfloor (L_0 + t)/\,\mathbb{E}[X_0' \mid G_0^\star] \rfloor$, we have $\mathbb{P}(L_{k_1} \leq \omega_n \mid G_{k_1}) \leq \exp(-\omega_n^2)$.

To complete the proof, it suffices to estimate the difference between k_0 and k_1. We have

$$\begin{aligned}\mathbb{E}[X_0 \mid G_0^\star] &= \mathbb{E}[R_0(1 + \cos 2\alpha_0) \mid G_0^\star] - \frac{2\,\mathbb{E}[R_0^2 \mid G_0^\star]}{\omega_n} \\ &= \mathbb{E}[\mathcal{R}(1 + \cos 2\alpha)] + O(1/\omega_n),\end{aligned}$$

and similarly, $\mathbb{E}[X_0' \mid G_0^\star] = \mathbb{E}[\mathcal{R}(1 + \cos 2\alpha)]$. It follows that $|k_1 - k_0| = O(L_0/\omega_n)$, which is not strong enough to prove the claim. So we need to strengthen the upper bound on the second sum in the right-hand side of (5.7). We quickly sketch how to obtain the required estimate. The idea is to use a dyadic argument to decompose $\kappa(\omega_n)$ into the number of steps to reach $L_0/2^j$, for $j \geq 1$, until one gets to ω_n for $j = j_0 := \lceil \log_2(L_0/\omega_n) \rceil$. For the steps i which are taken from Z_i with $L_i/L_0 \in (2^{-j}, 2^{-j+1}]$, we use the improved bound

$$L_{i+1} \leq L_i - R_i(1 + \cos 2\alpha) + \frac{2}{L_0 2^{-j}} R_i^2. \tag{5.10}$$

Then write

$$\kappa(\omega_n) = \sum_{j=1}^{j_0} \Big[\kappa(L_0/2^j) - \kappa(L_0/2^{j-1})\Big], \tag{5.11}$$

and observe that the j-th summand is stochastically dominated by $\kappa(L_0'/2)$ where $L_0' = L_0/2^{j-1}$. For each j, we define $k_0(j) = \lceil (L_0/2^j + t_j)/\,\mathbb{E}[X_0 \mid G_0^\star] \rceil$ where $t_j := \omega_n^2 \sqrt{L_0/2^j}$ and note that

$$\begin{aligned}\sum_{j=1}^{j_0} k_0(j) &\leq \frac{1}{\mathbb{E}[X_0 \mid G_0^\star]} \sum_{j=1}^{j_0} (L_0/2^j + t_j) + \lceil \log_2(L_0/\omega_n) \rceil \\ &\leq \frac{L_0}{\mathbb{E}[X_0 \mid G_0^\star]} + 2\omega_n^2 \sqrt{2L_0} + \omega_n,\end{aligned}$$

for $\omega_n \geq \log n$, since $\pi L_0^2 \leq n$. In other words, if $\kappa(L_0/2^j) - \kappa(L_0/2^{j-1}) \leq k_0(j)$ for every j, then $\kappa(\omega) \leq L_0/\,\mathbb{E}[X_0 \mid G_0^\star] + 2\omega_n^2 \sqrt{2L_0} + \omega_n$. The claim follows easily by using the union bound, where in each stretch $[L_0/2^j, L_0/2^{j-1})$ we bound the number of steps using the previous arguments. □

Corollary 5.16. *Let $z \in D_n$, and let κ denote the number of steps of the walk to reach the objective q in $\mathbf{X}_z \cup \{q\}$ when starting from the site $z \in \mathbf{X}_z$ at distance $L_0 = \|zq\|$. Then*

$$\mathbb{P}\left(\kappa > \frac{L_0}{\mathbb{E}[\mathcal{R}(1 + \cos 2\alpha)]} + 2\omega_n^2 \sqrt{2L_0} + \omega_n^3\right) \leq 5 \exp\left(-\omega_n^{3/2}\right). \tag{5.12}$$

Proof. It suffices to bound the number of steps i such that $L_i < \omega_n$. Since L_i is decreasing, the walk only stops at most once at any given site, and the number of steps i with $L_i \le \omega_n$ is at most the number of sites lying within distance ω_n of q. Recalling that $\mathrm{Po}(x)$ denotes a Poisson random variable with mean x. We have [51]

$$\begin{aligned}\mathbb{P}\left(\#\{i < \kappa : L_i \le \omega_n\} \ge 2\pi\omega_n^2\right) &\le \mathbb{P}\left(\mathrm{Po}(\pi\omega_n^2) \ge 2\pi\omega_n^2\right)\\ &\le \exp\left(-\pi\omega_n^2/3\right).\end{aligned}$$

The claim then follows easily from the upper bound in Proposition 5.15. □

The number of vertices in the discs

We now bound the total number of vertices visited, which we recall is exactly the the number of points in $\mathbf{X}_n$ falling within union of all of the discs in the walk. Proposition 5.15 will be the key to analysing the path constructed by the walk: representations based on sums of random variables similar to the one in (5.6) may be obtained to upper bound the number of steps and intermediate steps visited by the walk (which is an upper bound on the vertices visited by the path), and also the sum of the length of the edges.

Proposition 5.17. *Let $K = K(z)$ be the number of vertices visited by the walk starting from a given site z with $L_0 = \|zq\|$. Then, for all n large enough,*

$$\mathbb{P}\left(K \ge \frac{L_0}{\mathbb{E}[\mathcal{R}(1+\cos 2\alpha)]} \cdot \frac{\pi - A}{A} + \sqrt{L_0}\omega_n^4 + \omega_n^3\right) \le 7\exp\left(-\omega_n^{3/2}\right). \tag{5.13}$$

Proof. There are two contributions to $K - \kappa(\omega_n)$: first the number of intermediate steps which lie at distance greater than ω_n from q, and all the sites which are visited and lie within distance ω_n from q. Let $K = K_1 + K_2$ where K_1 and K_2 denote these two contributions, respectively. By the proof of Corollary 5.16, we have

$$\mathbb{P}\left(K_2 \ge 2\pi\omega_n^2\right) \le \exp\left(-\pi\omega_n^2/3\right). \tag{5.14}$$

To bound K_1, observe that the monotonicity of L_i implies that K_1 counts precisely the number of intermediate steps before reaching the disc of radius ω_n about q. Observe that if $L_0 < \omega_n$, $K_1 = 0$, so we may assume that $L_0 \ge \omega_n$. Recall that τ_i denotes the number of *intermediate* points at the i-th step. Note that the intermediate points counted by τ_i all lie in $\mathrm{Disc}(Z_i, q, R_i) \setminus \mathrm{Cone}(Z_i, q, R_i)$, and given the radius R_i, τ_i is stochastically bounded by a Poisson random variable with mean $(\pi - A)R_i^2$. Furthermore, on the event $G_{\kappa(\omega_n)}$, the random variables $R_i, i = 0, \dots, \kappa(\omega_n)$ are independent. Also, by Lemma 5.9 the regions $\mathrm{Disc}(Z_i, q, R_i) \setminus \mathrm{Cone}(Z_i, q, R_i)$, $i \ge 0$, are disjoint so that the random variables τ_i, $i = 0, \dots, \kappa$ are independent given R_i, $i = 0, \dots, \kappa$.

Let $\widetilde{\mathcal{R}}_i$, $i \ge 0$, be a sequence of iid random variables distributed like $\mathcal{R}$ conditioned on $\mathcal{R} \le \omega_n/\xi$ and given this sequence, let $\tilde{\tau}_i$, $i \ge 0$, be independent distributed like $\mathrm{Po}((\pi - A)\widetilde{\mathcal{R}}_i)$. As a consequence of the previous arguments, for $k = k_0 + 2\omega_n^2\sqrt{2L_0} + \omega_n$ with

$k_0 = \lceil L_0 / \mathbb{E}[\mathcal{R}(1+\cos 2\alpha)] \rceil$, we have

$$\begin{aligned}\mathbb{P}(K_1 \geq \ell) &\leq \mathbb{P}\left(\sum_{i=0}^{(k-1)\wedge\kappa(\omega_n)} \tau_i \geq \ell\right) + \mathbb{P}\left(\kappa(\omega_n) \geq k_0 + 2\omega_n^2\sqrt{2L_0} + \omega_n\right) \\ &\leq \mathbb{P}\left(\sum_{i=0}^{k-1} \tilde{\tau}_i \geq \ell\right) + \mathbb{P}(G_k^c) + \mathbb{P}\left(\kappa(\omega_n) \geq k_0 + 2\omega_n^2\sqrt{2L_0} + \omega_n\right) \\ &\leq \mathbb{P}\left(\sum_{i=0}^{k-1} \tilde{\tau}_i \geq \ell\right) + 5\exp\left(-\omega_n^{3/2}\right), \end{aligned} \tag{5.15}$$

by (5.4) and Proposition 5.15.

We now bound the first term in (5.15). Note that $i \geq 0$, we have

$$\mathbb{E}[\tilde{\tau}_i] = (\pi - A)\,\mathbb{E}[\widetilde{\mathcal{R}}_i^2] \leq (\pi - A)\,\mathbb{E}[\mathcal{R}^2] = \frac{\pi - A}{A} =: \gamma, \tag{5.16}$$

and we expect that $\sum_{i=0}^{k-1} \tilde{\tau}_i$ should not exceed its expected value, $k\gamma$ by much. Write $\ell = k\gamma + t$, for some t to be chosen later. For the sum to be exceptionally large either the radii of the search discs are large, or the discs are not too large but the number of points are:

$$\begin{aligned}\mathbb{P}\left(\sum_{i=0}^{k-1} \tilde{\tau}_i \geq k\gamma + t\right) &= \mathbb{P}\left(\mathrm{Po}\left((\pi - A)\sum_{i=0}^{k-1} \widetilde{\mathcal{R}}_i^2\right) \geq k\gamma + t\right) \\ &\leq \mathbb{P}\left(\mathrm{Po}\left(k\gamma + \frac{t}{2}\right) \geq k\gamma + t\right) + \mathbb{P}\left(\sum_{i=0}^{k-1} \widetilde{\mathcal{R}}_i^2 \geq \frac{k\gamma + t/2}{\pi - A}\right). \end{aligned} \tag{5.17}$$

The first term simply involves tail bounds for Poisson random variables. For $t = \sqrt{L_0}\omega_n^4$, we have

$$\begin{aligned}\mathbb{P}\left(\mathrm{Po}\left(k\gamma + \frac{t}{2}\right) \geq k\gamma + t\right) &\leq \exp\left(-\frac{(t/2)^2}{3(k\gamma + t/2)}\right) \\ &\leq \exp\left(-\omega_n^3\right), \end{aligned}$$

for n large (recall that we can assume here that $L_0 \geq \omega_n$.) The second term in (5.17) is bounded using the same technique as in the proof of Proposition 5.15 above. Since we have $0 \leq \widetilde{\mathcal{R}}_i^2 \leq \omega_n^2$ and $\mathbb{E}[\widetilde{\mathcal{R}}_i^2] \leq 1/A$, we obtain for some positive constant c,

$$\mathbb{P}\left(\sum_{i=0}^{k-1} \widetilde{\mathcal{R}}_i^2 \geq \frac{k\gamma + t/2}{\pi - A}\right) \leq \mathbb{P}(G_k) + \exp\left(-8\frac{k}{A^2\omega_n^4}\right) \leq \exp\left(-\frac{ct^2}{k\,\mathrm{Var}(\mathcal{R}) + \omega_n^2 t}\right).$$

Recalling that $L_0 \geq \omega_n$, yields

$$\mathbb{P}\left(\sum_{i=0}^{k-1} \tilde{\tau}_i \geq k\gamma + \omega_n^4\sqrt{L_0}\right) \leq \exp\left(-\omega_n^2\right), \tag{5.18}$$

for all n large enough, which together with (5.15) proves that

$$\mathbb{P}(K_1 \geq k\gamma + \omega_n^4\sqrt{L_0}) \leq 6\exp(-\omega_n^{3/2}). \tag{5.19}$$

Using (5.14) readily yields the claim. □

The length of Simple-Path

When using Competitive-Path, the path length is deterministically bounded by the length of the walk. However, for Simple-Path, the path length is dependent on the configuration of the points inside the discs. We will show that with strong probability and as long as the walk is sufficiently long, the path length given by Competitive-Path is no better in an asymptotic sense than that given by Simple-Path.

Proposition 5.18. *For $z \in D$, let $\Lambda = \Lambda(z)$ be the sum of the lengths of the edges of* $\mathrm{Del}(\mathbf{X}_z)$ *used by* Simple-Path *given a walk with objective q and starting from z such that $L_0 = \|zq\|$. Then,*

$$\mathbb{P}\left(\Lambda \geq cL_0 + (3\sqrt{L_0}+1)\omega_n^4\right) \leq 8\exp\left(-\omega_n^{3/2}\right) \qquad \text{where} \qquad c := \frac{22\pi - 4\sqrt{2}}{2 + 3\pi + 8\sqrt{2}}. \tag{5.20}$$

Proof. Write λ_i for the sum of the lengths of the edges used by the walk to go from Z_i to Z_{i+1}. So $\Lambda = \sum_{i=0}^{\kappa-1} \lambda_i$. Our bound here is very crude: all the intermediate points remain in $\mathrm{Disc}(Z_i, q, R_i)$, and given R_i, we have $\lambda_i \leq (1+\tau_i)\cdot 2R_i$. Again, on G_k the cones do not intersect provided that $L_k \geq \omega_n$, and by Lemma 5.9 the random variables λ_i, $0 \leq i < k$ are independent. We use once again the method of bounded variances (Theorem 2.7 of [88]).

We decompose the sum into the contribution of the steps before $\kappa(\omega_n)$ and the ones after:

$$\mathbb{P}\left(\Lambda \geq x + t\right) \leq \mathbb{P}\left(\sum_{i=0}^{(k-1)\wedge\kappa(\omega_n)} \lambda_i \geq x\right) + \mathbb{P}\left(\kappa(\omega_n) \geq k\right) + \mathbb{P}\left(\sum_{i=\kappa(\omega_n)}^{\kappa-1} \lambda_i \geq t\right). \tag{5.21}$$

For $i \geq \kappa(\omega_n)$, $\mathrm{Disc}(Z_i, q, R_i)$ is contained in $b(q, \omega_n)$, the disc of radius ω_n around q, and the contribution of the steps $i \geq \kappa(\omega_n)$ is at most $2\omega_n|\Phi \cap b(q,\omega_n)|$. In particular

$$\begin{aligned}\mathbb{P}\left(\sum_{i=\kappa(\omega_n)}^{\kappa-1} \lambda_i \geq t\right) &\leq \mathbb{P}\left(2\omega_n \,\mathrm{Po}(\pi\omega_n^2) \geq t\right) \\ &\leq \exp\left(-\omega_n^{3/2}\right),\end{aligned}$$

for all n large enough provided that $t \geq 4\pi\omega_n^3$. To make sure that the second contribution in (5.21) is also small, we rely on Proposition 5.15 and choose $k = \lceil L_0 / \mathbb{E}[\mathcal{R}(1+\cos 2\alpha)] + 2\omega_n^2\sqrt{2L_0} + \omega_n\rceil$ so that $\mathbb{P}(\kappa(\omega_n) \geq k) \leq 4\exp(-\omega_n^{3/2})$.

Finally, to deal with the first term in (5.21), we note that on G_k, the random variables λ_i, $i = 0, \ldots, k+1$ are independent given R_i, $i = 0, \ldots, k+1$. Let $\widetilde{\mathcal{R}}_i$, $i = 0, \ldots, k-1$ be iid copies of $\mathcal{R}$ conditioned on $\mathcal{R} \leq \omega_n/\xi$; then let $\tilde{\tau}_i$ be independent given R_i, $i = 0, \ldots, k+1$, and such that $\tilde{\tau}_i = \mathrm{Po}((\pi - A)\widetilde{\mathcal{R}}_i)$; finally, let $\tilde{\lambda}_i = 2\mathcal{R}_i(1+\tilde{\tau}_i)$. We choose $x = k\,\mathbb{E}[\tilde{\lambda}_0] + y$ with $y = \sqrt{L_0}\omega_n^4$. Using arguments similar to the ones we have used in the proofs of Propositions 5.15 and 5.17, we obtain

$$\begin{aligned}\mathbb{P}\left(\sum_{i=0}^{(k-1)\wedge\kappa(\omega_n)} \lambda_i \geq x\right) &\leq \mathbb{P}\left(\sum_{i=0}^{k-1} \tilde{\lambda}_i \geq x\right) + \mathbb{P}(G_k^c) \qquad (5.22)\\ &\leq \exp\left(-\frac{y^2}{2k\,\mathrm{Var}(\tilde{\lambda}_0) + 2\omega_n^2 y/3}\right) + \mathbb{P}\left(\exists i < k : \tilde{\lambda}_i \geq \omega_n^2\right) + \mathbb{P}\left(G_k^c\right).\end{aligned}$$

To bound the second term in the right-hand side above, observe that for $i < k$, we have, for any $x > 0$,

$$\begin{aligned}
\mathbb{P}\left(\tilde{\lambda}_i \geq x^2\right) &\leq \mathbb{P}\left((1+\tilde{\tau}_i)2\widetilde{\mathcal{R}}_i \geq x^2\right) \\
&\leq \mathbb{P}\left((1+\tilde{\tau}_i)2\mathcal{R}_i \geq x^2 \,\middle|\, 2\widetilde{\mathcal{R}}_i \leq x\right) + \mathbb{P}\left(2\widetilde{\mathcal{R}}_i \geq x\right) \\
&\leq \mathbb{P}\left(1+\tilde{\tau}_i \geq x \,\middle|\, 2\widetilde{\mathcal{R}}_i \leq x\right) + \mathbb{P}\left(2\widetilde{\mathcal{R}}_i \geq x\right) \\
&\leq \mathbb{P}\left(\mathrm{Po}((\pi - A)x^2/4) \geq x - 1\right) + \exp\left(-Ax^2/4\right) \\
&\leq 2\exp\left(-\eta x^2\right),
\end{aligned}$$

for some constant $\eta > 0$ and all x large enough. It follows immediately that $\mathrm{Var}(\tilde{\lambda}_0) < \infty$ and that, n large enough,

$$\begin{aligned}
\mathbb{P}\left(\exists i < k : \tilde{\lambda}_i \geq \omega_n^2\right) &\leq k\exp\left(-\eta\omega_n^2\right) \\
&\leq \exp\left(-\omega_n^{3/2}\right). \qquad (5.23)
\end{aligned}$$

Going back to (5.22), we obtain

$$\mathbb{P}\left(\sum_{i=0}^{(k-1)\wedge\kappa(\omega_n)} \lambda_i \geq x\right) \leq 3\exp\left(-\omega_n^{3/2}\right),$$

since here, we can assume that $L_0 \geq \omega_n$ (if this is not the case, the points outside of the disc of radius ω_n centred at q do not contribute). Putting the bounds together yields

$$\mathbb{P}\left(\Lambda \geq x + t\right) \leq 8\exp\left(-\omega_n^{3/2}\right), \qquad (5.24)$$

and the claim follows by observing that for

$$c := \frac{\mathbb{E}[2(1+\mathrm{Po}((\pi - A)\mathcal{R}^2))\mathcal{R}]}{\mathbb{E}[\mathcal{R}(1+\cos 2\alpha)]} = \frac{2\,\mathbb{E}[\mathcal{R}] + \mathbb{E}[2(\pi - A)\mathcal{R}^3]}{\mathbb{E}[\mathcal{R}(1+\cos 2\alpha)]}, \qquad (5.25)$$

it is the case that $cL_0 + (3\sqrt{L_0} + 1)\omega_n^4 \geq x + t$ for all n large enough. Simple integration using the distributions of $\mathcal{R}$ and α then yields the expression in (5.20). □

The number of sites accessed

In this section, we bound the total number of sites accessed by the cone-walk algorithm, counted with multiplicity. We note a point is accessed at step i if it is the endpoint of an edge whose other end lies inside the disc D_i. A given point may be accessed more than once, but via different edges, so we will bound the number of edges such that for some i, one end point lies inside D_i and the other outside; we call these *crossing edges*. (Note that the number of crossing edges does not quite bound the complexity of the algorithm for the complexity of a given step is not linear in the number of accessed points; however it gives very good information on the amount of data the algorithm needs to process.)

Proposition 5.19. *Let $A = A(z)$ be the number of sites in $\mathbf{X}$ accessed by the Cone Walk algorithm (with multiplicity) when walking towards q from z. Then there exists a constant $c > 0$ such that*

$$\mathbb{P}\left(A(z) > c\,L_0 + 4\left(\sqrt{L_0}+1\right)\omega_n^6\right) \leq 3\exp\left(-\omega_n^{5/4}\right). \tag{5.26}$$

For n sufficiently large.

Proof. In order to bound the number of such edges, we adapt the concept of the *border point* introduced by Bose and Devroye [15] to bound the stabbing number of a random Delaunay triangulation. For $B \subseteq D_n$ and a point $x \in D_n$, let $\|xB\| := \inf\{\|xy\| : y \in B\}$ denote the distance from x to B.

We consider the walk from z to q in D_n, letting

$$W = \bigcup_{i=1}^{\kappa} D_i \qquad \text{and} \qquad W^{\odot} := B(\mathbf{w}, 2\max\{\|zq\|, \omega_n^5\}), \tag{5.27}$$

where $\mathbf{w}$ denotes the centroid of the segment zq and we recall that $B(\mathbf{x},r)$ denotes the closed ball centred at $\mathbf{x}$ of radius r. Then, for $x \in W^{\odot}$, let C be the disc centred at x and with radius $\min\{\|xW\|, \|x\partial W^{\odot}\|\}$. Partition the disc C into 8 isometric cone-shaped sectors (such that one of the separation lines is vertical, say) truncated to a radius of $\sqrt{3}/2$ times that of the outer disc (see Figure 5.9). We say that x is a border point if one of the 24 cones does not contain any points in $\mathbf{X}$. If $x \in W^{\odot}$ is a border point then there is no Delaunay edge between x and a point lying outside C, since a circle trough x and $y \notin C \subset W^{\odot} \setminus W$ must entirely enclose at least one sector of C (see dotted circle in Figure 5.9). Thus if x has a Delaunay edge with extremity in W, then x must be a border point. The connection between border points and the number of crossing edges can be made via Euler's relation, since it follows that a crossing edge is an edge of the (planar) subgraph of $\mathrm{Del}(\mathbf{X})$ induced by the points which either lie inside W, are border points, or lie outside of $W^{\odot}$ and have a neighbour in W. Let B_W denote of set of border points, E_W the collection of crossing edges, and Y_W the collection of points lying outside of $W^{\odot}$ and having a Delaunay neighbour within W. Then

$$A(z) \leq |E_W| \leq 3(|W \cap \mathbf{X}| + |B_W| + |Y_W|). \tag{5.28}$$

Proposition 5.17 bounds $|W \cap \mathbf{X}|$, as this is exactly the set of *visited vertices*. Lemmas 5.20 and 5.21 bounding $|B_W|$ and $|Y_W|$ complete the proof. □

Lemma 5.20. *For all n large enough, we have*

$$\mathbb{P}\left(|Y_W| \geq 10\max\{L_0, \omega_n^5\}\right) \leq 2\exp\left(-\omega_n^{5/4}\right). \tag{5.29}$$

Proof. Heuristically, our proof will follow from the fact that, with high probability, a Delaunay edge away from the boundary of the domain is not long enough to span the distance between a point within the walk, and a point outside of $W^{\odot}$. Unfortunately our proof is complicated by points on the walk which are very close to the boundary of the

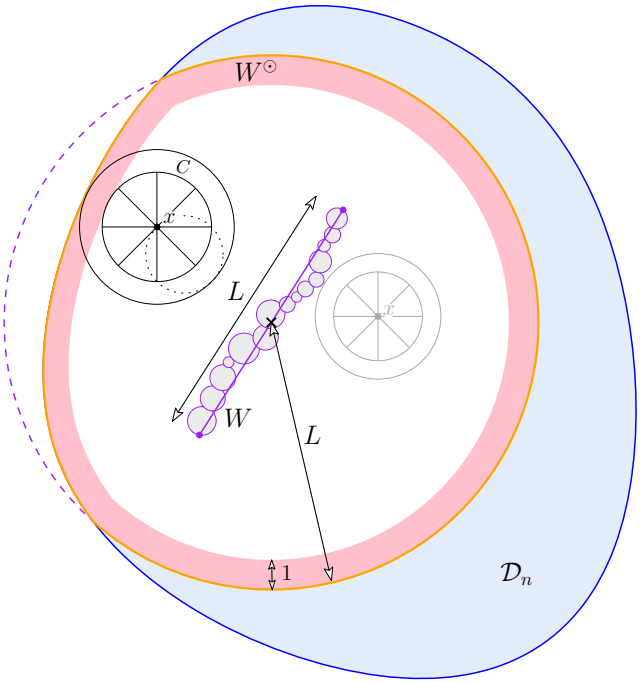


Figure 5.9 – For the proof of Proposition 5.19.

domain, since in this case, those points might have 'bad' edges which are long enough to escape $W^{\odot}$. To deal with this, we will take all points in the walk that are close to the border, and imagine that every Delaunay edge touching one of these points is such a 'bad' edge. The total number of these edges will be bounded by the maximum degree.

To begin, we give the first case. Consider an arbitrary point $x \in W \cap \mathbf{X}$ that is at least ω_n^{-3} away from the boundary of D_n. Suppose this point has a neighbour outside of $W^{\odot}$, then its circumcircle implicitly overlaps an unconditioned region of D_n with area at least $c\,\omega_n^{-3}\,\omega_n^5 = c\,\omega_n^2$ (for $c > 0$ a constant depending on the shape of the domain). The probability that this happens for x is thus at most $\exp(-\omega_n^2)$. Now note that there are at most $2n$ points in $\mathbf{X}$ with probability bounded by $\exp(-\omega_n^2)$ and at most $4n^2$ edges between points of $x \in W \cap \mathbf{X}$ and $x \in \{W^{\odot}\}^c \cap \mathbf{X}$. By the union bound, the probability that any such edge exists is at most

$$(4n^2)\exp\left(-c\,\omega_n^2\right) + \exp\left(-\omega_n^2\right) \leq \exp\left(-\omega_n^{3/2}\right). \tag{5.30}$$

For the second case, we count the number of points within ω_n^{-3} of the boundary of the domain. Using standard arguments, we have that there are no more than $10\max\{L, \omega_n^5\} \cdot \omega_n^{-3}$ such points, with probability at least $\exp(-\omega_n^2)$. Each of these has at most $\Delta_{\mathbf{X}}$ edges that could exit $W^{\odot}$, where $\Delta_{\mathbf{X}}$ is the maximum degree of any vertex in $\mathrm{Del}(\mathbf{X})$, which is bounded in Chapter 4. Thus, the number of such bad edges is at most $10\max\{L, \omega_n^5\}\omega_n^{-3} \cdot \omega_n^3$ with probability at least $\exp(-\omega_n^2) + \exp(-\omega_n^{5/4})$ □

Lemma 5.21. *For all n large enough, and universal constant $C > 0$,*

$$\mathbb{P}\left(|B_W| \geq C \max\{L_0, \omega_n^6\}\right) \leq 2\exp\left(-\omega_n^{3/2}\right). \tag{5.31}$$

Proof. Since $|B_W|$ is a sum of indicator random variables, it can be bounded using (a version of) Chernoff–Hoeffding's method. The only slight annoyance is that the indicators $\mathbb{1}_{x\in B_W}, x \in \mathbf{X}\cap W^{\odot}$ are not independent. Note however that $\mathbb{1}_{x\in B_W}$ and $\mathbb{1}_{y\in B_W}$ are only dependent if the discs used to define membership to B_W for x and y intersect. There is a priori no bound on the radius of these discs, and so we shall first discard the points $x \in \mathbf{X}$ lying far away from $\partial W^{\odot}$ and W. More precisely, let $B_W^{\star}$ denote the set of border points lying within distance ω_n of either W or $\partial W^{\odot}$, and $B_W^{\bullet} = B_W \setminus B_W^{\star}$. Observe now that we may bound $|B_W^{\bullet}|$ directly using Lemma 5.11, since a point is only a border point if one of its cones is empty, and each such empty cone contains a large empty circle. So for sufficiently large n,

$$\mathbb{P}\left(|B_W^{\bullet}| \neq 0\right) \leq \exp\left(-\omega_n^{3/2}\right). \tag{5.32}$$

Bounding $|B_W^{\star}|$ is now easy since the amount of dependence in the family $\mathbb{1}_{x\in B_W}, x \in \mathbf{X}\setminus W$ is controlled and we can use the inequality by Janson [51, 70]. We start by bounding the expected value $\mathbb{E}\,|B_W^{\star}|$. Note that for a single point $x \in \mathbf{X}_n$, by definition the disc used to define whether x is a border point does not intersect W and stays entirely within D, so

$$\begin{aligned}\mathbb{P}_x(x \in B_W^{\star}) = \mathbb{P}(x \in B_W^{\star}) &\leq 24\exp\left(-\frac{\pi}{32}\min\{\|xW\|, \|x\partial W^{\odot}\|\}^2\right)\\ &\leq 24\exp\left(-\frac{\pi}{32}\|xW\|^2\right) + 24\exp\left(-\frac{\pi}{32}\|x\partial W^{\odot}\|^2\right)\end{aligned} \tag{5.33}$$

and $\mathbf{X}_n$ is unconditioned in $D \setminus W$. Partition $D \setminus W$ into disjoint sets as follows:

$$D \setminus W = \bigcup_{i=0}^{\infty} U_i \tag{5.34}$$

where $U_i := \{x \in D \;:\; i \leq \|xW\| < i+1\}$. Similarly, the sets $U_i' := \{x \in W^{\odot} : i \leq \|x\partial W^{\odot}\| < i+1\}$ form a similar partition for $W^{\odot}$. Recalling that λ_2 denotes the 2-dimensional Lebesgue measure and using (5.33) above, we have

$$\begin{aligned}\mathbb{E}\,|B_W^{\star}| &= \mathbb{E}\left[\sum_{x\in\mathbf{X}} \mathbb{1}_{x\in W^{\odot}\setminus W}\;\mathbb{1}_{x\in B_W^{\star}}\right]\\ &= \int_{W^{\odot}\setminus W} \mathbb{P}_x(x \in B_W^{\star})\,\lambda_2(dx)\\ &\leq \sum_{i=0}^{\infty}\int_{U_i} 24\exp\left(-\pi i^2/32\right)\,\lambda_2(dx) + \sum_{i=0}^{\infty}\int_{U_i'} 24\exp\left(-\pi i^2/32\right)\,\lambda_2(dx)\\ &= 24\sum_{i=0}^{\infty}(\lambda_2(U_i) + \lambda_2(U_i'))\;\exp\left(-\pi i^2/32\right).\end{aligned}$$

We may now bound $\lambda_2(U_i)$ and $\lambda_2(U_i')$ as follows. Recall that W is a union of discs $W = \cup_i D_i$. We clearly have that

$$U_i \subseteq \bigcup_{j=0}^{\kappa-1} \left\{ x \in D \ : \ i \le \|xD_j\| < i+1 \right\}$$

Note that

$$\lambda_2\Big(\left\{x \in D \ : \ i \le \|xD_j\| < i+1\right\}\Big) \le \pi((R_j+i+1)^2 - (R_j+i)^2) \tag{5.35}$$

$$= \pi(2(R_j+i)+1). \tag{5.36}$$

So, assuming there are κ steps in the walk we get

$$\lambda_2(U_i) \le \sum_{j=0}^{\kappa-1} \pi(2(R_j+i)+1) = 2\pi \sum_{j=0}^{\kappa-1} R_j + \pi(i+1)\kappa.$$

Regarding $\lambda_2(U_i')$, note first that $W^{\odot}$ is convex for it is the intersection of two convex regions. It follows that its perimeter is bounded by $4\pi \max\{\|zq\|, \omega_n\}$, so that $\lambda_2(U_i') \le 4\pi \max\{\|zq\|, \omega_n\}$ for every $i \ge 0$. It now follows easily that there exist universal constants C, C' such that

$$\mathbb{E}\ |B_W^\star| = \mathbb{E}\ \mathbb{E}\left[|B_W^\star| \,\middle|\, R_i, i \ge 0\right] \le C\ \mathbb{E}\left[\sum_{j=0}^{\kappa-1} R_j + \kappa + \max\{\|zq\|, \omega_n\}\right]$$

$$\le C' \max\{\|zq\|, \omega_n\}.$$

For the concentration, we use the fact that if $\|xW\|, \|x\partial W^{\odot}\| \le \omega_n$ then the chromatic number χ of the dependence graph of the family $\mathbb{1}_{x \in B_W^\star}$ is bounded by the maximum number of points of $\mathbf{X}_n$ contained in a disc of radius $2\omega_n$. We then have

$$\begin{aligned}
\mathbb{P}(\chi \ge 8\pi\omega_n^2) &\le \mathbb{P}(\exists x \in D : \mathbf{X}_n \cap b(x, 2\omega_n)) \\
&\le \mathbb{E}\left[\sum_{x \in \mathbf{X}_n} \mathbb{P}_x(|b(x,2\omega_n) \cap \mathbf{X}_n| \ge 8\pi\omega_n^2)\right] \\
&\le \mathbb{E}\left[\sum_{x \in \mathbf{X}_n} \mathbb{P}(\mathrm{Po}(4\pi\omega_n^2) \ge 8\pi\omega_n^2)\right] \\
&\le \exp\left(-\omega_n^2\right),
\end{aligned}$$

for all n large enough, using the bounds for Poisson random variables we have already used in the proof of Corollary 5.16. Let

$$W^{\partial} := \left\{x \in W^{\odot} \,\middle|\, \max\{\|xW\|, \|x\partial W^{\odot}\|\} \le \omega_n\right\}.$$

Following Equation (5.35) and by the convexity of $W^{\odot}$, there exists a universal constant C'' such that

$$\mathbb{P}\left(|\mathbf{X}_n \cap W^{\partial}| \ge C'' \max\{L, \omega_n^5\} \cdot \omega_n^2\right) \le \exp\left(-\omega_n^2\right).$$

By Theorem 3.2 of Dubhashi and Panconesi [51], we thus obtain for $t > 0$,

$$\begin{aligned}
&\mathbb{P}(|B^{\star}_W| \geq \mathbb{E}\ |B^{\star}_W| + t) \\
&\leq \mathbb{E}\left[\exp\left(-\frac{2t^2}{\chi \cdot |\mathbf{X}_n \cap W^{\partial}|}\right)\right] \\
&\leq \exp\left(-\frac{t^2}{8\pi\, C'' \max\{L_0, \omega_n^5\} \cdot \omega_n^4}\right) + \mathbb{P}\left(\chi \geq 8\pi\omega_n^2\right) + \mathbb{P}\left(|\mathbf{X}_n \cap W^{\partial}| > C'' \max\{L_0, \omega_n^5\} \cdot \omega_n^2\right) \\
&\leq \exp\left(-\frac{t^2}{8\pi\, C'' \max\{L_0, \omega_n^5\} \cdot \omega_n^4}\right) + 2\exp\left(-\omega_n^2\right).
\end{aligned}$$

The result follows for n sufficiently large by choosing $t \coloneqq C''\left(L_0 + \omega_n^6\right)$. □

Algorithmic complexity

Whilst we have given explicit bounds on the number of sites in $\mathbf{X}_n$ that may be accessed by an instance of Cone-Walk, we recall that the complexity of the algorithm Cone-Walk(z, q) does not follow directly. This is because the Cone-Walk algorithm must do a small amount of computation at each step in order to compute the vertex which should be chosen next. We proceed by defining a random variable $T(z, q)$, which will denote the number of operations required by Cone-Walk(z, q) in the RAM model of computation given an implementation based upon a priority queue. We conjecture that the bound for Proposition 5.22 given in this section is not tight, and that the algorithmic complexity is fact be bounded by $O(L_0 + \omega_n^4)$. Unfortunately the dependency structure in algorithms of this type makes such bounds difficult to attain.

Proposition 5.22. *Let $T(z, q)$ be the number of steps required by the* Cone-Walk *algorithm to compute the sequence of stoppers given by the cone-walk process between z and q along with the path generated by* Simple-Path *in the RAM model of computation. Let $c_1 > 0$ be an implementation-dependent constant, then for n large enough, we have*

$$\mathbb{P}\left(T(z, q) > c_1 \cdot L_0 \log \omega_n + \omega_n^4\right) \leq 10\exp\left(-\omega_n^{3/2}\right).$$

Proof. Define M_i to be the number of sites accessed during the *i'th* step of the algorithm. We fix z, q and write T to denote $T(z, q)$ for brevity. From Section 5.1, we know that for sufficiently large n, we may choose a constant c_2 such that

$$T \leq c_2 \sum_{i=0}^{\kappa-1} \tau_i \log(M_i). \tag{5.37}$$

So that to bound T, it suffices to bound the sum in (5.37). We have

$$\begin{aligned}
&\mathbb{P}\left(\sum_{i=0}^{\kappa-1}\tau_i\log(M_i)\geq 15({L_0}+{\omega_n}^3)\log(\omega_n)\right)\\
&\qquad\leq \mathbb{P}\left(\sum_{i=0}^{\kappa-1}\tau_i\log(M_i)\geq 3({L_0}+{\omega_n}^3)\log(\omega_n^5)\mid M_{\max}<\omega_n^5\right)+\mathbb{P}\left(M_{\max}\geq\omega_n^5\right)\\
&\qquad\leq \mathbb{P}\left(\sum_{i=0}^{\kappa-1}\tau_i\geq 3({L_0}+{\omega_n}^3)\right)+\exp\left(-\omega_n^{3/2}\right).
\end{aligned}$$

The first term of which is bounded by Proposition 5.17, and the bound on $M_{\max}$ comes directly from the proof of Proposition 5.12. □

5.3 Relaxing the model and bounding the cost of the worst query

About the shape of the domain and the location of the aim

The analysis in Section 5.2 was provided given the assumption that q was the centre of a disc containing $\mathbf{X}$ for clarity of exposition. We now relax the assumptions on both the shape of the domain D and the location of the query q. Taking D to be a disc with q at its centre ensured that $\mathrm{Disc}(z,q,r)$ was included in D for $r\leq\|zq\|$ and thus the search cone and disc were always entirely contained in the domain D. If we now allow q to be close to the boundary, it may be that part of the search cone goes outside D.

To begin with, we leave D unchanged and allow q to be any point in D. Given any point $z\in D$, the convexity of D ensures that the line segment zq lies within D. Furthermore, one of the two halves of the disc of diameter zq is included within D. Thus for any $z,q\in D$ and $r\in\mathbf{R}$ the portion of $\mathrm{Cone}(z,q,r)$ (resp. $\mathrm{Disc}(z,q,r)$) within D has an area lower bounded by half of its actual area (including the portion outside D). Since the distributions of all of the random variables rely on estimations for the portions of area of $\mathrm{Cone}(z,q,r)$ or $\mathrm{Disc}(z,q,r)$ lying inside D, we have the same order of magnitude for κ, K, Λ and T, with only a degradation of the relevant constants. The proofs generalise easily, and we omit the details. (Note however, that upper and lower bounds in an equivalent of Proposition 5.15 would not match any longer.)

The essential property we used above is that a disc with a diameter within D has one of its halves within D. This is still satisfied for smooth convex domains D and for discs whose radius is smaller than the minimal radius of curvature of ∂D. Thus our analysis may be carried out provided all the cones and discs we consider are small enough. The conditioning on the event G_k which we used in Section 5.2 precisely guarantees that for all n large enough, on G_k, all the regions we consider are small enough ($O(\log n/\sqrt{n})=o(1)$ in this scaling), and that G_k still occurs with high probability. These remarks yield the following result. As before, $D_n=\sqrt{n}D$ denotes the scaling of D with area n.

Proposition 5.23. *Let D be a fixed smooth convex domain of area 1 and diameter δ. Consider a Poisson point process $\mathbf{X}_z^n$ of intensity 1 contained in $D_n=\sqrt{n}D$. Let $z,q\in D_n$. Let $\mathbf{X}_z^n$ be*

$\mathbf{X}_n$ *conditioned on* $z \in \mathbf{X}_n$. *Then, there exist constants* $C_{\Gamma,D}$, $A_{\Gamma,D}$, $\Gamma \in \{\kappa, K, \Lambda, T\}$, *such that for the Cone Walk on* $\mathbf{X}_z^n$, *and all* n *large enough, we have*

$$\sup_{z,q\in D_n} \mathbb{P}\left(\Gamma(z,q) > C_{\Gamma,D} \cdot \|zq\| + (1 + \sqrt{\|zq\|})\omega_n^5\right) \leq A_{\Gamma,D} \cdot \exp\left(-\omega_n^{3/2}\right). \tag{5.38}$$

Thus we obtain upper tail bounds for the number of steps $\kappa(z,q)$, the number of visited sites $K(z,q)$, the length $\Lambda(z,q)$ and the complexity $T(z,q)$ which are *uniform* in the starting point z and the location of the query q. We now move on to strengthening the results to the worst queries (still in a random Delaunay).

The worst query in a Poisson Delaunay triangulation

In this section, we prove a Poissonised version of Theorems 5.1 and 5.3. The proof of the latter are completed in Section 5.3. Consider $\sup_{z\in\mathbf{X}_n, q\in D_n} \Gamma(z,q)$, the value of the parameter for the *worst possible* pair of starting point and query location, for $\Gamma \in \{\kappa, K, \Lambda, T\}$.

Theorem 5.24. *There exist a constant* $C_{\Gamma,D}$ *depending only on* Γ *and on the shape of* D *such that, for all* n *large enough,*

$$\mathbb{P}\left(\exists z \in \mathbf{X}_n,\, q \in D_n \;:\; \Gamma(z,q) > C_{\Gamma,D} \cdot \|zq\| + (1 + \sqrt{\|zq\|}) \log^4 n\right) \leq \frac{1}{n^2}.$$

By Lemma 5.10, the number of possible walks in a given Delaunay tessellation of size n is at most $n \times n^4$. Although it *seems* intuitively clear that this should be sufficient to bound the parameters for the worst query, it is not the case: one needs to guarantee that there is a way to sample points *independently* from $\mathbf{X}_n$ such that all the cells are hit, or in other words that the cells are not too small. We prove the following:

Proposition 5.25 (Stability of the walk)**.** *There exists a partition of* D *into at most* $2n^4$ *cells such that the sequence of steps used by the Cone Walk algorithm from any vertex of the triangulation does not change when* q *moves in a region of the partition. Furthermore, let* $a(\mathbf{X}_n)$ *denote the area of the smallest cell. Then*

$$\mathbb{P}(a(\mathbf{X}_n) < \eta) < 1 - O(n^8\eta^{1/3}), \tag{5.39}$$

for all n *large enough.*

The proof of Proposition 5.25 relies on bounds on the smallest angle and on the shortest line segment in the line arrangement. The argument is rather long, and we think it would dilute too much the focus of this section, so we prove it in Section 5.4.

Proof of Theorem 5.24. In order to control the behaviour of the walk aiming at any point $q \in D$, it suffices to control it for one point of each face of the subdivision associated with the line arrangement $\Xi(\mathbf{X}_n)$ introduced in Section 5.1. Let $\mathcal{Q}_n = \{q_i : 1 \leq 1 \leq n^{2k})\}$, for some $k \geq 1$ to be chosen later, be a collection of iid uniform random points in D, independent of $\mathbf{X}_n$. Let E_n be the event that every single face of the subdivision contains

at least one point of Q_n. Since there are at most $|\mathbf{X}_n|^4$ regions, Proposition 5.25 implies that

$$\begin{aligned}\mathbb{P}(E_n^c) &\leq \mathbb{P}\left(E_n^c \mid a(\mathbf{X}_n) \geq n^{-k}\right) + \mathbb{P}\left(a(\mathbf{X}_n) < n^{-k}\right)\\ &\leq \mathbb{E}[|\mathbf{X}_n|^4] \cdot \mathbb{P}(\mathrm{Bin}(n^{2k}, n^{-k}) = 0) + O(n^8 \cdot n^{-k/3})\\ &= O(n^4 \cdot \exp\left(-n^{k/2}\right)) + O(n^{8-k/3})\\ &= O(n^{8-k/3}).\end{aligned} \tag{5.40}$$

For $x > 0$ write $f_n(x) := C_{\Gamma,D}x + (1 + \sqrt{x})\log^4 n$. Then, we have

$$\begin{aligned}&\mathbb{P}\left(\exists z \in \mathbf{X}_n, q \in D_n \ : \ \Gamma(z,q) \geq f_n(\|zq\|)\right)\\ &\leq \mathbb{P}\left(\exists z \in \mathbf{X}_n, q \in D_n \ : \ \Gamma(z,q) \geq f_n(\|zq\|) \,\middle|\, E_n\right) + \mathbb{P}(E_n^c)\\ &= \mathbb{P}\left(\exists z \in \mathbf{X}_n, q \in Q_n \ : \ \Gamma(z,q) \geq f_n(\|zq\|)\right) + \mathbb{P}(E_n^c)\\ &\leq n^{2k} \cdot \sup_{q \in D} \mathbb{P}\left(\exists z \in \mathbf{X}_n \ : \ \Gamma(z,q) \geq f_n(\|zq\|)\right) + \mathbb{P}(E_n^c).\end{aligned} \tag{5.41}$$

Now for any fixed $q \in D$, and conditioning on $\mathbf{X}_n$, we see that

$$\begin{aligned}\mathbb{P}\left(\exists z \in \mathbf{X}_n \ : \ \Gamma(z,q) \geq f_n(\|zq\|)\right) &\leq \mathbb{E}\left[\sum_{z \in \mathbf{X}_n} \mathbb{1}_{\Gamma(z,q) \geq f_n(\|zq\|)}\right]\\ &\leq \mathbb{E}\left[\sum_{z \in \mathbf{X}_n} \mathbb{1}_{\Gamma(z,q) \geq f_n(\|zq\|), |\mathbf{X}_n| \leq 2n}\right] + \mathbb{P}\left(|\mathbf{X}_n| > 2n\right)\\ &\leq 2n \sup_{z \in D} \mathbb{P}(\Gamma(z,q) \geq f_n(\|zq\|)) + \exp\left(-n/3\right).\end{aligned}$$

The claim follows from (5.40), (5.41), and Proposition 5.23 by choosing $k = 40$. (Note that we could easily obtain sub-polynomial bounds.) □

Before concluding this section, we note that the same arguments also yield the Poissonised version of Theorem 5.1 about the number of neighbourhoods required to compute all the steps of any query. We omit the proof.

Theorem 5.26. *We have the following bound for any run of* Cone-walk *on* $\mathrm{Del}(\mathbf{X}_n)$*, for* $\mathbf{X}_n$ *a Poisson point process of rate* n *in* D*:*

$$\mathbb{P}\left(\exists z \in \mathbf{X}_n, q \in D_n : M(z,q) > \log^{1+\epsilon} n\right) \leq \frac{1}{n^2}. \tag{5.42}$$

De-Poissonisation: Proof of Theorem 5.3

In this section, we prove that the results proved in the case of a Poisson point process of intensity n in a compact convex domain D (Theorem 5.24) may be transferred to the situation where the collection of sites consists of n independent uniformly random points in D. In the present case, the concentration for our events is so strong that the de-Poissonisation is straightforward.

Note that conditional on $|\mathbf{X}_n| = n$, the collection of points $\mathbf{X}_n$ is precisely distributed like n independent uniforms in D. Furthermore, $|\mathbf{X}_n|$ is a Poisson random variable with mean n, so that by Stirling's formula, as $n \to \infty$,

$$\mathbb{P}(|\mathbf{X}_n| = n) = \frac{n^n \exp(-n)}{n!} \sim \frac{1}{\sqrt{2\pi n}}. \tag{5.43}$$

This is small, but we can consider multiple copies of the process; if one happens to have exactly n points, but that none of them actually behaves badly, then it must be the case that the copy that has n points behave as it should, which is precisely what we want to prove. To make this formal, consider a sequence $\mathbf{X}_{ni}$, $i \geq 1$, of iid Poisson point processes with mean n in D. Let η_n be the first $i \geq 1$ for which $|\mathbf{X}_{ni}| = n$. Then, for any event E^i defined on $\mathbf{X}_{ni}$,

$$\begin{aligned}
\mathbb{P}(E_1 \mid |\mathbf{X}_{n1}| = n) &= \mathbb{P}(E_{\eta_n}) \\
&\leq \mathbb{P}(\exists j \leq \eta_n : E_j) \\
&\leq \mathbb{P}(\exists j \leq n : E_j, \eta_n \leq n) \\
&\leq n\,\mathbb{P}(E_1) + \mathbb{P}(\eta_n \geq n) \\
&\leq n\,\mathbb{P}(E_1) + \exp\left(-\sqrt{n/(2\pi)}\right),
\end{aligned}$$

where the last line follows from (5.43). In the present case, we apply the argument above to estimate the probabilities that the number of steps, the number of sites visited, the length of the walk or the complexity of the algorithm exceeds a certain value. In the case of the Poisson point process, we have proved that any of these events has probability at most $1/n^2$, so that we immediately obtain a bound of $O(1/n)$ in the case of n iid uniformly random points, which proves the main statement in Theorem 5.3.

This implies immediately that, for δ the diameter of D, we have

$$\begin{aligned}
\mathbb{E}\left[\sup_{z \in \mathbf{X}^n, q \in D_n} \Gamma(z,q)\right] &\leq 2\delta C_\Gamma \sqrt{n} + \mathbb{P}\left(\sup_{z \in \mathbf{X}_n, q \in D_n} \Gamma(z,q) \geq 2\delta C_\Gamma \sqrt{n}\right) \\
&\leq 2\delta C_\Gamma \sqrt{n} + O(1/n) \\
&\leq 3\delta C_\Gamma \sqrt{n},
\end{aligned}$$

for n large enough, so that the proof of Theorem 5.3 is complete. The proof of Theorem 5.1 from Theorem 5.26 relies on the same ideas and we omit the proof.

5.4 Extremes in the arrangement $\Xi(\mathbf{X}_n)$

In this section, we obtain the necessary results about the geometry of the line arrangement Ξ introduced in Section 5.1. In particular, we prove Proposition 5.25 providing a uniform lower bound on the areas of the cells of the subdivision associated to Ξ which is crucial to the proofs of our main results Theorems 5.1 and 5.3.

The rays (half-lines) of the arrangement Ξ all have one end at a site of $\mathbf{X}_n$. Let Ξ_x denotes the set of rays of Ξ with one end at x; we say that the rays in Ξ_x originate at x. The rays come in two kinds:

- *regular rays* are defined by a triple x, y, y' such that there exists $r > 0$ and $q \in \mathbf{R}^2$ for which y and y' both lie on the front arc of the $\mathrm{Cone}(x, q, r)$; such a ray is denoted by $\rho^-(x, y, y')$;
- *extreme rays* correspond to one of the straight boundaries of $\mathrm{Cone}(x, y, \infty)$, for some pair of points x, y; these two rays are denoted by $\rho_1^+(x, y)$ and $\rho_2^+(x, y)$ in such a way that the angle from ρ_1^+ to ρ_2^+ is $\pi/4$.

We let Ξ_x^- and Ξ_x^+ denote the collections of regular and extreme rays of Ξ_x, respectively. Our strategy to bound the area of the smallest cell is to lower bound the angle between any two rays (Section 5.4) and the length of any line segment (Section 5.4). We put together the arguments and prove Proposition 5.25 in Section 5.4.

The smallest angle between two rays

We start with a bound on the smallest angle in the arrangement.

Lemma 5.27. *Let $\langle \mathbf{X}_n \rangle$ denote the minimum angle between any two lines of $\Xi(\mathbf{X}_n)$ which intersect within D. Then, there exists a constant C such that, for any $\beta \in (0, \pi/8)$,*

$$\mathbb{P}(\langle \mathbf{X}_n \rangle < \beta) \leq Cn^5\beta. \tag{5.44}$$

Proof. For two intersecting rays ρ_1 and ρ_2, let $\langle \rho_1, \rho_2 \rangle$ denote the (smallest) angle they define. We first deal with the angles between two rays originating from different points. Consider a given ray $\rho_1 \in \Xi_{x_1}$, and let $\tilde{x}_1$ denote the intersection of ρ_1 with ∂D. For another ray $\rho_2 \in \Xi_{x_2}$, with $x_2 \neq x_1$, to intersect ρ_1 at an angle lying in $(-\beta, \beta)$, we must have $x_2 \in \mathrm{Cone}_\beta(x_1, \tilde{x}_1, \infty) \cup \mathrm{Cone}_\beta(\tilde{x}_1, x_1, \infty)$. Observe that for any two points x, y, the area of $\mathrm{Cone}_\beta(x, y, \infty) \cap D$ is bounded by $\delta^2 \tan\beta$, where δ denotes the diameter of D. In particular, for any fixed ray $\rho_1 \in \Xi_x$ for some $x \in \mathbf{X}_n$,

$$\begin{aligned}
&\mathbb{P}(\exists x_2 \in \mathbf{X}_{nx} \setminus \{x\}, \rho_2 \in \Xi_{x_2} : \langle \rho_1, \rho_2 \rangle < \beta) \\
&\leq \mathbb{P}(\exists x_2 \in \mathbf{X}_{nx} \setminus \{x\} : x_2 \in \mathrm{Cone}_\beta(x_1, \tilde{x}_1, \infty)) \\
&\quad + \mathbb{P}(\exists x_2 \in \mathbf{X}_{nx} \setminus \{x\} : x_2 \in \mathrm{Cone}_\beta(\tilde{x}_1, x_1, \infty)) \\
&\leq 2n\delta^2 \tan\beta,
\end{aligned}$$

since $\mathbf{X}_x^n \setminus \{x\}$ is a Poisson point process in D with intensity n. Now, since for any x, $|\Xi_x| \leq 2(|\mathbf{X}_n| - 1)$,

$$\begin{aligned}
&\mathbb{P}(\exists x_1, x_2 \in \mathbf{X}_n, x_1 \neq x_2, \rho_1 \in \Xi_{x_1}, \rho_2 \in \Xi_{x_2} : \langle \rho_1, \rho_2 \rangle < \beta) \\
&\leq \mathbb{E}\left[\sum_{x \in \mathbf{X}_n} \mathbb{P}(\exists \rho_1 \in \Xi_x, x_2 \in \mathbf{X}_{nx} \setminus \{x\}, \rho_2 \in \Xi_{x_2} : \langle \rho_1, \rho_2 \rangle < \beta) \right] \\
&\leq \mathbb{E}\left[\sum_{x \in \mathbf{X}_n} 2(|\mathbf{X}_n| - 1) \cdot 2n\delta^2 \tan\beta \right] \\
&= 4n^3\delta^2 \tan\beta,
\end{aligned} \tag{5.45}$$

since $\mathbf{X}_{nx} \setminus \{x\}$ is distributed like $\mathbf{X}_n$.

We now deal with the smallest angle between any two rays in the same set Ξ_x, for some $x \in \mathbf{X}_n$. Any point $y \in \mathbf{X}_{nx} \setminus \{x\}$ defines at most two extreme rays in Ξ_x which intersect at an angle of $\pi/4$. For any two distinct points y, y' to define two extreme rays that intersect at an angle smaller than $\beta > 0$, then y' must lie in $\mathrm{Cone}_\beta(x, y, \infty)$ or one of its two images in the rotations of angle $+\pi/8$ or $-\pi/8$ about x. The arguments we have used to obtain (5.45) then imply that

$$\begin{aligned}\mathbb{P}(\exists x \in \mathbf{X}_n : \exists \rho, \rho' \in \Xi_x^+, \langle \rho, \rho' \rangle < \beta) &\leq 3n^3\delta^2 \tan\beta \\ &\leq 6n^3\delta^2\beta,\end{aligned}$$

for all $\beta \in [0, \pi/4]$, since then $\tan\beta \leq 2\beta$.

We now consider a regular ray $\rho = \rho^-(x, y, z)$. Then there exist $q \in \mathbf{R}^2$ and $r > 0$ such that ρ is the half-line with an end at x and going through q, and x, y, z all lie on the same circle C (more precisely, y, z lie on the intersection of C with $\mathrm{Cone}(x, q, \infty)$). We first bound the probability that there exists a point z' such that the regular ray $\rho' = \rho^-(x, y, z')$ intersects ρ at an angle at most β, that is $\rho' \in \mathrm{Cone}_\beta(x, q, \infty)$. We want to bound the area of the region

$$\left\{ z' \in D : \rho^-(x, y, z') \in \mathrm{Cone}_\beta(x, q, \infty) \right\}, \tag{5.46}$$

for which the angle between ρ and ρ' is at most β. By definition of ρ', x, y and z' lie on the same circle $C' = C'(c)$, which has a centre c' in $\mathrm{Cone}_\beta(x, q, \infty)$. More precisely, the centre c' lies on the intersection of the bisector of the line segment $[x, y]$ with $\mathrm{Cone}_\beta(x, q, \infty)$; so c' lies in a line segment ℓ of length $O(\beta)$ containing the centre c of C. The Hausdorff distance between C and $\cup_{c' \in \ell} C'(c)$ is $O(\beta)$ and it follows that

$$\lambda_2\left\{ z' \in D : \rho^-(x, y, z') \in \mathrm{Cone}_\beta(x, q, \infty) \right\} = O(\beta). \tag{5.47}$$

A similar argument applies to deal with rays $\rho^-(x, y', z')$ for $y', z' \in \mathbf{X}_n$: once the one of y' or z' is chosen, the second is forced to lie in a region of area $O(\beta)$. From there, arguments similar to the ones we used in (5.45) yield

$$\mathbb{P}(\exists x \in \mathbf{X}_n : \exists \rho, \rho' \in \Xi_x^-, \langle \rho, \rho' \rangle < \beta) = O(n^5\beta). \tag{5.48}$$

Finally, we deal with the intersections of a regular and an extreme ray. We $\rho = \rho^-(x, y, z)$ and $\rho' = \rho_1^+(x, y')$, for $y' \in \mathbf{X}_n$ (potentiall $y' = y$). For ρ' to lie in $\mathrm{Cone}_\beta(x, q, \infty)$, the point y' must lie in a cone with apex x, half-angle β and with axis one of the straight boundaries of $\mathrm{Cone}(x, q, \infty)$. So if $y \neq y'$, the point y' is constrained to lie in a region of area $O(\beta)$; in this case, the expected number of $x, y, z, y' \in \mathbf{X}_n$ in such a configuration is $O(n^4\beta)$. If on the other hand, $y' = y$ then y must lie on a portion of length $O(\beta)$ of the front arc of $C \cap \mathrm{Cone}(x, q, \infty)$. However, conditional on x, y, z indeed forming the ray $\rho = \rho^-(x, y, z)$, the points y is uniformly distributed on the front arc. In other words, the expected number of triplets x, y, z in such a configuration is $O(n^3\beta)$. It follows that

$$\mathbb{P}(\exists x \in \mathbf{X}_n : \exists \rho \in \Xi_x^-, \rho' \in \mathcal{S}_x^+, \langle \rho, \rho' \rangle < \beta) = O(n^4\beta). \tag{5.49}$$

Putting (5.45), (5.48) and (5.49) together completes the proof. □

The shortest line segment

Let $L(\mathbf{X}_n)$ denote the smallest line segment in the arrangement of lines $\Xi(\mathbf{X}_n)$. The aim of this section is to prove the following:

Lemma 5.28. *We have, for any γ small enough and n large enough,*

$$\mathbb{P}(L(\mathbf{X}_n) < \gamma) \leq n^9 \gamma^{1/2}. \tag{5.50}$$

In order to prove Lemma 5.28, we place a ball of radius γ at every intersection in $\Xi(\mathbf{X}_n)$ (including the ones which involve the boundary ∂D) and estimate the probability that any such ball is hit by another ray of the arrangement. The proof is longer and slightly more delicate than that of Lemma 5.27 simply because three rays, each involving up to three points of $\mathbf{X}_n$, might be involved the event of interest. The first step towards proving Lemma 5.28 is to establish uniform bounds on the area of location of some points for the rays to intersect a given ball. These bounds are stated in Lemmas 5.29 and 5.30 below.

Lemma 5.29. *There exists a constant C such that, for any $x, y, z \in D$, one has, for all $\gamma > 0$ small enough, we have*

1. $\lambda_2\{\bar{x} \in D : \rho^+(\bar{x}, y) \cap \delta(z, \gamma) \neq \varnothing\} \leq C\gamma$
2. $\lambda_2\{\bar{y} \in D : \rho^+(x, \bar{y}) \cap \delta(z, \gamma) \neq \varnothing\} \leq C\gamma / \|xz\|$.

Proof. Fix x, z and $\gamma > 0$. The location of points $\bar{y}$ for which $\rho^+(x, \bar{y}) \ni z$ is precisely $\partial \operatorname{Cone}(x, z, \infty)$. Then,

$$\bigcup_{q \in \delta(z,\gamma)} \partial \operatorname{Cone}(x, q, \infty) \tag{5.51}$$

consists of two cones of half-angle $\arcsin(\gamma / \|xz\|)$, which proves the second claim.

For the first claim, let $c_1 = c_1(z)$ and $c_2 = c_2(z)$ be the two intersections of the circles of radius $\|zy\| / \sqrt{2}$ centred at y and z (so the segment zy is seen from c_1 and c_2 at an angle of $\pi/2$). Let $C_1 = C_1(z)$ and $C_2 = C_2(z)$ denote the discs centred at c_1 and c_2, respectively, and having y and z on their boundaries. Then, the region of the points $\bar{x}$ for which $\rho^+(\bar{x}, y) \ni z$ is precisely the boundary of $C_1 \cup C_2$. When looking for the region of points $\bar{x}$ satisfying $\rho^+(\bar{x}, y) \cap \delta(z, \gamma) \ni q$, for some $q \in \delta(z, \gamma)$ observe that the centres $\|c_1(q) - c_1(z)\| = O(\gamma)$, $\|c_2(q) - c_2(z)\| = O(\gamma)$ and that the Hausdorff distance between $C_1(q) \cup C_2(q)$ and $C_1 \cup C_2$ is $O(\gamma)$, hence the first claim. □

Lemma 5.30. *There exists a constant C such that, for any $x, y, y', z \in D$, one has, for all $\gamma > 0$ small enough*

1. $\lambda_2\{\bar{x} \in D : \rho^-(\bar{x}, y, y') \cap \delta(z, \gamma) \neq \varnothing\} \leq C\gamma$
2. $\lambda_2\{\bar{y} \in D : \rho^-(x, \bar{y}, y') \cap \delta(z, \gamma) \neq \varnothing\} \leq C\gamma$.

Proof. The ray $\rho^-(x,y,y')$ is the half-line with one end at x and going through the centre c of the circle which contains all three x, y and y'. So the region of the points $\bar{x}$ for which $\rho^-(\bar{x},y,y') \ni z$ is naturally indexed by the points of the bisector Δ of the segment $[y,y']$: for a point $c \in \Delta$, the only possible points $\bar{x} = \bar{x}(c)$ lie the intersection of the circle C centred at c and containing y (and y') and the line containing c and z. So if $z \neq c$, there are two such points and at most one of them is such that $y, y' \in \mathrm{Cone}(\bar{x}, c, \infty)$. If $z = c$, then there is potentially a continuum of points $\bar{x}(c)$, consisting of the arc of C for which $y, y' \in \mathrm{Cone}(\bar{x}, c, \infty)$. It follows that for every $c \in \Delta$ such that $\|zc\| > 2\gamma$, the points $\bar{x}(c)$ for which $\rho^-(\bar{x},y,y')$ intersects $\delta(z,\gamma)$ is an arc of length $O(\gamma)$. On the other hand, the cumulated area of the circles $C(c)$ for which $\|zc\| \leq 2\gamma$ is $O(\gamma)$. Hence the first claim.

For x, y and q fixed, the set of points $\bar{y}$ such that $\rho^-(x,y,\bar{y}) \ni q$ is contained in the circle $C'(q)$ whose centre is the intersection of the bisector of $[x,y]$ and the line (xq), and which contains x (and y). The cumulated area of $C'(q)$ when $q \in \delta(z,\gamma)$ is $O(\gamma)$, which proves the second claim. □

Proof of Lemma 5.28. Recall that, in order to lower bound the length of the shortest line segment in $\Xi(\mathbf{X}_n)$, we prove using Lemmas 5.29 and 5.30, that if we place a ball of radius γ at every intersection in $\Xi(\mathbf{X}_n)$, every ball contains a single intersection with probability at least $1 - O(n^9\gamma^{1/2})$. The intersections to consider are of three different types: the points $x \in \mathbf{X}_n$, the intersections of one ray and the boundary of the domain D, and intersections between two rays.

The case of balls centred at points of $\mathbf{X}_n$ first is easily treated. We have

$$\mathbb{P}(\exists z, x \in \mathbf{X}_n : z \neq x, \Xi_x \cap \delta(z,\gamma) \neq \varnothing) \leq \mathbb{E}\left[\sum_{z \in \mathbf{X}_n} \mathbb{1}_{\exists x \in \mathbf{X}_n \setminus \{z\} : \Xi_x \cap \delta(z,\gamma) \neq \varnothing}\right].$$

Note that, in the expected value on the right, although $x \neq z$, the set Ξ_x does depend on z in a non-trivial way. The main point about the proof is to check that the conditioning on $z \in \mathbf{X}_n$ does not make it much easier for a ray to intersect $\delta(z,\gamma)$; and this is where Lemmas 5.29 and 5.30 enter the game. We have

$$\begin{aligned}\mathbb{E}\left[\sum_{z \in \mathbf{X}_n} \mathbb{1}_{\exists x \in \mathbf{X}_n \setminus \{z\} : \Xi_x \cap \delta(z,\gamma) \neq \varnothing}\right] &\leq \mathbb{E}\left[\sum_{z \in \mathbf{X}_n} \sum_{y \in \mathbf{X}_n} \sum_{x \in \mathbf{X}_n \setminus \{z,y\}} \mathbb{1}_{\rho^+(x,y) \cap \delta(z,\gamma) \neq \varnothing}\right] \\ &\quad + \mathbb{E}\left[\sum_{z \in \mathbf{X}_n} \sum_{y,y' \in \mathbf{X}_n} \sum_{x \in \mathbf{X}_n \setminus \{z,y,y'\}} \mathbb{1}_{\rho^-(x,y,y') \cap \delta(z,\gamma) \neq \varnothing}\right] \\ &\leq Cn^4\gamma,\end{aligned}$$

by Lemmas 5.29 and 5.30, since conditional on $z, y, y' \in \mathbf{X}_n$, $\mathbf{X}_n \setminus \{x,y,y'\}$ is distributed like $\mathbf{X}_n$.

We now move on to the intersections of rays with the boundary of D. Let ρ be a ray, and let z denote its intersection with ∂D. If ρ is an extreme ray, then there exists $x, y \in \mathbf{X}_n$ such that $\rho = \rho_1^+(x,y)$ (or $\rho = \rho_2^+(x,y)$) and we are interested in the probability that some other ray ρ' intersects $\delta(z,\gamma)$; note that the ray ρ' might arise from a set of points x', y' (if it is extreme) or x', y', y'' (if it is regular) which uses some of x or y. If there is at least

one point of x', y' that is not in $\{x, y\}$ then Lemma 5.29 allows us to bound the probability that $\rho^+(x', y')$ intersects $\delta(z, \gamma)$; in the other case, there must be one of x', y', y'' which is not in $\{x, y\}$ and Lemma 5.30 applies: For each such choice of points, the probability that ρ' intersects $\delta(z, \gamma)$ is $O(\gamma)$, uniformly in x, y and z. So the only case remaining to check is when ρ' is only defined by the two points x, y. Here,

- either $\rho' \in \Xi_x$ and for ρ' to intersect $\delta(z, \gamma)$ one must have $x \in \delta(z, \gamma)$, which only happens if x lies within distance γ of ∂D, hence with probability at most $O(n\gamma)$;

- or $\rho' \in \Xi_y$ and the ball $\delta(z, \gamma)$ should be close to one of the two points in $\partial \operatorname{Cone}(x, y, \infty) \cap \partial \operatorname{Cone}(y, x, \infty)$; Write $i_1 = i_1(x, y)$ and $i_2 = i_2(x, y)$ for these two points. More precisely, since the angle between ρ and ρ' is $\pi/4$, the point z should lie within distance $\gamma\sqrt{2}$ of one of the two intersections (or $\delta(z, \gamma)$ does not intersect ρ'). Dually, for a point q, let $\bar{y} = \bar{y}(q)$ be the point such that $i_1(x, \bar{y}) = q$. Then, for $\|zq\| \leq \gamma' = \gamma\sqrt{2}$ the angle between the vectors $x\bar{y}(q)$ and $x\bar{y}(z)$ is at most $\arcsin(\gamma'/\|xz\|)$ and the difference in length is an additive term of at most $\gamma'\sqrt{2}$. Overall, $\{\bar{y} : i_1(x, \bar{y}) \in \delta(z, \gamma')\}$ is contained in a ball of radius at most $3\gamma'\sqrt{2} = 6\gamma$. Finally, since $z \in \partial D$, the probability that such a situation occurs for some $x, y \in \mathbf{X}_n$ is at most $O(n^2\gamma)$.

As a consequence, the probability that there exist two rays ρ and ρ', such that $\rho' \cap \delta(z, \gamma) \neq \emptyset$, for $z = \rho \cap D$ is $O(n^5\gamma)$.

It now remains to deal with the case of balls centred at the intersection of two rays. Consider two rays, $\rho_1 \in \Xi_{x_1}$ and $\rho_2 \in \Xi_{x_2}$, for $x_1 \neq x_2$, which are supposed to intersect at a point z (the case $x_1 = x_2$ has already been covered since then, we have $z \in \mathbf{X}_n$). The definition of ρ_1 and ρ_2 may involve up to six points x_1, y_1, y_1' and x_2, y_2, y_2' of $\mathbf{X}_n$. The probability that some third ray ρ_3, whose definition uses at least one new point of $\mathbf{X}_n$, intersects $\delta(z, \gamma)$ can be upper bounded using Lemmas 5.29 and 5.30 as before and we omit the details. The only new cases we need to cover are the ones when the definition of ρ_3 only involves points among x_1, y_1, y_1' and x_2, y_2, y_2'. We now show that all the possible configurations on six points only are unlikely to occur for γ small in the random point set $\mathbf{X}_n$.

Because of the number of cases to be treated, without a clear way to develop a big picture, we only sketch the remainder of the proof. We treat the cases where $\rho_1 = \rho^-(x_1, y_1, y_1')$ and $\rho_2 = \rho^-(x_2, y_2, y_2')$, so in particular the two rays which intersect at z are regular rays.

- If $\rho_3 \in \Xi_{x_1}$ (or, by symmetry, $\rho_3 \in \Xi_{x_2}$), the bound on the smallest angle between any two rays in Lemma 5.27 ensures that z ought to be close to x_1: one must have $\arcsin(\gamma/\|x_1 z\|) \leq \eta$ where η denotes the smallest angle. Then, either x_2 is also close to z, or y_2 and y_2' actually almost lie on a circle whose centre lies on the line (x_1, x_2). Both are unlikely: (1) the closest pair of points lie at distance $\Omega(1/n)$ (which is much larger than what we are aiming for) and (2) for fixed x_1, x_2 and y_2, the location of points y_2' such that y_2 lies near the circle whose centre lies on (x_1, x_2) and which contains y_2 has area of order $O(\gamma)$.

- If $\rho_3 \in \Xi_{y_1}$ (or the other symmetric cases), there are the following possibilities: either ρ_3 is a regular ray and (up to symmetry)

$$\rho_3 \in \left\{ \begin{array}{llll} \text{(a)} & \rho^-(y_1, x_1, y_1'), & \text{(b)} & \rho^-(y_1, x_1, y_2) \\ \text{(c)} & \rho^-(y_1, x_2, y_1'), & \text{(d)} & \rho^-(y_1, x_2, y_2) \\ \text{(e)} & \rho^-(y_1, y_1', y_2'), & \text{(f)} & \rho^-(y_1, y_2, y_2') \end{array} \right\}, \tag{5.52}$$

  or ρ_3 is an extreme ray and

$$\rho_3 \in \left\{ \begin{array}{llll} \text{(g)} & \rho^+(y_1, y_1'), & \text{(h)} & \rho^+(y_1, y_2) \end{array} \right\}. \tag{5.53}$$

  Since $\rho^-(x_1, y_1, y_1')$ is a ray, y_1, y_1' both lie in a cone of half-angle $\pi/8$, and it is impossible that we also have x_1, y_1' both lying in some cone of half-angle $\pi/8$; so configuration (a) does not occur. For situation (b), we use Lemma 5.30 (ii). For situations (c)–(f), we use the arguments in the proof of Lemma 5.30 to exhibit the constraints about the location of the third point defining ρ_3, once the first two are chosen.

  To deal with the cases where $\rho_3 \in \Xi^+_{y_1}$, observe that once y_1 is placed, y_1' (or y_2) must lie in a cone of angle $\arcsin(\gamma/\|y_1 z\|)$: so either $\|y_1 z\| \geq r$, and the cone has area $O(\gamma/r)$ or $\|y_1 z\| \leq r$, but then y_1 must lie in the portion of the front arc of length $O(r)$ about ρ_1. Since the conditional distribution of y_1 is uniform on the arc, the probability that this configuration occurs is $O(\inf_{r \geq 0}\{\gamma/r + r\}) = O(\gamma^{1/2})$.

The remaining cases, where at least one of ρ_1 and ρ_2 is an extreme ray, may all be treated using similar arguments and we omit the tedious details. □

Lower bounding the area of the smallest cell: Proof of Proposition 5.25

With Lemmas 5.27 and 5.28 under our belt, we are now ready to prove Proposition 5.25. We use the line arrangement $\Xi(\mathbf{X}_n)$ of Section 5.1. The faces of the corresponding subdivision are convex regions delimited by the line segments between intersections of rays, and pieces of the boundary of the domain D. Let $L(\mathbf{X}_n)$ and $\langle \mathbf{X}_n \rangle$ denote the length of the shortest line segment and the smallest angle in the arrangement arising from $\mathbf{X}_n$. Consider a face of the subdivision, and one of its vertices u which does not lie on the ∂D. Then the triangle formed by the u and its two adjacent vertices on the boundary of the face is contained in the face. The lengths of the two line segments adjacent to u are at least $L(\mathbf{X}_n)$ long, and the angle they make lies between $\langle \mathbf{X}_n \rangle$ and $\pi - \langle \mathbf{X}_n \rangle$, so that the area of the corresponding triangle is at least

$$L(\mathbf{X}_n)^2 \cdot \frac{\tan\langle \mathbf{X}_n \rangle}{2}. \tag{5.54}$$

It follows easily from Lemmas 5.27 and 5.28 that the area of the smallest face it at least $\beta\gamma/2$ with probability at least $1 - O(n^4\beta) - O(n^9\gamma^{1/2})$. Choosing β, γ such that $n^4\beta = n^9\gamma^{1/2}$ yields the claim.

5.5 Comparison with Simulations

We implemented Cone-Walk in C++ using the Cgal libraries. For simulation purposes, we generated 10^7 points uniformly at random in a disc of area 10^7. We then simulated Cone-Walk on 10^6 different walks starting from a point within a disc having a quarter of the radius of the outer disc (to help reduce boundary effects) with a uniformly random destination. For comparison, we give give the expected bounds for a walk whose destination is at infinity. See Table 5.1. With respect to the path, we give two values for the number of extra vertices visited in a step. The first is the number obtained by using Simple-Path and the second, in brackets, is the average number of 'intermediary vertices' within each disc.

	Theory	**Theory (5 s.f.)**	**Simulation**
Radius	$\mathbb{E}[R] = \sqrt{\frac{\pi}{2\sqrt{2}+\pi}}$	0.72542	0.72557
# Intermediary path steps	$\leq \mathbb{E}[\tau_i] \leq \frac{4\pi}{\pi+2\sqrt{2}}$	≤ 1.1049	0.41244 (1.09574)
Simple-Path Length	$\leq \mathbb{E}[\Lambda/L_0] \leq \frac{22\pi-4\sqrt{2}}{2+3\pi+8\sqrt{2}}$	≤ 2.7907	1.51877

Table 5.1 – Comparison of theory with simulations. Inequalities are used to show when values are bounds.

Chapter 6

Pivot Walk

In this chapter, we describe a new walking strategy which traverses faces of planar Delaunay triangulations using orientation predicates. In particular, our algorithm gives a new way to navigate the *walk graph* introduced in Section 3.5. We will show that it is possible to design an orientation-based walk that requires only one orientation per face visited on average (according to simulations on uniformly random points), whilst not visiting significantly more faces than other well-known walking strategies. These gains will follow by exploiting the combinatorial structure in the walk graph defined in Definition 3.1. This result is an improvement on the average of 1.3 orientations per face achieved by Remembering Visibility Walk and 2 orientations achieved by Straight Walk (again, in the model of uniformly random points.) Kolingerová [77] and De Castro and Devillers [38] previously introduced heuristic methods to try and achieve a similar goal, by proposing heuristics that reduce the number of orientations to one per triangle. For their methods to be effective however, a good estimate of the number of steps that the walk will require is needed. Our technique does not work in the same way, and indeed the same trick used in these contexts could be used to further reduce the number of orientations required by our algorithm.

Unfortunately, whilst our result appears to be of general theoretical interest, we were unable to use it to gain a net speed-up in point location compared to other walking algorithms used for point location. This was initially surprising, and to understand the reasoning behind this, we provide optimised implementations for our algorithms and then analyse them in order to understand the cause of this behaviour.

Index test In addition to orientation predicates, we also make use of one more operation which we call the *index test*. For two neighbouring simplicies in a triangulation, $\sigma, \tau \in \mathrm{T}(\mathbf{X})$ the index test returns the index, $i \in \{0, \ldots, d\}$ such that the i*th* neighbour of σ is τ,

$$\textsc{Index}(\sigma, \tau) \mapsto i : \sigma.\,\mathrm{neighbour}(i) = \tau.$$

Given this definition, Remembering Visibility Walk (defined in Section 3.4) may be implemented using one index test and at most d orientations per simplex (for all simplices apart from the first.) We briefly remark that of these two operations, only the orientation

predicate actually provides information about the locations of the points, since the index test only gives combinatorial information about the triangulation.

6.1 Preliminaries

We will focus on the case of a locally finite set of n points, $\mathbf{X} \subset \mathbf{R}^2$ in general position, with convex hull $\mathrm{Conv}(\mathbf{X})$. We then construct a triangulation $\mathrm{T}(\mathbf{X})$. For now, we make no explicit assumptions on this triangulation, although for the most part we focus on the case where it is the Delaunay triangulation. We use $n_f := |\mathrm{T}(\mathbf{X})|$ as shorthand for the number of triangular-faces in $\mathrm{T}(\mathbf{X})$. We assume the walk starts from a face σ_0 and ends once a face is found containing the destination, q. We assume that the set $\mathbf{X} \cup \{q\}$ is also in general position to avoid unnecessary technicalities. Finally, for $x \in \mathbf{X}$, we define $\mathrm{star}(x) := \mathrm{star}(x, \mathrm{T}(\mathbf{X}))$ to be the set of faces in $\mathrm{T}(\mathbf{X})$ having x as a vertex.

Properties of the walk graph

We now prove some small results on the walk graph which will be useful in the following sections.

Lemma 6.1 (Sources and Sinks)**.** *For every $x \in \mathbf{X}$, let $\mathbf{W}_x := \mathbf{W}(\mathrm{star}(x), q)$ be the restriction of $\mathbf{W}$ to the star of x. Then, for all $x \in \mathbf{X}$, there exists exists a unique face in* $\mathrm{star}(x)$ *with in-degree 2 in $\mathbf{W}_x$. We call this face the* sink *of x. In addition, for every $x \in \mathbf{X}$ not on the convex hull of $\mathbf{X}$, there exists a unique face in* $\mathrm{star}(x)$ *having out-degree 2 in $\mathbf{W}_x$. We call this face the* source *of x. For an example, see Figure 6.2.*

Proof. We imagine tracing a ray between x and q. This ray can intersect at most 2 faces in the star of x (if x is on the convex hull, the ray only intersects one). It is easy to see that the face closest to q always exists, and has in-degree 2 in $\mathbf{W}_x$ (if it contains q the in-degree is 3 in $\mathbf{W}$.) If the ray intersects a second face, it is unique and has out-degree 2 by the same argument. □

Lemma 6.2. *Suppose that the number of points on the convex hull of $\mathbf{X}$ is c, then every face in $\mathrm{T}(\mathbf{X})$ (node in $\mathbf{W}$) is either the source or the sink of some point $x \in \mathbf{X}$. Also, there are exactly $n - c$ nodes in $\mathbf{W}$ with out-degree 2, $n - 3$ nodes with in-degree 2, and 1 node with in-degree 3 which represents the face containing q in $\mathrm{T}(\mathbf{X})$.*

Proof. We consider the source and sink triangles defined in Lemma 6.1. We observe that each of the $n - c$ points $x \in \mathbf{X}$ not on the convex hull must have a unique source face with out-degree 2 in $\mathbf{W}$. Now, the $n - 3$ points $x \in \mathbf{X}$ that don't contain q in their star have a unique sink with in-degree 2. Finally, there is one face remaining that contains q and has in-degree 3. □

Corollary 6.3. *Suppose that $c \in o(n)$ and let σ be a face selected uniformly at random from $\mathrm{T}(\mathbf{X})$, then the probability that σ has out-degree (or in-degree) 2 in $\mathbf{W}$ converges to $1/2$ as n tends to infinity.*

We briefly remark that Lemma 6.2 can be used to give an alternative proof to the well known identity $f \equiv 2n - 2 - c$.

Counting orientations

We already stated that Straight Walk can be implemented using an algorithm that requires exactly two orientations per face, and that the path given is a path in the walk graph $\mathbf{W}(\mathbf{X}, q)$. We additionally recall that Stochastic Visibility Walk works by doing a random walk in the walk graph $\mathbf{W}(\mathbf{X}, q)$, choosing among the possible out-going arcs (edges) for each step uniformly at random. The number of orientations required per face is not fixed in this case, however we can come up with an approximation by using Lemma 6.2 and considering the expected number of orientations required for a face chosen uniformly at random from $\mathrm{T}(\mathbf{X})$. Let N be the number of orientations required for such a face. To find the way on, Visibility Walk must check each of the three edges of the current face until it finds an out-going edge satisfying the visibility condition. We thus have

$$\begin{aligned}\mathbb{E}[N] &= \frac{1}{2}\mathbb{E}\left[N \;\middle|\; \left\{ 2 \text{ ways on} \right\}\right] + \frac{1}{2}\mathbb{E}\left[N \;\middle|\; \left\{ 1 \text{ way on} \right\}\right] \\ &= \frac{1}{2}\left(\left(\frac{2}{3}\cdot 1 + \frac{1}{3}\cdot 2\right) + \left(\frac{1}{3}\cdot 1 + \frac{1}{3}\cdot 2 + \frac{1}{3}\cdot 3\right)\right) \\ &= 1 + \frac{2}{3}.\end{aligned}$$

As we mentioned previously, a small improvement can be had by 'remembering' the last edge walked through by computing the index of the previous face visited and avoiding this orientation test (since it must fail). In practice, this means Remembering Walk requires fewer orientations on average per face. Repeating the above calculations in this case gives

$$\begin{aligned}\mathbb{E}[N] &= \frac{1}{2}\mathbb{E}\left[N \;\middle|\; \left\{ 2 \text{ ways on} \right\}\right] + \frac{1}{2}\mathbb{E}\left[N \;\middle|\; \left\{ 1 \text{ way on} \right\}\right] \\ &= \frac{1}{2}\left(1 + \left(\frac{1}{2}\cdot 1 + \frac{1}{2}\cdot 2\right)\right) \\ &= 1 + \frac{1}{4}.\end{aligned}$$

Evidently, these calculations may only be considered as an approximation, since the triangles are not chosen uniformly from the triangulation, but are chosen by an algorithm. This difference is underlined by the fact that our simulations give slightly different values to the estimates computed here. Correctly computing the expected number of orientations per triangle remains a difficult and unsolved problem.

6.2 Pivot Walk

We now introduce our new algorithm, which we call *Pivot Walk*. Pivot Walk traverses the walk graph similarly to other orientation-based algorithms given in Section 3.4, however it additionally exploits the combinatorial structure of the triangulation in order to reduce the average number of orientation predicates to approximately one per face visited (measured on random data). This result seems intuitively surprising, since orientation predicates are the only way the algorithm learns information about the locations of the underlying points, and we do not observe a large increase in the number of faces visited by the walk compared to, for example, Visibility Walk.

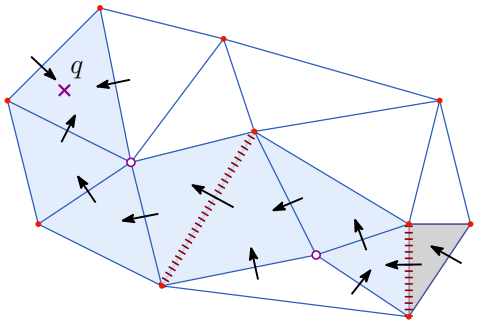


Figure 6.1 – An example of Pivot Walk. The faces visited by the walk are shaded, and the pivot points are marked. The pivot changes are signified by the thick dotted edges, and the orientations performed are marked by arrows.

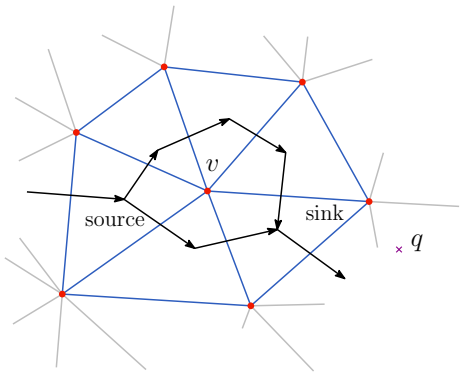


Figure 6.2 – Illustration for the definition of *sources* and *sinks*. We label the vertex v the *pivot*.

Pivot Walk is based on the simple observation in Lemma 6.1 that every vertex $x \in \mathbf{X}$ of has a unique sink. Thus, given any face in the star of x, we can *pivot* around x, doing a single orientation between the pivot and successive vertices in the star until the first time the orientation reverses. At this point we must have reached the sink for x. We then walk across the edge opposite x and choose a new pivot. Since the edge we walk across to the new pivot is the only way out of the sink, the new pivot we walk into tends to be well directed towards the destination. Walking around the pivot in this way is efficient since for every face visited, we only have to do one index test and one orientation. Unfortunately though, these gains are slightly negated by the cost of the first and last faces visited around a pivot. This is because we may need two orientation predicates to determine the correct 'pivot direction'. Also, every time we leave a pivot we will always do three orientations for one face (to ensure that the destination has not been reached). Figure 6.1 demonstrates an example run of Pivot Walk.

Skipping To improve Pivot Walk, we modify the algorithm to make it more lazy. Instead of checking for a direction before pivoting, we simply choose any direction at random and then *skip* a few steps around the pivot, without performing any orientation tests. Since the average degree in $\mathrm{T}(\mathbf{X})$ of a randomly chosen point $x \in \mathbf{X}$ is 6 (for $c \in o(n)$), independently of the triangulation (see de Berg et al. [36, Theorem 9.1]). This strategy almost always results in us getting closer to the sink face. Note that whilst we may walk the 'wrong way' across some edges in the walk graph, we are guaranteed that there exists a path that we *could have* taken, so this strategy does not affect the correctness of the algorithm.

Implementation We provide the pseudocode for Pivot Walk using the skipping strategy in Listing 6.1, which gives a complete description of a possible implementation of the algorithm already given. The code depends on the procedure Find-First- Face, which simply returns a valid neighbouring face of the given face which can be walked through, or the same face again if none exists. We use '$\otimes$' to mean logical exclusive or. We note that it is possible to further optimise the given procedure by caching the results of the index tests for the skipped faces, though we omit this from the pseudocode. As before, we assume that the orientation tests return Visible,Not-Visible or Colinear, depending on the configuration of the three input points.

Correctness The correctness of Pivot Walk follows since every we time we choose a new pivot, we must have visited at least one new face that has not previously been visited. In addition, for each pivot, we visit every face at most twice. The result then follows from Theorem 3.5.

6.3 Experiments

For testing purposes, we implemented Pivot Walk, Visibility Walk and Remembering Walk in C++ using the Cgal libraries with the standard EPIC (Exact Predicates Inexact Constructions) kernel [115]. This allowed us to be sure that our orientations could not fail due to numerical instability. We compiled our code using llvm 3.4 with `-O3`. We then generated random sets of uniform points in a unit square and constructed the Delaunay triangulation. Testing data for the walks was generated by choosing points uniformly at random away from the border, to prevent border effects from making the results unstable. We provide one version of Pivot Walk without the skipping strategy, and then a further four versions which skip different numbers of faces. To refer to these variations, we shall simply refer to 'Pivot k' to mean Pivot Walk using k skips (we omit k for Pivot Walk with no skipping.) Along with the standard implementations of our algorithms, we additionally consider two 'optimised' versions, one of which is for Pivot 2 and one of which is for Remembering Walk. These were added after observing that significant improvements could be gained from unrolling loops and expanding index tests. Our results are given in Table 6.1 and Fig 6.3. 3

Analysis of Results We observe that on random data, Pivot Walk with skipping tends to do significantly fewer orientations than the other walking strategies. In particular, Pivot

```
Pivot-Walk(q, current)
 1  previous = current
 2  // Find the first face we can walk through.
 3  current = Find-First-Face(q, current)
 4  // If no face was found, the destination was in the first face.
 5  if current = previous
 6      return current
 7  // Each iteration of this loop represents a new pivot.
 8  while true
 9      done = false
10      oriented = false
11      ccw = Rand(0, 1)
12      i = current.Index(previous)
13      previous = current
14      current = previous. neighbor(i + 2 − ccw  mod 3)
15      // Walk two steps around the pivot without orientations.
16      Skip(current, previous, ccw)
17      Skip(current, previous, ccw)
18      // Keep executing advance until it returns false.
19      while Advance(previous, current, q, pivot, oriented, ccw, done)
20      if done
21          return current
```

Listing 6.1 – A pseudocode implementation of Pivot-Walk. The Advance algorithm is given in Listing 6.2.

	Piv.	Piv. 1	Piv. 2	Piv. 3	Piv. 4	Vis.	Rem.	Rem. (opt)	Piv. (opt)
Time / ms	12.427	12.956	12.242	12.838	13.856	12.664	12.011	9.391	12.109
Net Orientations	16628556	14759528	12218120	11076802	11999199	18629794	13883677	-	-
Net Unique Faces	10209282	11761111	12087970	13145055	16174565	10967139	10797610	-	-

Table 6.1 – Combined table of results for two experiments. For timing, we generated a triangulation of 10^7 points and timed 10^4 randomly chosen walks for each strategy. When counting orientations and faces, we used 10^6 points and averaged over 10^4 walks. We do not give the number of index tests since (apart from Visibility Walk) this is (essentially) the same as the total number of faces.

3 required 20% fewer orientations than Remembering Walk and 40% fewer than Visibility Walk. Despite these gains, the number of faces visited in each of these cases was slightly higher than for Visibility Walk. We remark that Pivot 2 tended to be the fastest of the Pivot Walks, even though it does a few more orientations than Pivot 3. We thus propose that Pivot 2 should be the default implementation for Pivot Walk. We note that this version does only 1 orientation per face visited on average.

```
ADVANCE(previous, current, q, pivot, oriented, ccw, done)
 1  // Advance one step around the pivot.
 2  i = current.INDEX(previous)
 3  p = current.vertex(i)
 4  // If we found an edge pointing against us.
 5  if ccw ⊗ ORIENTATION(pivot, p, q) ≠ VISIBLE
 6      p' = current.vertex(i + 1 + ccw  mod 3)
 7      if not oriented
 8          // If we were going the wrong way.
 9          if (ccw ⊗ ORIENTATION(pivot, p', q) ≠ VISIBLE)
10              // Turn around.
11              SWAP(current, previous)
12              ccw = 1 − ccw
13              oriented = true
14              return true
15      // If we reached the destination.
16      if ccw ⊗ ORIENTATION(p, p', q) ≠ VISIBLE
17          done = true
18          return false
19      // Go to next pivot.
20      previous = current
21      current = previous.neighbour(i + 2 − ccw  mod 3)
22      return false
23  // Continue around pivot.
24  previous = current
25  current = previous.neighbour(i + ccw + 1  mod 3)
```

Listing 6.2 – The ADVANCE algorithm required by the Pivot Walk algorithm. Note that we allow the algorithm ADVANCE to modify its arguments to avoid unwieldy return statements.

It seems surprising that despite significantly reducing the number of exact computations required, Pivot 3 is still no faster than Remembering Walk. This appears to be because the cost of accessing a face in the triangulation has a cost comparable to that of doing an orientation. This was unexpected at first, and we therefore suggest some possible reasons for this behaviour. Firstly, it could simply be because Cgal filtered arithmetic is extremely efficient, with non-degenerate orientations generally reducing to a small number of arithmetic instructions which a modern processor can execute very efficiently. It could also be partly due to the increase in index tests required by Pivot Walk. Despite their simplicity, index tests are likely to cause branch mispredictions because the ordering of faces is close random. Finally, there could also be problems due to cache effects, since adjacent faces need not be local in memory, meaning that face accesses are more likely to result in cache misses. To test this we did some timing on very small triangulations, however we did not notice any significant differences. Ultimately it seems probable that the final explanation is a combination of the above.

For our optimised code, we found that unrolling the loops and index tests (and some other small optimisations we do not detail here) resulted in a significant speed increase for Remembering Walk. Interestingly, the same optimisations appeared to do very little to improve Pivot Walk. This could be because compiler-level optimisations are more easily applied to the simpler Remembering Walk. It could also be simply that Pivot Walk does not play as nicely with processor branch prediction and pre-loading as Remembering Walk does.

6.4 Conclusions

We have presented a series of novel results on the Visibility Walk graph and also demonstrated that it is possible to design a walk that only requires approximately one orientation per face, without significantly increasing the number of faces visited. Despite the fact that our algorithm did not result in a speed up for point location in Cgal (with the EPIC kernel), we believe that our results could be of interest in situations where orientations may be more expensive. This situation could arise with different kernel representations or on highly degenerate data. In addition, we believe that the techniques presented here open up new possibilities for the construction of new walking algorithms on the walk graph, with plenty of scope for new optimisations and different walking algorithms remaining for consideration in the future.

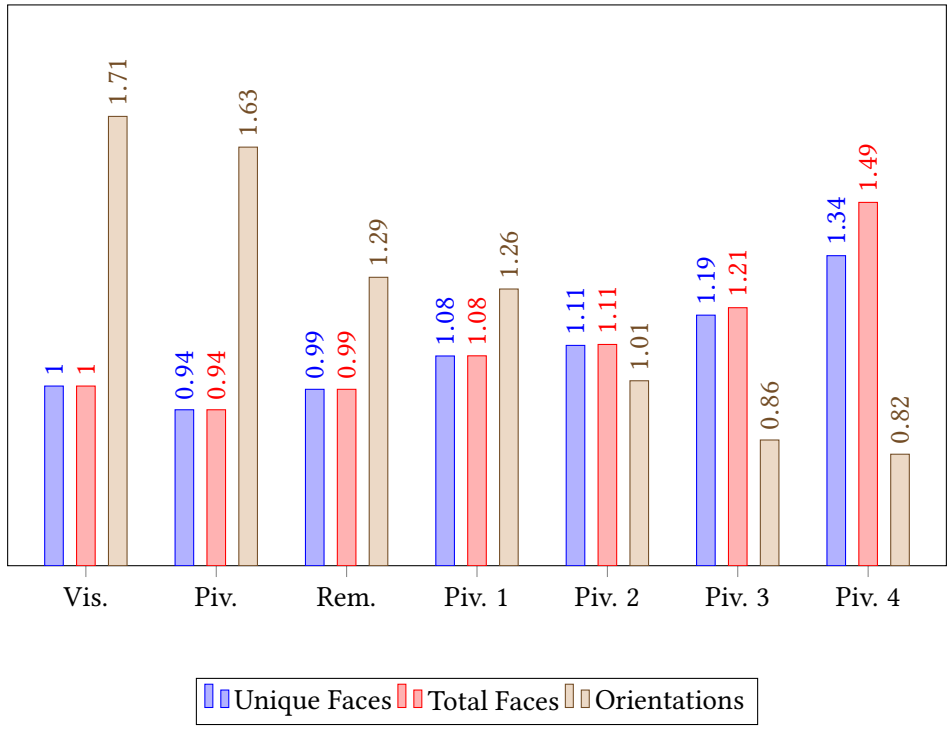


Figure 6.3 – A comparison of walking strategies applied to uniformly random points in a square. The first bar in each case gives the number of unique faces visited by each walk; the second gives the total number of faces visited, and the final bar gives the average number of orientations performed per (unique) face. All face counts are normalised by the number of faces visited by Remembering Visibility Walk.

Part II

Extrema in geometric structures

Chapter 7

Extremes in random tessellations

In this chapter we introduce the problem of computing extremes for characteristics of random tessellations. We begin by giving a brief introduction to extreme value theory, followed by a summary of results in this area which apply to the study of tessellations. Notably, whilst many results for moments and limit theorems have been obtained for properties of cells such as the *typical* cell (which is defined below in Equation (7.2)), very few results giving the extremes of geometric characteristics of cells in random tessellations have been given. Configurations of random lines have found many uses in a diverse array of up-and coming fields such as high dimensional estimation, dimension reduction [98, 110], locality consistent hashing [57], compressed sensing [97] and also in the study of natural phenomena such as the trajectories of particles in bubble chambers [60]. In particular, in the context of compressed sensing, the authors point to a lack of results extremal properties for line tessellations [97]. We are optimistic that our techniques may be extended to prove even stronger results in these cases.

Charactaristics of random tessellations

The results in this part will be focussed on properties of random tessellations. Whilst we focus on the particular cases of Poisson-Voronoi, Poisson-Delaunay and the Poisson hyperplane tessellation, applications to more general tessellations may be kept in mind. We denote a general random tessellation as $\mathfrak{m}$ which represents the collection of (convex) cells in the tessellation. We associaite with the cells of each tessellation a function, $z\colon \mathcal{K}_d \to \mathbf{R}^d$ which deterministically maps a cell to its *nucleus* which satisifies $z(C + x) \equiv z(C)$ for any $x \in \mathbf{R}^d$.[1] We note that an appropriate function $z(\cdot)$ may be chosen depending on the specific problem in question to simplify calculations.

When a tessellation $\mathfrak{m}$ is *stationary*, so that its distribution is invariant under translations, we may associate with it a fixed *intensity*, $\gamma_{\mathfrak{m}}$. In particular, for a set $\mathbf{W} \in \mathcal{B}(\mathbf{R}^d)$

[1] The traditional use of z in this context is due to the German word 'zentrum', meaning 'centre'.

satisfying $0 < \lambda_d(\mathbf{W}) < \infty$, we define the intensity of the tessellation as,

$$\gamma_{\mathfrak{m}} := \frac{1}{\lambda_d(\mathbf{W})} \cdot \mathbb{E}\left[\left| \left\{ C \in \mathfrak{m} : z(C) \in \mathbf{W} \right\} \right| \right]. \tag{7.1}$$

Thus in particular, the intensity gives the expected number of cells whose *nucleii* fall into a region of unit area.

Typical cell Another standard object of study in random tessellations is the *typical cell*, denoted $\mathscr{C}$, which is a random polytope obtained by averaging over all cells in the tessellation. Its distribution is characterised by

$$\mathbb{E}[f(\mathscr{C})] := \frac{1}{\lambda_d(\mathbf{W})} \cdot \mathbb{E}\left[\sum_{\substack{C \in \mathfrak{m}, \\ z(C) \in \mathbf{W}}} f(C - z(C)) \right], \tag{7.2}$$

For all bounded and measurable $f : \mathcal{K}_d \to \mathbf{R}$, where $\mathcal{K}_d$ represents the set of compact convex bodies in $\mathbf{R}^d$. The tools of stochastic geometry have enabled great progress to be made involving the properties of the typical cell, including moments and limit theorems [25, 94, 104]. Much less is known about properties of maxima and minima over all cells, to which we may apply techniques from extreme value theory. This will be the subject of the next section.

7.1 Extreme value theory

Given a sequence $\{\xi_i\}_{i \geq 1}$ of random variables, the field of *extreme value theory* in a basic sense is concerned with the study of the limiting distribution of the maximum,

$$M_n := \max\left\{ \xi_1, \xi_2, \ldots, \xi_n \right\}$$

as n tends to infinity. Perhaps the most celebrated result in this field is the result by Fischer, Tippett and Gnedenko [59] which gives a classification of the possible limiting distributions of M_n when the ξ_i are independent and identically distributed (iid) random variables.

Theorem 7.1 (Fisher-Tippett-Gnedenko)**.** *Suppose there exist $a_n > 0$, $b_n \in \mathbf{R}, n \geq 1$ such that there exists a non-degenerate limiting distribution, G,*

$$\mathbb{P}(a_n(M_n - b_n) \leq t) \xrightarrow{d} G(t).$$

Then G must be one of three types, $G(x) := H(ax + b)$ for $a > 0, b \in reals$.

I. $H(x) = \exp(-e^{-x})$,

II. $H(x) = \begin{cases} \exp(-x^{-\alpha}) & if\, x > 0 \\ 0 & if\, x \leq 0 \end{cases}$,

III. $H(x) = \begin{cases} \exp(-(-x)^{\alpha}) & if\, x \leq 0 \\ 1 & if\, x > 0 \end{cases}$

Proof omitted. □

Remark 7.2. The three classes given in Theorem 7.1 are known respectively as the Gumbel, Fréchet and Weibull distributions.

When the sequence under consideration is not iid, one may naturally ask whether or not such a classification exists under more general conditions. This question is indeed answered in the affirmative, with one possible set of conditions being given by Leadbetter [83], in which a weak mixing condition, $D(v_\rho)$ and a 'local' condition, $D'(v_\rho)$ is shown to imply the same classification given by Theorem 7.1 for stationary sequences (where v_ρ is a *threshold* depending on the sequence in question.) In Leadbetter [84] it is shown that if the condition $D(v_\rho)$ is satisfied without the locality condition $D'(v_\rho)$, then one still obtains a limiting distribution of the same *type* (in the sense of Theorem 7.1), but not necessarily with the same normalisation constants. Given slightly more extraneous conditions, it may be shown that the omission of the locality conditions only alters the normalisation by a fixed constant when compared with the iid setting [84]. This constant is known as the *extremal index*, often denoted θ. It is known that $\theta \in (0, 1]$ and that it may be be interpreted as the inverse of the mean number of *exceedants* that appear in the region about a cell which passes the given threshold. Studying and estimating the extremal index in stationary sequences has become a field in its own right [54, 102, 111].

In this work, we are interested in a more general problem which still shares many properties with the case of stochastic sequences (or random fields.) In particular, we are interested in studying the maximum of a *geometric characteristic*, $f : \mathcal{K}_d \to \mathbf{R}$ which maps a cell in a tessellation to a property of the cell (such as the area or circumradius) applied to the cells of a tessellation. Studying extremes in this context is difficult due to a few key factors that separate the study of spatial tessellations from the better known cases of random sequences and random fields [41, 85, 101]. Firstly, cells in tessellations can exhibit complicated dependence due, for example, to cells having non-deterministic sizes and numbers of neighbours. Other difficulties include the handling of *boundary effects*, in which the behaviour of the tessellation close to the edge of the 'window' may require special treatment. Finally, the geometric characteristics we consider may not depend on a pre-determined number of lines or points of the process.

The first results relating to this work appear to be those by Penrose [96], in which the author studies the limiting distribution of the maximum and minimum degrees of a class of random geometric graphs. Schulte and Thäle [105] also study the order statistics of functionals of k points of a Poisson point process, a process known as computing *U-statistics*.

Extremes in tessellations

It seems that the first authors to specifically investigate the extremes of cells of tessellations were Calka and Chenavier [26], who give the limiting distribution for the maximum and minimum inradius and circumradius of cells in a random Voronoi tessellation observed in a *window*. In particular, they consider a Voronoi tessellation constructed on a homogeneous Poisson process in $\mathbf{R}^2$, and then study the limiting distributions for cells whose *nucleus* falls within a compact convex *window* $\mathbf{W}_\rho := \rho^{1/2}\mathbf{W}$ in the limit as $\rho \to \infty$.

This work was later extended by Chenavier [30], [29] who proved a generalised theorem to compute the limiting distribution for the extremes of characteristics of cells in tessellations $\mathbf{R}^d$ under certain conditions; the first of these conditions is a *Typical Cell Property*, which involves computing the mean of the geometric characteristic on the typical cell. The second is the *Finite Range Condition*, which is a mixing condition analogous to the condition $D(v_\rho)$ of Leadbetter. The final condition is a *Local Correlation Condition*, which bounds the probability of two exceedances being close in a similar vein to that Leadbtter's condition $D'(v_\rho)$. When these three conditions are satisfied, the precise limiting distribution of the order statistics is given. This general theorem was obtained by using Poisson approximation by the *Chen-Stein* method, which gives convergence in distribution of a sum to a Poisson random variable under conditions on the dependency graph [96]. In the same paper, Chenavier [30] also gives several applications of this main theorem to both the Poisson-Delaunay and Poisson-Voronoi tessellations, including the inradius, the circumradius, the area, the volume of the Voronoi flower and the distance to the farthest neighbour.

7.2 Contributions

In Chapter 8, we compute the limiting distribution for the maximum and minimum inradius for a Poisson line tessellation. The results that we obtain can not be obtained through the application of the general theorem by Chenavier [30]. Whilst the mixing property is satisfied (see [104, Theorem 10.5.3]), the required 'finite range condition' is not. The difficulty comes from the fact that even cells which are arbitrarily spatially separated can depend on the same line. Our result instead follows from the combination of a Poisson approximation theorem generalised from a work by Henze [67], and an application of a theorem by Schulte and Thäle [105].

Previous results relating to the study of Poisson line tessellation includes limit theorems [66] and mean values calculations [89, 90] for various geometrical characteristics of the typical cell. A long standing conjecture due to D.G. Kendall about the asymptotic shape of large typical cell is proved in [69]. The shape of small cells is also considered in [4] in the particular setting of rectangular Poisson line tessellation.

CHAPTER 8

Extrema in the Poisson line tessellation

In this chapter we consider extremes for the inradius of cells in a Poisson line tessellation. We begin by giving a formal construction of the tessellation and a description of our results. This will then be followed by a collection of preliminaries which we will require throughout the rest of the chapter.[1]

8.1 The Poisson line tessellation

Let $\mathcal{A}$ denote the set of affine lines in $\mathbf{R}^2$ which do not pass through the origin 0. For $t > 0$ and $u \in \mathbf{S}$, each line in $\mathcal{A}$ has a unique representation

$$H(u,t) := \left\{ x \in \mathbf{R}^2,\ \langle x, u\rangle = t \right\},$$

where we recall that $\mathbf{S}$ is the unit sphere in $\mathbf{R}^2$. Given a fixed $\gamma \in \mathbf{R}_+$, we construct the stationary and isotropic Poisson process $\hat{\mathbf{X}}$ with intensity γ on the space of affine lines, $\mathcal{A}$ with intensity measure given by

$$\mu(A) := 2\gamma \int_{\mathbf{S}} \int_{\mathbf{R}_+} \mathbb{1}_{H(u,r)\in A}\, dr\sigma(du), \tag{8.1}$$

for any Borel subset $A \in \mathcal{B}(\mathcal{A})$, and where $\sigma(\cdot)$ denotes the uniform measure on $\mathbf{S}$ with normalization $\sigma(\mathbf{S}) = 2\pi$. From now on, we assume without loss of generality that the intensity of $\hat{\mathbf{X}}$ is $\gamma := \pi$. The *Poisson line tessellation*, denoted $\mathfrak{m}_{\text{PHT}}$ is now defined to be the set of closures of the connected components of $\mathbf{R}^2 \setminus \hat{\mathbf{X}}$. An example realisation of the Poisson line tessellation truncated to a window is given in Figure 8.1. With each cell $C \in \mathfrak{m}_{\text{PHT}}$, we associate the nucleus $z(C)$ as follows. Almost surely, for each cell C, there

[1] This chapter was written in collaboration with Nicolas Chenavier, and is currently in the process of being prepared for submission.

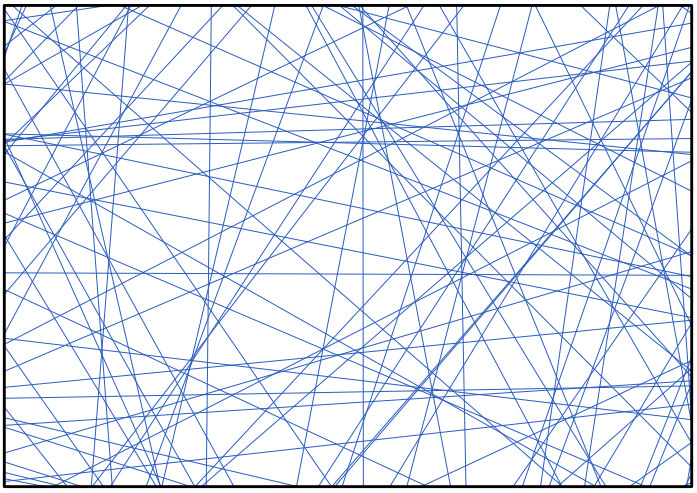


Figure 8.1 – A realisation of the Poisson line tessellation truncated to a window.

exists a unique inball in C which intersects the boundary of C at three points. The point $z(C)$ is then defined as the centre of this ball and we write $R(C)$ to denote its radius.

The problem We are interested in the case where only a part of the tessellation is observed in the window $\mathbf{W}_\rho := B(0, \pi^{-1/2}\rho^{1/2})$, $\rho > 0$, where we recall that $B(z,r)$ denotes the (closed) disc centred at $z \in \mathbf{R}^2$ with radius $r \in \mathbf{R}_+$. Let $f\colon \mathcal{K}_2 \to \mathbf{R}$ be a translation invariant measurable function, so that $f(C+x) = f(C)$ for all $C \in \mathcal{K}_2$ and $x \in \mathbf{R}^2$. We seek to investigate the asymptotic behaviours of the order statistics for the image of f given the cells of the tessellation whose incentres lie within the window, $f(C \in \mathfrak{m}_{\text{PHT}} : z(C) \in \mathbf{W}_\rho)$, in the limit as ρ goes to infinity.

In this work we investigate the particular case where $f(C) := R(C)$ is the inradius of any cell $C \in \mathfrak{m}_{\text{PHT}}$. In particular, we give the asymptotic behaviour of $m_{\mathbf{W}_\rho}[r]$ and $M_{\mathbf{W}_\rho}[r]$, $r \geq 1$, which we use respectively to denote the inradii of the r*th* smallest and the r*th* largest inballs. Thus for $r = 1$ we have,

$$m_{\mathbf{W}_\rho}[1] = \min_{\substack{C\in\mathfrak{m}_{\text{PHT}},\\ z(C)\in\mathbf{W}_\rho}} R(C) \quad \text{and} \quad M_{\mathbf{W}_\rho}[1] = \max_{\substack{C\in\mathfrak{m}_{\text{PHT}},\\ z(C)\in\mathbf{W}_\rho}} R(C).$$

The asymptotic behaviours of $m_{\mathbf{W}_\rho}[r]$ and $M_{\mathbf{W}_\rho}[r]$ are given in our following main theorem.

Theorem 8.1. *Let $\mathfrak{m}_{\text{PHT}}$ be a stationary, isotropic Poisson line tessellation in $\mathbf{R}^2$ with intensity π and let $r \geq 1$ be fixed. Then,*

(a) For any $t \geq 0$,

$$\mathbb{P}\left(m_{\mathbf{W}_\rho}[r] \geq (2\pi^2\rho)^{-1}t\right) \underset{\rho\to\infty}{\longrightarrow} e^{-t}\sum_{k=0}^{r-1}\frac{t^k}{k!}.$$

(b) For any $t \in \mathbf{R}$,

$$\mathbb{P}\left(M_{\mathbf{W}_\rho}[r] \leq \frac{1}{2\pi}(\log(\rho) + t)\right) \underset{\rho\to\infty}{\longrightarrow} e^{-e^{-t}} \sum_{k=0}^{r-1} \frac{(e^{-t})^k}{k!}.$$

The limit distributions are of type II and type III in Theorem 7.1, meaning that $m_{\mathbf{W}_\rho}[1]$ and $M_{\mathbf{W}_\rho}[1]$ belong to the domains of attraction of Weibull and Gumbel distributions respectively. The techniques we employ to investigate the asymptotic behaviours of $m_{\mathbf{W}_\rho}[r]$ and $M_{\mathbf{W}_\rho}[r]$ are both different. For the case of the smallest inballs, we show that asymptotically $m_{\mathbf{W}_\rho}[r]$ has the same behaviour as the rth smallest value associated with a U-statistic. In particular, this allows us to apply the main theorem in Schulte and Thäle [105]. The order statistics for the largest inballs are more delicate since the problem cannot be formulated as a U-statistic. Our idea is to extend a result due to Henze [67] which will allow us to reduce the problem to studying the *finite dimensional distributions*.

Layout of chapter

The remainder of this chapter is organised as follows. We begin by Section 8.2, in which we give the notation and key concepts required throughout the rest of the chapter. In Section 8.3, we provide the asymptotic behaviour of $m_{\mathbf{W}_\rho}[r]$. As a corollary, we shall show that the cells which have a small radius are triangles with high probability. In Section 8.4, we establish some technical lemmas which will be used to derive the asymptotic behaviour of $M_{\mathbf{W}_\rho}[r]$. We conclude the proof for $M_{\mathbf{W}_\rho}[r]$ in Section 8.5.

8.2 Preliminaries

In this section, we outline our notation and introduce the concept of the *typical cell* of a tessellation. Let $C \in \mathcal{K}_2$ be a convex body in $\mathbf{R}^2$. If there exists a unique inball in C, we denote it by $B(C)$. The inradius and the incentre of C, defined as the radius and the centre of $B(C)$, are denoted by $R(C)$ and $z(C)$ respectively. Let E be a measurable set and $K \geq 1$. For any K-tuple of points $x_1, \ldots, x_K \in E$, $K \geq 1$, we write $x_{1:K} := (x_1, \ldots, x_K)$. By $E^K_{\neq}$, we mean the set of K-tuple of points $x_{1:K}$ such that $x_i \neq x_j$ for all $1 \leq i \neq j \leq K$. In addition, if ν is a measure, we write $\nu(x_{1:K}) := \nu(dx_1) \cdots \nu(dx_K)$. We shall occasionally omit the lower bounds in the ranges of sums and unions, and the arguments of functions when they are clear from context. We shall use c to signify a universal positive constant not depending on ρ but which may depend on other quantities. When required we assume ρ is sufficiently large to avoid trivialities. For any pair of functions $f, g : \mathbf{R} \to \mathbf{R}$ we write

$$f(\rho) \underset{\rho\to\infty}{\sim} g(\rho) \Longleftrightarrow f(\rho)/g(\rho) \underset{\rho\to\infty}{\longrightarrow} 1$$

For any line $H \in \mathcal{A}$, we denote by H^+ the half-plane delimited by H and containing 0. In particular, we have

$$H^+(u,t) := \left\{ x \in \mathbf{R}^2,\ \langle x, u\rangle \leq t \right\}$$

for any $t > 0$ and $u \in \mathbf{S}$. Given three lines $H_{1:3} \in \mathcal{A}^3_{\neq}$, we denote by $\triangle(H_{1:3})$ the unique triangle that can be formed by the intersection of the halfspaces induced by the lines H_1, H_2 and H_3. In the same spirit, we denote by $B(H_{1:3})$, $R(H_{1:3})$ and $z(H_{1:3})$ the inball, the inradius and the incenter of $\triangle(H_{1:3})$ respectively. For any $z \in \mathbf{R}^2$ and $r \geq 0$, we denote by $B(z,r)$ the (closed) ball centred in z with radius r. For any $B \in \mathcal{B}(\mathbf{R}^2)$, we denote by $\mathcal{A}(B) \subset \mathcal{A}$, the set $\mathcal{A}(B) := \{H \in \mathcal{A}, H \cap B \neq \varnothing\}$.

The expected number of lines intersecting a region We shall often need to compute the expected number of lines intersecting a region in $\mathbf{R}^2$. To do this, we introduce the function $\phi\colon \mathcal{B}(\mathbf{R}^2) \to \mathbf{R}_+$ as

$$\phi(B) := \mu(\mathcal{A}(B)) = \int_{\mathcal{A}(B)} \mathbb{1}_{H\cap B\neq\varnothing}\, \mu(dH) = \mathbb{E}\left[\#\left\{ H \in \hat{\mathbf{X}}, H \cap B \neq \varnothing \right\}\right]. \tag{8.2}$$

Remark 8.2. The function ϕ is easily evaluated for compact convex regions $B \in \mathcal{B}(\mathbf{R}^2)$. In particular, we may directly apply the *Crofton formula* (see Schneider and Weil [104, Theorem 5.1.1]) to prove that $\phi(B) \equiv \ell(B)$ where $\ell(B)$ is the perimeter of B.

Remark 8.3. Since $\hat{\mathbf{X}}$ is a Poisson process on $\mathcal{A}$ we have for any $B \in \mathcal{B}(\mathcal{A})$,

$$\mathbb{P}\left(\hat{\mathbf{X}} \cap B = \varnothing\right) = \mathbb{P}\left(\#\left\{ \hat{\mathbf{X}} \cap \mathcal{A}(B) \right\} = 0\right) = \exp(-\phi(B)).$$

An explicit representation of the typical cell The typical cell of a Poisson line tessellation, as defined in (7.2), can be made explicit in the following sense. For any measurable function $f\colon \mathcal{K}_2 \to \mathbf{R}$, we have by Schneider and Weil [104, Theorem 10.4.6],

$$\mathbb{E}[f(\mathscr{C})] = \frac{1}{24\pi} \int_0^\infty \int_{\mathbf{S}^3} \mathbb{E}\left[f\left(C\left(\hat{\mathbf{X}}, u_{1:3}, r\right)\right)\right] e^{-2\pi r} a(u_{1:3}) \sigma(du_{1:3}) dr, \tag{8.3}$$

where

$$C\left(\hat{\mathbf{X}}, u_{1:3}, r\right) := \bigcap_{H\in\hat{\mathbf{X}}\cap(\mathcal{A}(B(0,r)))^c} \left\{ H^+ \cap \bigcap_{j=0}^{3} H^+(u_j, r) \right\} \tag{8.4}$$

given that $a(u_{1:3})$ is the area of the triangle with vertices u_1, u_2 and u_3. In particular, when $f(C) = R(C)$ is the inradius of any cell $C \in \mathfrak{m}_{\text{PHT}}$, we have

$$\mathbb{P}\left(R(\mathscr{C}) \leq v\right) = 1 - e^{-2\pi v}, \tag{8.5}$$

for all $v \geq 0$.

8.3 Order statistics for the small inballs

In this section we prove Theorem 8.1 Part (a). Our result will follow by applying a result by Schulte and Thäle [105] relating to the order statistics of functions of triples of lines in the process. Due to technicalities dealing with border effects, we are required to prove the stated result in several steps. The first of these will be to consider the inballs for

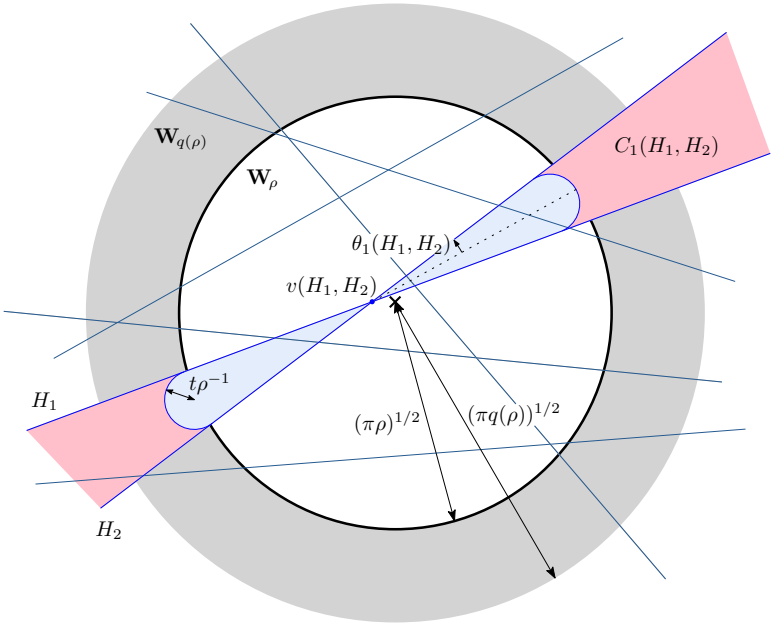


Figure 8.2 – Illustration for the construction in the proof of Lemma 8.7 Part (c). We trace the lines H_1 and H_2 and consider the double cone $C_1(H_1, H_2)$. The convex region we integrate over, given by $E_1(H_1, H_2)$, is equivalent by the shaded 'capped' region between H_1 and H_2.

all triples of lines in the process whose inballs lie within the window, and such that the triangles formed by the three lines intersecting the inball lie *within* the window *plus* an extra 'ring'. This is dealt with by Proposition 8.6. We then argue that this is sufficient to bound the inradius of all *cells* in the tessellation, $\mathfrak{m}_{\text{PHT}}$ such that the triangle formed by the lines intersecting the inball is fully contained within the window plus the same extra ring. Finally, we show that this result is asymptotically equivalent to considering all cells in the mosaic $\mathfrak{m}_{\text{PHT}}$ whose inballs are contained in the window. This last step will follow by proving that the cells minimising the inradius are triangles.

The smallest inradii of triangles included in $\mathbf{W}_\rho$

We begin by recalling a result due to Schulte and Thäle [105], which we re-write to better suit our context. Let $g\colon \mathcal{A}^3 \to \mathbf{R}$ be a measurable function which is symmetric up to permutations of the arguments, and consider a function $q(\rho)$ satisfying the two following properties,

$$q(\rho) \cdot \log \rho \cdot \rho^{-2} \underset{\rho\to\infty}{\longrightarrow} 0 \qquad \text{and} \qquad \pi\left(q(\rho)^{1/2} - \rho^{1/2}\right) - \log \rho \underset{\rho\to\infty}{\longrightarrow} +\infty. \tag{8.6}$$

The function $q(\cdot)$ is used to add an extra 'band' around the window, $\mathbf{W}_\rho$ which contains the nuclei of the cells, and is required in order to deal with border effects. The reader may

assume $q(\rho) := \rho^{4/3}$ in the remainder of the chapter. We now define the set,

$$\left\{ g(H_1,H_2,H_3) : z(H_1,H_2,H_3) \in \mathbf{W}_\rho,\ \triangle(H_1,H_2,H_3) \subset \mathbf{W}_{q(\rho)},\ \{H_1,H_2,H_3\} \subset \hat{\mathbf{X}} \right\}$$

which almost surely defines a locally finite subset of $\mathbf{R}_+$. We define $\overset{\triangle}{m}_{g,\mathbf{W}_\rho}[r]$ to be the r*th* smallest element of this set, for $r \geq 1$. Furthermore, we define for constants $a, t \geq 0$,

$$\alpha_\rho^{(g)}(t) := \frac{1}{6} \int_{\mathcal{A}^3} \mathbb{1}_{g(H_{1:3}) < t\rho^{-a}}\, \mathbb{1}_{z(H_{1:3}) \in \mathbf{W}_\rho}\, \mathbb{1}_{\triangle(H_{1:3}) \subset \mathbf{W}_{q(\rho)}}\, \mu(dH_{1:3}), \tag{8.7a}$$

$$r_{\rho,1}^{(g)}(t) := \int_{\mathcal{A}} \left(\int_{\mathcal{A}^2} \mathbb{1}_{g(H_{1:3}) < t\rho^{-a}}\, \mathbb{1}_{z(H_{1:3}) \in \mathbf{W}_\rho}\, \mathbb{1}_{\triangle(H_{1:3}) \subset \mathbf{W}_{q(\rho)}}\, \mu(dH_{2:3}) \right)^2 \mu(dH_1), \tag{8.7b}$$

$$r_{\rho,2}^{(g)}(t) := \int_{\mathcal{A}^2} \left(\int_{\mathcal{A}} \mathbb{1}_{g(H_{1:3}) < t\rho^{-a}}\, \mathbb{1}_{z(H_{1:3}) \in \mathbf{W}_\rho}\, \mathbb{1}_{\triangle(H_{1:3}) \subset \mathbf{W}_{q(\rho)}}\, \mu(dH_3) \right)^2 \mu(dH_{1:2}). \tag{8.7c}$$

Theorem 8.4. *(Schulte and Thäle) Let $t \geq 0$ be fixed and suppose that $\alpha_\rho^{(g)}(t)$ converges to a positive constant, $\alpha \cdot t^\beta$, for some fixed $\alpha, \beta > 0$. Suppose in addition that*

$$r_{\rho,1}^{(g)}(t) \underset{\rho\to\infty}{\longrightarrow} 0 \quad \textit{and} \quad r_{\rho,2}^{(g)}(t) \underset{\rho\to\infty}{\longrightarrow} 0.$$

Then,

$$\mathbb{P}\left(\overset{\triangle}{m}_{g,\mathbf{W}_\rho}[r] \geq t\rho^{-a} \right) \underset{\rho\to\infty}{\longrightarrow} e^{-\alpha t^\beta} \sum_{k=0}^{r-1} \frac{\left(\alpha t^\beta\right)^k}{k!}.$$

Proof omitted. □

Remark 8.5. Theorem 8.4 is stated in Schulte and Thäle [105] in another context. Firstly, Schulte and Thäle [105, Theorem 1.1] concerns any stationary Poisson point process $\mathbf{X}$ in any Borel measurable space $\mathbf{E}$ and any symmetric measurable function $f\colon \mathbf{E}^k \to \mathbf{R}_+$. In addition, the original theorem is stated for a Poisson point process with intensity going to infinity. By scaling invariance, we have re-written this result for a fixed intensity (equal to π) and for a window $\mathbf{W}_{q(\rho)} = B(0, \pi^{-1/2} q(\rho)^{1/2})$ with ρ going to infinity. Finally, it is easy to show that the theorem due to Schulte and Thäle remains true when we consider the functional $f(H_1,H_2,H_3)\mathbb{1}_{z(H_1,H_2,H_3)\in\mathbf{W}_\rho}$ instead of $f(H_1,H_2,H_3)$.

In our particular application, we are interested in the case where when

$$g(H_1,H_2,H_3) := R(H_1,H_2,H_3).$$

is the inradius of $\triangle(H_{1:3})$. We shall write $\overset{\triangle}{m}_{\mathbf{W}_\rho}[r] := \overset{\triangle}{m}_{R,\mathbf{W}_\rho}[r]$ from now on. In particular, $\overset{\triangle}{m}_{\mathbf{W}_\rho}[r]$ is the r*th* smallest inradius over all triangles generated by triples of lines in $\hat{\mathbf{X}}$ which are completely contained in $\mathbf{W}_\rho$. Its asymptotic behaviour is dealt with in the following propostion.

Proposition 8.6. *For fixed $r \geq 1$ and $t \geq 0$,*

$$\mathbb{P}\left(\overset{\triangle}{m}_{\mathbf{W}_\rho}[r] \geq (2\pi^2\rho)^{-1}t\right) \underset{\rho\to\infty}{\longrightarrow} e^{-t}\sum_{k=0}^{r-1}\frac{t^k}{k!}.$$

The proof of Proposition 8.6 will follow directly from Lemma 8.7, and Theorem 8.4, taking $g := R$ and $a := 1$. The proof of Lemma 8.7 itself is split into three cases, each of which deals with the convergence for one of the required functions.

Lemma 8.7.

(a) $\alpha_\rho^{(R)}(t) \underset{\rho\to\infty}{\longrightarrow} 2\pi^2 t$,

(b) $r_{\rho,1}^{(R)}(t) \underset{\rho\to\infty}{\longrightarrow} 0$,

(c) $r_{\rho,1}^{(R)}(t) \underset{\rho\to\infty}{\longrightarrow} 0$.

Proof of Lemma 8.7 Part (a). Let $t \geq 0$ be fixed. We apply Theorem 8.4 for $g := R$ and $a := 1$. First, we compute the quantity $\alpha_\rho(t) := \alpha_\rho^{(R)}(t)$ as defined in (8.7a). From the Blaschke-Petkantschin formula (Theorem 2.24)[2], we have

$$\begin{aligned}
\alpha_\rho(t) &= \frac{1}{24}\int_{\mathbf{R}^2}\int_0^\infty\int_{\mathbf{S}^3} a(u_{1:3})\mathbb{1}_{z\in\mathbf{W}_\rho}\,\mathbb{1}_{z+r\triangle(H(u_1),H(u_2),H(u_3))\subset\mathbf{W}_{q(\rho)}}\,\mathbb{1}_{r<\rho^{-1}t}\,\sigma(du_{1:3})drdz\\
&= \frac{1}{24}\int_{\mathbf{R}^2}\int_0^\infty\int_{\mathbf{S}^3} a(u_{1:3})\mathbb{1}_{z\in\mathbf{W}_1}\,\mathbb{1}_{z+r\rho^{-3/2}\triangle(H(u_1),H(u_2),H(u_3))\subset\mathbf{W}_{q(\rho)/\rho}}\,\mathbb{1}_{r<t}\,\sigma(du_{1:3})drdz\\
&\underset{\rho\to\infty}{\longrightarrow} \frac{1}{24}\int_{\mathbf{R}^2}\int_0^\infty\int_{\mathbf{S}^3} a(u_{1:3})\mathbb{1}_{z\in\mathbf{W}_1}\,\mathbb{1}_{r<t}\,\sigma(du_{1:3})drdz\\
&= 2\pi^2 t
\end{aligned}$$

Where the penultimate line follows by monotone convergence, and since

$$\int_{\mathbf{S}^3} a(u_{1:3})\sigma(du_{1:3}) = 48\pi^2 \qquad \text{and} \qquad \lambda_2(\mathbf{W}_1) = 1.$$

□

Proof of Lemma 8.7 Part (b). Let H_1 be fixed and define

$$G_\rho(H_1) := \int_{\mathcal{A}^2}\mathbb{1}_{R(H_{1:3})<\rho^{-1}t}\,\mathbb{1}_{z(H_{1:3})\in\mathbf{W}_\rho}\,\mathbb{1}_{\triangle(H_{1:3})\subset\mathbf{W}_{q(\rho)}}\,\mu(dH_{2:3}).$$

We shall give a suitable upper bound for $G_\rho(H_1)$. To do this, we let $H_2, H_3 \in \mathcal{A}$ be such that $R(H_{1:3}) < \rho^{-1}t$ and $z(H_{1:3}) \in \mathbf{W}_\rho$. Denoting by $A \oplus B$ the Minkowski sum between two Borel sets $A, B \in \mathcal{B}(\mathbf{R}^2)$ and by $d(x,H)$ the distance between a point $x \in \mathbf{R}^2$ and a

[2]Note that the measure μ_1 as defined in [104] equals $\frac{1}{\pi}\mu$ where μ is given in (8.1).

line $H \in \mathcal{A}$, we have $z(H_{1:3}) \in H_1 \oplus \rho^{-1}tB(0,1)$ since $d(z(H_{1:3}), H_1) = R(H_{1:3})$. Moreover, $d(0,H_1) \leq |z(H_{1:3})| + R(H_{1:3}) \leq \pi^{-1/2}\rho^{1/2} + \rho^{-1}t$. This implies that for ρ large enough

$$\begin{aligned} G_\rho(H_1) &\leq \int_{\mathcal{A}^2} \mathbb{1}_{z(H_{1:3}) \in \mathbf{W}_\rho \cap \left(H_1 \oplus \rho^{-1}tB(0,1)\right)} \mathbb{1}_{d(0,H_1) < \rho^{1/2}} \, \mu(dH_{2:3}) \\ &= \int_{(\mathbf{R}_+^* \times [0,2\pi))^2} \mathbb{1}_{z(H_1, H_2(s_2,\vec{\alpha_2}), H_3(s_3,\vec{\alpha_3})) \in \mathbf{W}_\rho \cap \left(H_1 \oplus \rho^{-1}tB(0,1)\right)} \mathbb{1}_{d(0,H_1) < \rho^{1/2}} \, ds_{2:3} \, d\alpha_{2:3}, \end{aligned} \tag{8.8}$$

where $\vec{\alpha} := (\cos\alpha, \sin\alpha)$ for any $\alpha \in [0,2\pi)$. With each $z \in \mathbf{R}^2$ and $\alpha_2, \alpha_3 \in [0,2\pi)$, we associate the two lines $H_2(s(z,\alpha_2), \vec{\alpha_2})$ and $H_3(s(z,\alpha_3), \vec{\alpha_3})$, where

$$s(z,\alpha) := \left| d(z,H_1) + \cos\alpha z_1 + \sin\alpha z_2 \right|$$

for any $\alpha \in [0,2\pi)$. We also take

$$E := \left\{ (z,\alpha_2,\alpha_3) \in \mathbf{R}^2 \times [0,2\pi)^2 : s(z,\alpha_2)s(z,\alpha_3) = 0 \right\}$$

and let ϕ_{H_1} be the change of variables

$$\begin{aligned} \phi_{H_1} : \mathbf{R}^2 \times [0,2\pi)^2 \setminus E &\longrightarrow (\mathbf{R}_+^* \times [0,2\pi))^2 \\ (z,\alpha_2,\alpha_3) &\longmapsto (s(z,\alpha_2), \alpha_2, s(z,\alpha_3), \alpha_3). \end{aligned}$$

The Jacobian $J\phi_{H_1}(z,\alpha_1,\alpha_2)$ of ϕ_{H_1} is of the form

$$P\left(\tfrac{\partial d(z,H_1)}{\partial z_1}, \tfrac{\partial d(z,H_1)}{\partial z_2}, \cos\alpha_2, \sin\alpha_2, \cos\alpha_3, \sin\alpha_3\right),$$

where P is polynomial with degree 2. Since

$$\left|\tfrac{\partial d(z,H_1)}{\partial z_1}\right| < 1 \qquad \text{and} \qquad \left|\tfrac{\partial d(z,H_1)}{\partial z_2}\right| < 1,$$

it follows that $J\phi_{H_1}(z,\alpha_1,\alpha_2)$ is bounded. From (8.8), we deduce that

$$\begin{aligned} G_\rho(H_1) &\leq \int_{\mathbf{R}^2 \times [0,2\pi)^2 \setminus E} |J\phi_{H_1}(z,\alpha_2,\alpha_3)| \mathbb{1}_{z \in \mathbf{W}_\rho \cap \left(H_1 \oplus \rho^{-1}tB(0,1)\right)} \mathbb{1}_{d(0,H_1) < \rho^{1/2}} \, dz d\alpha_{2:3} \\ &\leq c \cdot \lambda_2\left(\mathbf{W}_\rho \cap \left(H_1 \oplus \rho^{-1}tB(0,1)\right)\right) \mathbb{1}_{d(0,H_1) < \rho^{1/2}} \\ &\leq c \cdot \rho^{-1/2} \mathbb{1}_{d(0,H_1) < \rho^{1/2}}. \end{aligned} \tag{8.9}$$

It follows from (8.1) and (8.9) that

$$\begin{aligned} r_{\rho,1}(t) &= \int_{\mathcal{A}} G_\rho(H_1)^2 \mu(dH_1) \\ &\leq c \cdot \rho^{-1} \int_{\mathcal{A}} \mathbb{1}_{d(0,H_1) < \rho^{1/2}} \, \mu(dH_1) = O\left(\rho^{-1/2}\right). \end{aligned}$$

□

Proof of Lemma 8.7 Part (c). Suppose that H_1 and H_2 intersect at the unique point, $v(H_{1:2})$. The set $H_1 \cup H_2$ then divides $\mathbf{R}^2$ into two double-cones $C_1(H_{1:2})$ and $C_2(H_{1:2})$ (see Figure 8.2.) We write $\theta_1(H_{1:2})$ and $\theta_2(H_{1:2})$ to denote the half-angles of these cones respectively, so that $2(\theta_1(H_{1:2}) + \theta_2(H_{1:2})) = \pi$. In addition, we write

$$E_i(H_{1:2}) := \left\{ H_3 \in \mathcal{A} : z(H_{1:3}) \in \mathbf{W}_\rho \cap C_i(H_{1:2}), \triangle(H_{1:3}) \subset \mathbf{W}_{q(\rho)},\ R(H_{1:3}) < \rho^{-1} t \right\}.$$

We now give a suitable upper bound for $G_\rho(H_1, H_2)$, defined as

$$\begin{aligned} G_\rho(H_1, H_2) &:= \int_{\mathcal{A}} \mathbb{1}_{R(H_{1:3}) < \rho^{-1} t}\, \mathbb{1}_{z(H_{1:3}) \in \mathbf{W}_\rho}\, \mathbb{1}_{\triangle(H_{1:3}) \subset \mathbf{W}_{q(\rho)}}\, \mu(dH_3) \\ &\pounds = \sum_{i=1}^{2} \int_{\mathcal{A}} \mathbb{1}_{E_i(H_{1:2})}\, \mu(dH_3). \end{aligned} \tag{8.10}$$

We observe that each E_i defines back-to-back cones with 'rounded' bases. Since these are both convex region, we can bound the required integral using Crofton's formula, as stated in Remark 8.2. An illustration of the construction for this proof is given in Figure 8.2. In the following, we shall require the following lemma.

Lemma 8.8. *Let $H_3 \in E_i(H_{1:2})$, $1 \le i \le 2$. Then*

1. *$|v(H_{1:2})| \le c \cdot q(\rho)^{1/2}$ for some c,*
2. *$H_3 \cap W_{c \cdot \rho} \neq \varnothing$ for some c,*
3. *$H_3 \cap B\left(v(H_{1:2}), \frac{c \cdot \rho^{-1}}{\sin \theta_i(H_{1:2})}\right) \neq \varnothing$,*

Proof of Lemma 8.8. The first statement comes from the fact that $v(H_{1:2}) \in \mathbf{W}_{q(\rho)}$. The second statement follows from the inequalities,

$$d(0, H_3) \ \le\ \left|z(H_{1:3})\right| + d(z(H_{1:3}), H_3) \ \le\ \pi^{-1/2} \rho^{1/2} + \rho^{-1} t \ \le\ c \cdot \rho^{1/2}.$$

For the third statement, we write

$$d(v(H_{1:2}), H_3) \ \le\ d(z(H_{1:3}), H_3) + |v(H_{1:2}) - z(H_{1:3})| \ \le\ \rho^{-1} t + \frac{R(H_{1:3})}{\sin \theta_i(H_{1:2})} \ \le\ \frac{c \cdot \rho^{-1}}{\sin \theta_i(H_{1:2})}.$$

□

It now follows from (8.10) and Lemma 8.8 that

$$\begin{aligned} G_\rho(H_1, H_2) \le \sum_{i=1}^{2} \int_{\mathcal{A}} & \mathbb{1}_{H_3 \cap \mathbf{W}_{c \cdot \rho} \neq \varnothing}\, \mathbb{1}_{\sin \theta_i(H_{1:2}) \le \rho^{-3/2}}\, \mathbb{1}_{|v(H_{1:2})| \le c \cdot q(\rho)^{1/2}}\, \mu(dH_3) \\ + \int_{\mathcal{A}} & \mathbb{1}_{H_3 \cap B\left(v(H_{1:2}), \frac{c \cdot \rho^{-1}}{\sin \theta_i(H_{1:2})}\right) \neq \varnothing}\, \mathbb{1}_{\sin \theta_i(H_{1:2}) > \rho^{-3/2}}\, \mathbb{1}_{|v(H_{1:2})| \le c \cdot q(\rho)^{1/2}}\, \mu(dH_3). \end{aligned} \tag{8.11}$$

Integrating over H_3 by applying the Crofton formula, (8.2), we get

$$G_\rho(H_1, H_2) \le c \cdot \sum_{i=1}^{2} \left(\rho^{1/2} \mathbb{1}_{\sin \theta_i(H_{1:2}) \le \rho^{-3/2}} + \frac{\rho^{-1}}{\sin \theta_i(H_{1:2})} \mathbb{1}_{\sin \theta_i(H_{1:2}) > \rho^{-3/2}} \right) \mathbb{1}_{|v(H_{1:2})| \le c \cdot q(\rho)^{1/2}}.$$

(8.12)

Putting these steps together gives us

$$\begin{aligned} r_{\rho,2}(t) &= \int_{\mathcal{A}} G_\rho(H_1,H_2)^2 \mu(dH_{1:2}), \\ &\leq c \cdot \int_{\mathcal{A}^2} \left(\rho \mathbb{1}_{\sin\theta(H_{1:2}) \leq \rho^{-3/2}} + \frac{\rho^{-2}}{\sin^2\theta(H_{1:2})} \mathbb{1}_{\sin\theta(H_{1:2}) > \rho^{-3/2}} \right) \mathbb{1}_{|v(H_{1:2})| \leq c \cdot q(\rho)^{1/2}} \, \mu(dH_{1:2}), \end{aligned}$$

where $\theta(H_{1:2}) \in [0, \frac{\pi}{2})$ is the smallest half-angle between two lines H_1 and H_2. We now consider an appropriate change of variables. Given $v = (v_1, v_2) \in \mathbf{R}^2$, $\theta \in [0, \frac{\pi}{2})$ and $\beta \in [0, 2\pi)$, we associate the lines H_1 and H_2 in the following way. We define $L(v_1, v_2, \beta)$ as the line containing $v = (v_1, v_2)$ and orthogonal to $\vec{\beta} = (\cos\beta, \sin\beta)$. We then define H_1 and H_2 as the lines containing $v = (v_1, v_2)$ with angles θ and $-\theta$ with respect to $L(v_1, v_2, \beta)$ respectively. We define

$$\begin{aligned} &\psi \colon \mathbf{R}^2 \times [0, 2\pi) \times [0, \frac{\pi}{2}) \longrightarrow \mathbf{R}_+ \times [0, 2\pi) \times \mathbf{R}_+ \times [0, 2\pi) \\ &\quad (v_1, v_2, \beta, \theta) \longmapsto \left(\left| -\sin(\beta - \theta) v_1 + \cos(\beta - \theta) v_2 \right|, \overline{\beta - \theta}, \left| \sin(\beta + \theta) v_1 + \cos(\beta + \theta) v_2 \right|, \overline{\beta + \theta} \right). \end{aligned}$$

Almost everywhere, ψ is differentiable and we can easily prove that its Jacobian satisfies $|J\psi(v_1, v_2, \beta, \theta)| = 2 \sin 2\theta$ for any differentiable point $(v_1, v_2, \beta, \theta)$. Taking the change of variables as defined above, we deduce from (8.12) that

$$\begin{aligned} r_{\rho,2}(t) &= \int_{\mathcal{A}} G_\rho(H_1,H_2)^2 \mu(dH_{1:2}). \\ &\leq c \cdot \int_{\mathbf{R}^2} \int_0^{2\pi} \int_0^{\frac{\pi}{2}} \sin(2\theta) \left(\rho \mathbb{1}_{\sin\theta \leq \rho^{-3/2}} + \frac{\rho^{-2}}{\sin^2\theta} \mathbb{1}_{\sin\theta > \rho^{-3/2}} \right) \mathbb{1}_{|v| \leq c \cdot q(\rho)^{1/2}} \\ &= O\left(\log\rho \cdot q(\rho) \cdot \rho^{-2} \right). \end{aligned}$$

According to (8.6), the last term converges to 0 as ρ goes to infinity. □

The smallest inradii of cells C such that $z(C) \in \mathbf{W}_\rho$ and $\triangle(C) \subset \mathbf{W}_{q(\rho)}$

For any $r \geq 1$, we denote by $\mathring{m}_{\mathbf{W}_\rho}[r]$ the r*th* smallest value of the inradius over all cells $C \in \mathfrak{m}_{\text{PHT}}$ such that $z(C) \in \mathbf{W}_\rho$ and $\triangle(C) \subset \mathbf{W}_{q(\rho)}$. We observe that almost surely, we have $\mathring{m}_{\mathbf{W}_\rho}[1] = \mathring{m}_{\mathbf{W}_\rho}[1]$. Actually, the following proposition shows that the deviation between $\mathring{m}_{\mathbf{W}_\rho}[r]$ and $\mathring{m}_{\mathbf{W}_\rho}[r]$ is negligible as ρ goes to infinity.

Lemma 8.9. *For any fixed $r \geq 1$,*

$$\mathbb{P}\left(\mathring{m}_{\mathbf{W}_\rho}[r] \neq \mathring{m}_{\mathbf{W}_\rho}[r] \right) \underset{\rho \to \infty}{\longrightarrow} 0.$$

Proof of Lemma 8.9. Let $r \geq 1$ be fixed. Almost surely, there exists a unique triangle induced by $\hat{\mathbf{X}}$ and included in $\mathbf{W}_\rho$, say $\triangle_{\mathbf{W}_\rho}[r]$, such that $R(\triangle_{\mathbf{W}_\rho}[r]) = \mathring{m}_{\mathbf{W}_\rho}[r]$. Besides,

$z(\triangle_{\mathbf{W}_\rho}[r])$ is the incenter of a cell in $\mathfrak{m}_{\mathrm{PHT}}$ if and only if $\hat{\mathbf{X}} \cap B(\triangle_{\mathbf{W}_\rho}[r]) = \varnothing$. Since $\mathring{m}_{\mathbf{W}_\rho}[r] \geq \mathring{m}_{\mathbf{W}_\rho}[r]$, this implies that

$$\begin{aligned}
&\mathbb{P}\left(\mathring{m}_{\mathbf{W}_\rho}[r] \neq \mathring{m}_{\mathbf{W}_\rho}[r]\right) \\
&= \mathbb{P} \bigcup_{1 \leq k \leq r} \left\{ \hat{\mathbf{X}} \cap B(\triangle_{\mathbf{W}_\rho}[k]) \neq \varnothing \right\} \\
&\leq \sum_{k=1}^{r} \left(\mathbb{P}\left(\hat{\mathbf{X}} \cap B(\triangle_{\mathbf{W}_\rho}[k]) \neq \varnothing, R(\triangle_{\mathbf{W}_\rho}[k]) < \rho^{-1+\epsilon}\right) + \mathbb{P}\left(R(\triangle_{\mathbf{W}_\rho}[k]) > \rho^{-1+\epsilon}\right)\right)
\end{aligned} \tag{8.13}$$

for any $\epsilon > 0$. The second term of the series converges to 0 as ρ goes to infinity thanks to Proposition 8.6. For the second term, we obtain from the Slivnyak-Mecke formula (Theorem 2.22)

$$\begin{aligned}
&\mathbb{P}\left(\hat{\mathbf{X}} \cap B(\triangle_{\mathbf{W}_\rho}[k]) \neq \varnothing, R(\triangle_{\mathbf{W}_\rho}[k]) < \rho^{-1+\epsilon}\right) \\
&\leq \mathbb{P}\left(\bigcup_{H_{1:4} \in \hat{\mathbf{X}}^4_{\neq}} \left\{ z(H_{1:3}) \in \mathbf{W}_\rho \right\} \cap \left\{ R(H_{1:3}) < \rho^{-1+\epsilon} \right\} \cap \left\{ H_4 \cap B(z(H_{1:3}), \rho^{-1+\epsilon}) \neq \varnothing \right\} \right) \\
&\leq \mathbb{E}\left[\sum_{H_{1:4} \in \hat{\mathbf{X}}^4_{\neq}} \mathbb{1}_{z(H_{1:3}) \in \mathbf{W}_\rho} \, \mathbb{1}_{R(H_{1:3}) < \rho^{-1+\epsilon}} \, \mathbb{1}_{H_4 \cap B(z(H_{1:3}), \rho^{-1+\epsilon}) \neq \varnothing} \right] \\
&= \int_{\mathcal{A}^4} \mathbb{1}_{z(H_{1:3}) \in \mathbf{W}_\rho} \, \mathbb{1}_{R(H_{1:3}) < \rho^{-1+\epsilon}} \, \mathbb{1}_{H_4 \cap B(z(H_{1:3}), \rho^{-1+\epsilon}) \neq \varnothing} \, \mu(dH_{1:4})
\end{aligned}$$

for any $1 \leq k \leq r$. Applying the Blaschke-Petkantschin formula (Theorem 2.24), we get

$$\begin{aligned}
&\mathbb{P}\left(\hat{\mathbf{X}} \cap B(\triangle_{\mathbf{W}_\rho}[k]) \neq \varnothing, R(\triangle_{\mathbf{W}_\rho}[k]) < \rho^{-1+\epsilon}\right) \\
&\leq c \cdot \int_{\mathbf{W}_\rho} \int_0^{\rho^{-1+\epsilon}} \int_{\mathbf{S}^3} \int_{\mathcal{A}} a(u_{1:3}) \mathbb{1}_{H_4 \cap B(z, \rho^{-1+\epsilon}) \neq \varnothing} \, \mu(dH_4) \sigma(du_{1:3}) dr dz.
\end{aligned}$$

According to (8.2) and (8.2), we have

$$\int_{\mathcal{A}} \mathbb{1}_{H_4 \cap B(z, \rho^{-1+\epsilon}) \neq \varnothing} \, \mu(dH_4) = c \cdot \rho^{-1+\epsilon}$$

for any $z \in \mathbf{R}^2$. Integrating over $z \in \mathbf{W}_\rho$, $r < \rho^{-1+\epsilon}$ and $u_{1:3} \in \mathbf{S}^3$, we obtain

$$\mathbb{P}\left(\hat{\mathbf{X}} \cap B(\triangle_{\mathbf{W}_\rho}[k]) \neq \varnothing, R(\triangle_{\mathbf{W}_\rho}[k]) < \rho^{-1+\epsilon}\right) \leq c \cdot \rho^{-1+\epsilon} \tag{8.14}$$

since $\lambda_2(\mathbf{W}_\rho) = \rho$. Taking $\epsilon < \frac{1}{2}$, we deduce Proposition 8.9 from (8.13) and (8.14). □

As a corollary of Proposition 8.6 and Lemma 8.9, we get the following.

Corollary 8.10. *For any fixed $r \geq 1$ and $t \geq 0$,*

$$\mathbb{P}\left(\mathring{m}_{\mathbf{W}_\rho}[r] \geq (2\pi^2\rho)^{-1} t\right) \underset{\rho \to \infty}{\longrightarrow} e^{-t} \sum_{k=0}^{r-1} \frac{t^k}{k!}.$$

The smallest inradii over all cells with incenter in $\mathbf{W}_\rho$

The shape of cells with small inradius As in the proof of Proposition 8.9, we have that for any $r \geq 1$, there almost surely exists a unique cell in $\mathfrak{m}_{\text{PHT}}$ with incenter in $\mathbf{W}_\rho$, say $C_{\mathbf{W}_\rho}[r]$, such that $R(C_{\mathbf{W}_\rho}[r]) = m_{\mathbf{W}_\rho}[r]$. In this subsection, we are interested in the random variable $n(C_{\mathbf{W}_\rho}[r])$ where, for any (convex) polygon P in $\mathbf{R}^2$, we denote by $n(P)$ the number of vertices of P. We remark that it has been shown that the cell which minimises the circumradius for a Poisson-Voronoi tessellation is a triangle with high probability in Calka and Chenavier [26, Corollary 1]. We demonstrate that the same property holds for cells of a Poisson line tessellation with a small inradius.

Proposition 8.11. *For all $r \geq 1$, we have*

$$\mathbb{P}\left(\bigcap_{1 \leq k \leq r} \Big\{ n(C_{\mathbf{W}_\rho}[k]) = 3 \Big\} \right) \underset{\rho \to \infty}{\longrightarrow} 1.$$

Proof of Proposition 8.11. Let $\epsilon \in \left(0, \frac{1}{2}\right)$ be fixed. For any $1 \leq k \leq r$, we write

$$\begin{aligned}
&\mathbb{P}\left(n(C_{\mathbf{W}_\rho}[k]) \neq 3\right) \\
&= \mathbb{P}\left(n(C_{\mathbf{W}_\rho}[k]) \geq 4, m_{\mathbf{W}_\rho}[k] \geq \rho^{-1+\epsilon}\right) + \mathbb{P}\left(n(C_{\mathbf{W}_\rho}[k]) \geq 4, m_{\mathbf{W}_\rho}[k] < \rho^{-1+\epsilon}\right).
\end{aligned}$$

According to Corollary 8.10 and the fact that $m_{\mathbf{W}_\rho}[k] \leq \mathring{m}_{\mathbf{W}_\rho}[k]$, the first term of the right-hand side converges to 0 as ρ goes to infinity. For the second term, we obtain from (7.2) that

$$\begin{aligned}
&\mathbb{P}\left(n(C_{\mathbf{W}_\rho}[k]) \geq 4, m_{\mathbf{W}_\rho}[k] < \rho^{-1+\epsilon}\right) \\
&= \mathbb{P}\left(\min_{\substack{z(C) \in \mathbf{W}_\rho,\, n(C) \geq 4, \\ C \in \mathfrak{m}_{\text{PHT}}}} R(C) < \rho^{-1+\epsilon} \right) \\
&\leq \mathbb{E}\left[\sum_{\substack{z(C) \in \mathbf{W}_\rho, \\ C \in \mathfrak{m}_{\text{PHT}}}} \mathbb{1}_{R(C) < \rho^{-1+\epsilon}} \mathbb{1}_{n(C) \geq 4} \right] \\
&= \pi\rho \cdot \mathbb{P}\left(R(\mathscr{C}) < \rho^{-1+\epsilon},\ n(\mathscr{C}) \geq 4\right)
\end{aligned} \tag{8.15}$$

We give below an integral representation of

$$\mathbb{P}\left(R(\mathscr{C}) < \rho^{-1+\epsilon},\ n(\mathscr{C}) \geq 4\right).$$

Let $r > 0$ and $u_1, u_2, u_3 \in \mathbf{S}$ be fixed. We additionally use $\triangle(u_{1:3}, r)$ to denote the triangle $\triangle(H(u_1, r), H(u_2, r), H(u_3, r))$. We observe that the random polygon $C(\hat{\mathbf{X}}, u_{1:3}, r)$, as defined in (8.4), imples that

$$n\left(C(\hat{\mathbf{X}}, u_{1:3}, r)\right) \geq 4 \quad \Longleftrightarrow \quad \#\Big\{ \hat{\mathbf{X}} \cap \mathcal{A}\Big(\triangle(u_{1:3}, r) \setminus B(0, r)\Big) \Big\} \neq 0.$$

According to (8.3) and (8.3), this implies that

$$\begin{aligned}&\pi\rho\cdot\mathbb{P}\left(R(\mathscr{C})<\rho^{-1+\epsilon},\ n(\mathscr{C})\geq 4\right)\\&=\frac{\rho}{24}\int_0^{\rho^{-1+\epsilon}}\int_{\mathrm{S}^3}\left(1-e^{-\phi(\triangle(u_{1:3},r)\setminus B(0,r))}\right)e^{-2\pi r}a(u_{1:3})\sigma(du_{1:3})dr\end{aligned}$$

Using the fact that $1-e^{-x}\leq x$ for all $x\in\mathbf{R}$ and

$$\phi\Big(\triangle(u_{1:3},r)\setminus B(0,r)\Big)\quad\leq\quad\phi\Big(\triangle(u_{1:3},r)\Big)\quad=\quad r\cdot\ell\Big(\triangle(u_{1:3},r)\Big),$$

which follows from (8.2); we get

$$\begin{aligned}&\pi\rho\cdot\mathbb{P}\left(R(\mathscr{C})<\rho^{-1+\epsilon},\ n(\mathscr{C})\geq 4\right)\\&\leq\frac{\rho}{24}\int_0^{\rho^{-1+\epsilon}}\int_{\mathrm{S}^3}re^{-2\pi r}\ell(\triangle(u_{1:3}))\sigma(du_{1:3})dr\\&=O\left(\rho^{-1+2\epsilon}\right).\end{aligned}$$

This together with (8.15) shows that

$$\mathbb{P}\left(n(C_{\mathbf{W}_\rho}[k])\geq 4,\ m_{\mathbf{W}_\rho}[k]<\rho^{-1+\epsilon}\right)\underset{\rho\to\infty}{\longrightarrow}0$$

□

Remark 8.12. In [105], Schulte and Thäle conjectured that cells with a small area are triangles with high probability. In the same spirit as the above and using the fact that

$$\mathbb{P}\left(\lambda_2(\mathscr{C})<v\right)\leq\mathbb{P}\left(R(\mathscr{C})<(\pi^{-1}v)^{1/2}\right)$$

for all $v>0$, we can prove that their conjecture is true. In particular, this implies that the cell which minimises the area has area of order ρ^{-2} according to Theorem 2.6 in [105].

Proof of Theorem 8.1, Part (a). According to Proposition 8.10, it is enough to prove the following lemma.

Lemma 8.13. *For any $r\geq 1$,*

$$\mathbb{P}\left(m_{\mathbf{W}_\rho}[r]\neq\mathring{m}_{\mathbf{W}_\rho}[r]\right)\underset{\rho\to\infty}{\longrightarrow}0.\tag{8.16}$$

Proof of Lemma 8.13. Since $m_{\mathbf{W}_\rho}[r]\neq\mathring{m}_{\mathbf{W}_\rho}[r]$ if and only if $\triangle\left(C_{\mathbf{W}_\rho}[k]\right)\cap W_{q(\rho)}\neq\varnothing$ for some $1\leq k\leq r$, we get

$$\mathbb{P}\left(m_{\mathbf{W}_\rho}[r]\neq\mathring{m}_{\mathbf{W}_\rho}[r]\right)\tag{8.17}$$

$$\leq\sum_{k=1}^{r}\left(\mathbb{P}\left(\triangle\left(C_{\mathbf{W}_\rho}[k]\right)\cap W_{q(\rho)}\neq\varnothing,\,n(C_{\mathbf{W}_\rho}[k])=3\right)+\mathbb{P}\Big(\,n(C_{\mathbf{W}_\rho}[k])\neq 3\,)\right).\tag{8.18}$$

According to Proposition 8.11, the second term of the series converges to 0. Moreover, for any $1 \leq k \leq r$, we obtain

$$\begin{aligned}
&\mathbb{P}\left(\triangle\left(C_{\mathbf{W}_\rho}[k]\right) \cap W_{q(\rho)} \neq \varnothing,\, n(C_{\mathbf{W}_\rho}[k]) = 3\right) \\
&\leq \mathbb{P}\left(\bigcup_{H_{1:3} \in \hat{\mathbf{X}}^3_{\neq}} \left\{ z(H_{1:3}) \in \mathbf{W}_\rho \right\} \cap \left\{ \triangle(H_{1:3}) \cap \mathbf{W}_{q(\rho)} \neq \varnothing \right\} \cap \left\{ \triangle(H_{1:3}) \cap \hat{\mathbf{X}} = \varnothing \right\}\right) \\
&\leq \int_{\mathcal{A}^3} \mathbb{P}\left(\triangle(H_{1:3}) \cap \hat{\mathbf{X}} = \varnothing\right) \mathbb{1}_{z(H_{1:3}) \in \mathbf{W}_\rho}\, \mathbb{1}_{\triangle(H_{1:3}) \cap \mathbf{W}_{q(\rho)} \neq \varnothing}\, \mu\,(dH_{1:3}) \\
&\leq \int_{\mathcal{A}^3} e^{-\pi^{-1}\left(q(\rho)^{1/2} - \rho^{1/2}\right)} \mathbb{1}_{z(H_{1:3}) \in \mathbf{W}_\rho}\, \mu\,(dH_{1:3})\,,
\end{aligned}$$

where the second and the third inequalities come from the Slivnyak-Mecke formula and (8.3) respectively. Applying the Blaschke-Petkantschin formula, we get

$$\mathbb{P}\left(\triangle\left(C_{\mathbf{W}_\rho}[k]\right) \cap W_{q(\rho)} \neq \varnothing,\, n(C_{\mathbf{W}_\rho}[k]) = 3\right) \leq \rho \cdot e^{-\pi^{-1}\left(q(\rho)^{1/2} - \rho^{1/2}\right)}.$$

According to (8.6), the last term converges to 0 as ρ goes to infinity. This together with (8.17) proves Lemma 8.13. □

□

8.4 Technical results

In this section, we provide two results which will be required in order to derive the asymptotic behaviour of $M_{\mathbf{W}_\rho}[r]$ for the proof of Theorem 8.1 Part b. The first result is a Poisson approximation theorem and it will allow us to demonstrate that it suffices to investigate a *finite* number of cells. The second result is more technical in nature and provides a bound on the function ϕ for unions of balls.

Poisson approximation

Consider a measurable function $f : \mathcal{K}_2 \to \mathbf{R}$, and a *threshold* v_ρ such that $v_\rho \to \infty$ as $\rho \to \infty$. For any $r \geq 1$, we denote the r*th* largest value of $f(C)$ over all cells $C \in \mathfrak{m}_{\text{PHT}}$ with inradius $z(C) \in \mathbf{W}_\rho$. Besides, we call the cells C in the tessellation $\mathfrak{m}_{\text{PHT}}$ such that $f(C) > v_\rho$ and $z(C) \in \mathbf{W}_\rho$ the *exceedances*. A classic tool in extreme value theory is to estimate the limiting distribution of the number of exceedances by a Poisson random variable. In our case, we achieve this with the following lemma.

Lemma 8.14. *Let $\mathfrak{m}_{\text{PHT}}$ be a Poisson line tessellation embedded in $\mathbf{R}^2$. Let us assume that for any $K \geq 1$,*

$$\mathbb{E}\left[\sum_{\substack{C_{1:K} \in (\mathfrak{m}_{\text{PHT}})^K_{\neq} \\ z(C_{1:K}) \in \mathbf{W}_\rho}} \mathbb{1}_{f(C_{1:K}) > v_\rho}\right] \underset{\rho \to \infty}{\longrightarrow} \tau^K. \tag{8.19}$$

Then

$$\mathbb{P}\left(M_{f,\mathbf{w}_\rho}[r] \leq v_\rho\right) \underset{\rho\to\infty}{\longrightarrow} \sum_{k=0}^{r-1} \frac{\tau^k}{k!} e^{-\tau}.$$

Lemma 8.14 can be generalised for any window $\mathbf{W}_\rho$ and for any tessellation in any dimension. Besides, this is an extension of a lemma due to Henze (see Section 2 in Henze [67]) in the sense that we investigate order statistics and not only a maximum. The same tool was used to provide the asymptotic behaviour for couples of random variables in the particular setting of a Poisson-Voronoi tessellation (see Proposition 2 in Calka and Chenavier [26]). The main difficulty will be to apply Lemma 8.14.

Proof of Lemma 8.14. Let the number of exceedance cells be denoted

$$U(v_\rho) := \sum_{\substack{C\in\mathfrak{m},\\ z(C)\in\mathbf{W}_\rho}} \mathbb{1}_{f(C)>v_\rho},$$

and let $\left\{ {n \atop K} \right\}$ denote the Stirling number of second kind with parameters $1 \leq K \leq n$. Directly applying the assumptions given in the lemma, and using the generating function for $\left\{ {n \atop K} \right\}$ along with a small combinatorial argument, we obtain for all $n \geq 1$,

$$\begin{aligned}
\mathbb{E}\left[U(v_\rho)^n\right] &= \mathbb{E}\left[\sum_{K=1}^{n} \left\{ {n \atop K} \right\} U(v_\rho)\cdot\Big(U(v_\rho)-1\Big)\cdot\Big(U(v_\rho)-2\Big)\cdots\Big(U(v_\rho)-K+1\Big)\right] \\
&= \sum_{K=1}^{n} \left\{ {n \atop K} \right\} \mathbb{E}\left[\sum_{\substack{C_{1:K}\in\mathfrak{m}_{\neq}^{K},\\ z(C_{1:K})\in\mathbf{W}_\rho}} \mathbb{1}_{f(C_{1:K})>v_\rho}\right] \\
&\underset{\rho\to\infty}{\longrightarrow} \sum_{K=1}^{n} \left\{ {n \atop K} \right\} \tau^K \\
&= \mathbb{E}[\mathrm{Po}(\tau)^n].
\end{aligned}$$

Thus by the method of moments, $U(v_\rho)$ converges in distribution to a Poisson distributed random variable with mean τ. We conclude the proof by noting that $M_{f,\mathbf{w}_\rho}[r] \leq v_\rho$ if and only if $U(v_\rho) \leq r-1$. □

A uniform upper bound for ϕ for the union of discs

Let $\phi : \mathcal{B}(\mathbf{R}^2) \to \mathbf{R}$ as in (8.2). In this subsection, we evaluate $\phi(B)$ in the particular setting where $B = \bigcup_{1\leq i\leq K}^{K} B(z_i, r_i)$ is a finite union of balls centred in z_i and with radius r_i, $1 \leq i \leq K$. Closed formulas for $\phi(B)$ could be provided but the ones are not easy to manipulate. We provide below (see Proposition 8.15) some estimates of $\phi\left(\bigcup_{1\leq i\leq K}^{K} B(z_i, r_i)\right)$ with more simple and quasi-optimal lower bounds.

Our bound will follow by splitting the discs into a collection of *connected components*. Suppose we are given a threshold v_ρ such that $v_\rho \to \infty$ as $\rho \to \infty$ and $K \geq 2$ discs $B(z_i, r_i)$,

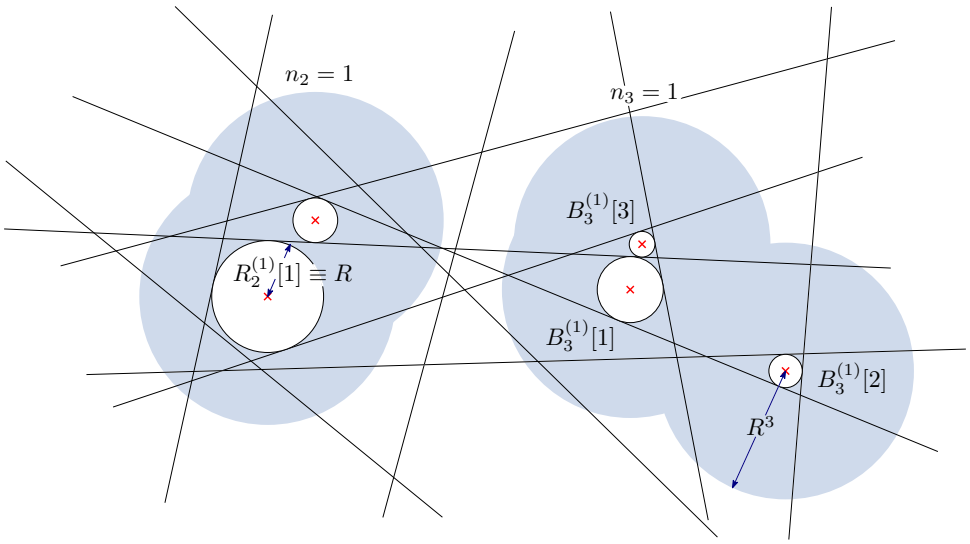


Figure 8.3 – Example connected components for $K = 5$ and $(n_1, \ldots, n_K) = (0, 1, 1, 0, 0)$

satisfying $z_i \in \mathbf{R}^2$, $r_i \in \mathbf{R}_+$ and $r_i > v_\rho$, for all $i = 1, \ldots, K$. We take $R := \max_{1 \le i \le K} r_i$. The connected components are constructed from the graph with vertices $B(z_i, r_i), i = 1, \ldots, K$ and edges

$$B(z_i, r_i) \leftrightarrow B(z_j, r_j) \iff B(z_i, R^3) \cap B(z_j, R^3) \neq \varnothing. \tag{8.20}$$

In the right-hand side, we have chosen radii of the form R^3 to provide a more simple lower bound in Proposition 8.15. The *size* of a component is the number of discs in that component. To refer to these components, we use the following notation which is highlighted for ease of reference.

Notation for connected components

i. For all $k \le K$, write $n_k := n_k(z_{1:K}, r_{1:K})$ to denote the number of connected components of size k. Observe that in particular, $\sum_{k=1}^{K} k \cdot n_k = K$.

ii. Suppose that each component of size k is assigned a unique label $1 \le j \le n_k$. We will then write $B_k^{(j)} := B_k^{(j)}(z_{1:K}, r_{1:K})$, to refer to the union of balls in the j*th* component of size k.

iii. Within a component, we write $B_k^{(j)}[\ell] := B_k^{(j)}(z_{1:K}, r_{1:K})[\ell]$, $1 \le \ell \le k$, to refer to the ball having the ℓ*th* largest radius in the j*th* cluster of size k. We also write $z_k^{(j)}[\ell]$ and $r_k^{(j)}[\ell]$ as shorthand to refer to the centre and radius of the ball $B_k^{(j)}[\ell]$.

The uniform upper bound

In extreme value theory, a classical method to investigate the maximum of a sequence of random variables consists in checking two conditions of the sequence. This was first

introduced by Leadbetter [83], who gave two conditions known respectively as $D(u_n)$ and $D'(u_n)$, which concern an asymptotic and a local property of the sequence. In the same spirit, we consider two cases in Proposition 8.15. The first of these may be considered as analogous to the asymptotic property, and concerns balls which are distant enough. The second condition is a locality condition and concerns the balls which are close together.

Proposition 8.15. *Given a collection of K balls, $B(z_i, r_i)$ for $i = 1, \dots, K$ such that $r_{1:K} > v_\rho$ and $R := \max_{1 \le i \le K} r_i$.*

(a) If $\min_{1 \le i,j \le K} |z_i - z_j| > R^3$, i.e. $(n_1, \dots, n_K) = (K, 0, \dots, 0)$, we have for ρ large enough

$$\phi\left(\bigcup_{1 \le i \le K} B(z_i, r_i) \right) \ge 2\pi \sum_{i=1}^{K} r_i - c \cdot v_\rho^{-1}.$$

(b) If $\min_{1 \le i,j \le K} |z_i - z_j| \le R^3$, we have

(b1) for ρ large enough,

$$\phi\left(\bigcup_{1 \le i \le K} B(z_i, r_i) \right) \ge 2\pi R + \left(\sum_{k=1}^{K} n_k - 1 \right) 2\pi v_\rho - c \cdot v_\rho^{-1},$$

(b2) when $R \le (1+\epsilon) v_\rho$, for some $\epsilon > 0$, we obtain for ρ large enough

$$\phi\left(\bigcup_{1 \le i \le K} B(z_i, r_i) \right) \ge 2\pi R + \left(\sum_{k=1}^{K} n_k - 1 \right) 2\pi v_\rho + \sum_{k=2}^{K} n_k (4 - \epsilon\pi) v_\rho - c \cdot v_\rho^{-1}.$$

Remark 8.16. We observe that (a) in Proposition 8.15 is quasi-optimal since we also have

$$\phi\left(\bigcup_{1 \le i \le K} B(z_i, r_i) \right) \le \sum_{i=1}^{K} \phi\left(B(z_i, r_i) \right) = 2\pi \sum_{i=1}^{K} r_i. \tag{8.21}$$

Thanks to (8.3), Proposition 8.15, (a) and (8.21) and assuming that the balls are distant enough (e.g. $|z_i - z_j| > R^3$ for any $1 \le i, j \le K$), we obtain

$$\left| \mathbb{P}\left(\bigcap_{1 \le i \le K} \left\{ \hat{\mathbf{X}} \cap B(z_i, r_i) = \varnothing \right\} \right) - \prod_{1 \le i \le K} \mathbb{P}\left(\hat{\mathbf{X}} \cap B(z_i, r_i) = \varnothing \right) \right| \le c \cdot v_\rho^{-1} \underset{\rho \to \infty}{\longrightarrow} 0. \tag{8.22}$$

The fact that the events considered in the probabilities above tend to be independent is well-known and is related to the fact that the tessellation $\mathfrak{m}_{\text{PHT}}$ satisfies a mixing property (e.g. proof of Theorem 10.5.3 in [104]). Our contribution is to provide a *uniform rate of convergence* (in the sense that it does not depend on the centres and the radii) which will be necessary to check (8.19).

Proof of Proposition 8.15

First, we state two lemmas. The first ones concerns two balls.

Lemma 8.17. *Let $z_1, z_2 \in \mathbf{R}^2$ and $R \geq r_1 \geq r_2 > v_\rho$ such that $|z_2 - z_1| > r_1 + r_2$.*

(i) If $|z_2 - z_1| > R^3$, we have $\mu\left(A(B(z_1, r_2)) \cap A(B(z_2, r_2))\right) \leq c \cdot v_\rho^{-1}$ for ρ large enough.

(ii) If $R \leq (1+\epsilon)v_\rho$ for some $\epsilon > 0$, we have $\mu\left(A(B(z_1, r_2)) \cap A(B(z_2, r_2))\right) \leq 2\pi r_2 - (4 - \epsilon\pi)v_\rho$.

The following lemma is a generalisation of the previous result.

Lemma 8.18. *Under the same conditions as Proposition 8.15, we have*

(i) $\mu\left(\bigcup_{1\leq i\leq K} A\left(B\left(z_i, r_i\right)\right)\right) \geq \sum_{k=1}^{K}\sum_{j=1}^{n_k} 2\pi \cdot r_k^{(j)}[1] - c \cdot v_\rho^{-1}$.

(ii) If $R \leq (1+\epsilon)v_\rho$ for some $\epsilon > 0$, we have the following more precise inequality

$$\mu\left(\bigcup_{1\leq i\leq K} A\left(B\left(z_i, r_i\right)\right)\right) \geq \sum_{k=1}^{K}\sum_{j=1}^{n_k} 2\pi \cdot r_k^{(j)}[1] + \sum_{k=2}^{K} n_k(4 - \epsilon\pi)v_\rho - c \cdot v_\rho^{-1}.$$

Proof of Proposition 8.15. (a) is a direct consequence of Lemma 8.18, (i). Using the inequality $r_k^{(j)}[1] > v_\rho$ for all $1 \leq k \leq K$ and $1 \leq j \leq n_k$ such that $r_k^{(j)}[1] \neq R$, we prove that (b1) and (b2) follow immediately from Lemma 8.18, (i) and (ii) respectively. □

Proof of Lemma 8.17. Let $z_1, z_2 \in \mathbf{R}^2$ and $R \geq r_1 \geq r_2 > v_\rho$ such that $|z_2 - z_1| > r_1 + r_2$ be fixed. According to (8.1) and the fact that μ is invariant under translations, we have

$$\begin{aligned}\mu\Big(A(B(z_1, r_1)) \cap A(B(z_2, r_2))\Big) &= 2\pi \int_{\mathbf{S}}\int_{\mathbf{R}_+} \mathbb{1}_{H(u,t)\cap B(0,r_1)\neq\varnothing}\, \mathbb{1}_{H(u,t)\cap B(z_2-z_1,r_2)\neq\varnothing}\, dt\sigma(du) \\ &= f(r_1, r_2, |z_2 - z_1|),\end{aligned}$$

where

$$\begin{aligned}f(r_1, r_2, h) \coloneqq 2(r_1 + r_2)\arcsin\left(\frac{r_1 + r_2}{h}\right) &- 2(r_1 - r_2)\arcsin\left(\frac{r_1 - r_2}{h}\right) \\ &- 2h\left(\sqrt{1 - \left(\frac{r_1 - r_2}{h}\right)^2} - \sqrt{1 - \left(\frac{r_1 + r_2}{h}\right)^2}\right)\end{aligned}$$

for all $h > r_1 + r_2$. It may be demonstrated that the function $g\colon (r_1 + r_2, \infty) \to \mathbf{R}_+,\ h \mapsto f(r_1, r_2, h)$ is positive, strictly decreasing and converges to 0 as h tends to infinity. We now consider each of the two cases given above.

Proof of (i). Suppose that $|z_2 - z_1| > R^3$. Using the fact that $r_1 + r_2 \leq 2R$, $\arcsin((r_1 + r_2)/(|z_2 - z_1|)) \leq \arcsin(2/R^2)$ and $r_1 \geq r_2$ we obtain for ρ large enough,

$$f(r_1, r_2, |z_2 - z_1|) \ < \ f(r_1, r_2, R^3) \ \leq \ 4R\arcsin\left(\frac{2}{R^2}\right) \ \leq \ c \cdot R^{-1} \leq c \cdot v_\rho^{-1}$$

Proof of (ii). Suppose that $R \leq (1+\epsilon)v_\rho$. Since $|z_2 - z_1| > r_1 + r_2$, we get

$$f(r_1, r_2, |z_2 - z_1|) \;<\; f(r_1, r_2, r_1 + r_2) \;=\; 2\pi r_2 + 2(r_1 - r_2)\arccos\left(\frac{r_1 - r_2}{r_1 + r_2}\right) - 4\sqrt{r_1 r_2}.$$

Using the fact that $r_1 \geq r_2 > v_\rho$, $\arccos\left(\frac{r_1-r_2}{r_1+r_2}\right) \leq \frac{\pi}{2}$ and $r_1 \leq R \leq (1+\epsilon)v_\rho$, we obtain

$$f(r_1, r_2, |z_2 - z_1|) \;<\; 2\pi r_2 + (r_1 - v_\rho)\pi - 4v_\rho \;\leq\; 2\pi r_2 - (4 - \epsilon\pi)v_\rho.$$

□

Proof of Lemma 8.18 (i). Using the notation defined in Section 8.4, we obtain from Bonferroni inequalities

$$\begin{aligned}\mu\left(\bigcup_{1\leq i\leq K} A\big(B(z_i, r_i)\big)\right) &= \mu\left(\bigcup_{k\leq K}\bigcup_{j\leq n_k} A\left(B_k^{(j)}\right)\right) \\ &\geq \sum_{k=1}^{K}\sum_{j=1}^{n_k} \mu\left(A\left(B_k^{(j)}\right)\right) \;-\; \sum_{(k_1,j_1)\neq(k_2,j_2)} \mu\left(A\left(B_{k_1}^{(j_1)}\right) \cap A\left(B_{k_2}^{(j_2)}\right)\right).\end{aligned} \tag{8.23}$$

We begin by observing that for all $1 \leq k_1 \neq k_2 \leq K$ and $1 \leq j_1 \leq n_{k_1}$, $1 \leq j_2 \leq n_{k_2}$

$$\mu\left(A\left(B_{k_1}^{(j_1)}\right) \cap A\left(B_{k_2}^{(j_2)}\right)\right) \;\leq \sum_{1\leq \ell_1 \leq k_1, 1\leq \ell_2 \leq k_2} \mu\left(A\left(B_{k_1}^{(j_1)}[\ell_1]\right) \cap A\left(B_{k_2}^{(j_2)}[\ell_2]\right)\right) \;\leq\; c \cdot v_\rho^{-1} \tag{8.24}$$

when ρ is sufficiently large, with the final inequality following directly from Lemma 8.17, (i) with $r_1 := r_{k_1}^{(j_1)}[\ell_1]$ and $r_2 := r_{k_2}^{(j_2)}[\ell_2]$. In addition,

$$\mu\left(A\left(B_k^{(j)}\right)\right) \geq \mu\left(A\left(B_k^{(j)}[1]\right)\right) = 2\pi \cdot r_k^{(j)}[1]. \tag{8.25}$$

We then deduce (i) from (8.23), (8.24) and (8.25). □

Proof of Lemma 8.18 (ii). We proceed along the same lines as in the proof of (i). The only difference concerns the lower bound for $\mu(A(B_k^{(j)}))$. We consider two cases. Firstly, for each of the n_1 clusters of size one, we have $\mu(A(B_1^{(j)})) = 2\pi r_1^{(j)}[1]$. Otherwise, we obtain

$$\begin{aligned}\mu\left(A\left(B_k^{(j)}\right)\right) &= \mu\left(\bigcup_{\ell=1}^{k} A\left(B_k^{(j)}[\ell]\right)\right) \\ &\geq \mu\left(A\left(B_k^{(j)}[1]\right) \cup A\left(B_k^{(j)}[2]\right)\right) \\ &= 2\pi r_k^{(j)}[1] + 2\pi r_k^{(j)}[2] - \mu\left(A\left(B_k^{(j)}[1]\right) \cap A\left(B_k^{(j)}[2]\right)\right) \\ &\geq 2\pi \cdot r_k^{(j)}[1] + (4 - \epsilon\pi)v_\rho\end{aligned}$$

which follows from Lemma 8.17, (ii). We then deduce (ii) from the previous inequality and (8.23), (8.24). □

8.5 Order statistics for the large inballs

In this section, we make use a number of additional notational conventions. Let $t \geq 0$ be fixed. We denote by

$$\tau := \tau(t) := e^{-t} \quad \text{and} \quad v_\rho := v_\rho(t) := \frac{1}{2\pi}(\log(\pi\rho) + t). \tag{8.26}$$

For any $K \geq 1$ and for any K-tuple of convex bodies $C_1 \ldots, C_K$ such that each C_i has a unique inball, define the events:

$$E_{C_{1:K}} := \left\{ \min_{1 \leq i \leq K} R(C_i) \geq v_\rho,\ R(C_1) = \max_{1 \leq i \leq K} C_i \right\} \tag{8.27a}$$

$$E^{\star}_{C_{1:K}} := \left\{ \min_{1 \leq i,j \leq K} |z(C_i) - z(C_j)| > R(C_1)^3 \right\} \tag{8.27b}$$

$$E^{\circ}_{C_{1:K}} := \left\{ \forall 1 \leq i \neq j \leq K,\ B(C_i) \cap B(C_j) = \varnothing \right\}. \tag{8.27c}$$

For any $K \geq 1$, we take

$$I^{(K)}(\rho) := K\,\mathbb{E}\left[\sum_{\substack{C_{1:K} \in (\mathfrak{m}_{\text{PHT}})^K_{\neq}, \\ z(C_{1:K}) \in \mathbf{W}_\rho^K}} \mathbb{1}_{E_{C_{1:K}}} \right] \tag{8.28}$$

The configuration graph

When $K \geq 2$, we need a way to quantify the dependence between the inballs of the cells and the lines of the process. Since the inball of every cell in $\mathfrak{m}_{\text{PHT}}$ intersects exactly three lines (almost surely), it will be sufficient to consider all ways in which K triples of lines can depend on each other. This dependence structure is most naturally represented by a bipartite graph $\mathbf{G}(V_C, V_L, E)$, with vertex sets V_C, V_L and edges E. Each vertex in V_C will be taken to represent the triple of lines intersecting an inball in the process, and the vertices in V_L will represent the lines in the process. An edge between a vertex $L \in V_L$ and a vertex $C \in V_C$ will be interpreted as meaning that the inball represented by C intersects the line represented by L. An example of this construction is given in Figure 8.4. When clear from context, we use vertices in V_C, V_L interchangeably with their counterparts in the tessellation. Let Λ_K be the set of all bipartite graphs (up to isomorphism) whose vertex sets V_C, V_L satisfy

i. $|V_C| = K$

ii. $\forall C \in V_C$, $\text{degree}(C) = 3$

iii. $\forall L \in V_L$, $\text{degree}(L) > 0$

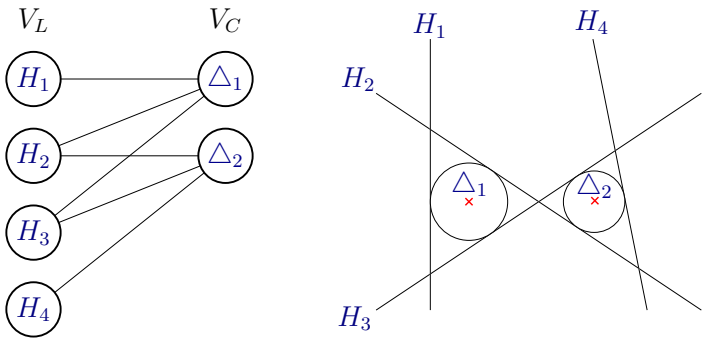


Figure 8.4 – Example of configuration of inballs and lines, with associated configuration graph.

iv. $\forall (C_1, C_2) \in (V_C)^2_{\neq}$, neighbours$(C_1) \neq$ neighbours(C_2)

Then Λ_K contains the set of all possible dependence configurations between K triples of lines in a line tessellation. Given a set of cells $C_{1:K}$, we write $\mathbf{G}(C_{1:K}) \in \Lambda_K$ to denote the configuration associated with the lines of $\mathbf{X}$ and the inballs of the cells $C_{1:K}$. We denote the associated vertex sets $V_C(C_{1:K})$, $V_L(C_{1:K})$. Let us notice that $|V_L(C_{1:K})| \leq 3K$.

According to (8.28), we can now write $I^{(K)}(\rho)$ as

$$I^{(K)}(\rho) = I_1^{(K)}(\rho) + I_2^{(K)}(\rho) + I_3^{(K)}(\rho), \tag{8.29}$$

where

$$I_1^{(K)}(\rho) := K\,\mathbb{E}\left[\sum_{\substack{C_{1:K}\in(\mathfrak{m}_{\mathrm{PHT}})^K_{\neq},\\ z(C_{1:K})\in\mathbf{W}_\rho^K}} \mathbb{1}_{E_{C_{1:K}}}\,\mathbb{1}_{E^\star_{C_{1:K}}}\,\mathbb{1}_{|V_L(C_{1:K})|=3K}\right], \tag{8.30a}$$

$$I_2^{(K)}(\rho) := K\,\mathbb{E}\left[\sum_{\substack{C_{1:K}\in(\mathfrak{m}_{\mathrm{PHT}})^K_{\neq},\\ z(C_{1:K})\in\mathbf{W}_\rho^K}} \mathbb{1}_{E_{C_{1:K}}}\,\mathbb{1}^{\mathsf{c}}_{E^\star_{C_{1:K}}}\,\mathbb{1}_{|V_L(C_{1:K})|=3K}\right], \tag{8.30b}$$

$$I_3^{(K)}(\rho) := K\,\mathbb{E}\left[\sum_{\substack{C_{1:K}\in(\mathfrak{m}_{\mathrm{PHT}})^K_{\neq},\\ z(C_{1:K})\in\mathbf{W}_\rho^K}} \mathbb{1}_{E_{C_{1:K}}}\,\mathbb{1}_{|V_L(C_{1:K})|<3K}\right]. \tag{8.30c}$$

Noting that on the event $|V_L(C_{1:K})| = 3K$, none of the inballs share any lines. The asymptotic behaviours of these functions are dealt with in the following result.

Proposition 8.19. *Let $I_1^{(K)}(\rho), I_2^{(K)}(\rho)$ and $I_3^{(K)}(\rho)$ be defined as in* (8.30a), (8.30b) *and* (8.30c). *Then*

(a) $I_1^{(K)}(\rho) \underset{\rho\to\infty}{\longrightarrow} \tau^K$.

(b) $I_2^{(K)}(\rho) \underset{\rho\to\infty}{\longrightarrow} 0$.

(c) $I_3^{(K)}(\rho) \underset{\rho\to\infty}{\longrightarrow} 0$.

The convergences in Proposition 8.19 can be understood intuitively as follows. For $I_1^{(K)}(\rho)$, the inradii of the cells behave as though they are independent, since they are far apart and no line in the process touches more than one of the inballs in the K-tuple (even though two *cells* in the K-tuple may share a line). For $I_2^{(K)}(\rho)$, we are able to show that, with high probability, the inradii of neighbouring cells cannot simultaneously exceed the level v_ρ, due to Proposition 8.15, (b). Finally, to bound $I_3^{(K)}(\rho)$ we use the fact that the proportion of K-tuples of cells which share at least one line is negligible relative to those that do not.

Proofs

Proof of Theorem 8.1 (b). According to Lemma 8.14, it is now enough to show that for all $K \geq 1$, we have $I^{(K)}(\rho) \underset{\rho\to\infty}{\longrightarrow} \tau^K$. This fact is a consequence of (8.29) and Proposition 8.19. □

Proof of Proposition 8.19 (a). For any $1 \leq i \leq K$ and any 3−tuple of lines $H_i^{(1:3)} = (H_i^{(1)}, H_i^{(2)}, H_i^{(3)})$, let us recall that $\triangle_i := \triangle_i(H_i^{(1)}, H_i^{(2)}, H_i^{(3)})$ denotes the unique triangle that can be formed by the intersection of the half- spaces induced by the lines $H_i^{(1:3)}$. For brevity, let us write $B_i = B(\Delta)_i$ and $H_{1:K}^{(1:3)} = \left(H_1^{(1:3)}, \ldots, H_K^{(1:3)}\right)$. We shall often omit the arguments when they are obvious from context. First, we write

$$\begin{aligned} I_1^{(K)}(\rho) &= \frac{K}{6^K}\,\mathbb{E}\left(\sum_{H_{1:K}^{(1:3)}\in \mathbf{X}_{\neq}^{3K}} \mathbb{1}_{\left\{\hat{\mathbf{X}}\setminus\cup_{i\leq K, j\leq 3} H_i^{(j)}\right\}\cap\left\{\cup_{i\leq K} B_i\right\}=\varnothing}\, \mathbb{1}_{E_{B_{1:K}}}\, \mathbb{1}_{E^{\star}_{B_{1:K}}}\, \mathbb{1}_{z(B_{1:K})\in \mathbf{W}_\rho^K}\right) \\ &= \frac{K}{6^K}\int_{\mathcal{A}^{3K}} e^{-\phi\left(\cup_{i\leq K} B_i\right)} \mathbb{1}_{E_{B_{1:K}}}\, \mathbb{1}_{E^{\star}_{B_{1:K}}}\, \mathbb{1}_{z(B_{1:K})\in \mathbf{W}_\rho^K}\, \mu\left(dH_{1:K}^{(1:3)}\right), \end{aligned}$$

where the last equality comes from (8.3) and the Slivnyak-Mecke formula. Applying the Blaschke-Petkantschin formula, we get

$$I_1^{(K)}(\rho) = \frac{K}{24^K}\int_{(\mathbf{W}_\rho\times\mathbf{R}_+\times\mathbf{S}^3)^K} \prod_{i\leq K} e^{-\phi\left(\cup_{i\leq K} B(z_i,r_i)\right)} a\left(u_i^{(1:3)}\right) \mathbb{1}_{E_{B_{1:K}}}\, \mathbb{1}_{E^{\star}_{B_{1:K}}}\, dz_{1:K} dr_{1:K} \sigma\left(du_{1:K}^{(1:3)}\right),$$

where we recall that $a\left(u_i^{(1:3)}\right)$ is the area of the triangle spanned by $u_i^{(1:3)}$. From Proposition 8.15 (a) and (8.21), we have for any $1 \leq i \leq K$

$$e^{-2\pi r_i}\mathbb{1}_{E_{B_{1:K}}} \quad \leq \quad e^{-\phi\left(\cup_{i\leq K} B(z_i,r_i)\right)}\mathbb{1}_{E_{B_{1:K}}} \quad \leq \quad e^{-2\pi r_i}\mathbb{1}_{E_{B_{1:K}}} \cdot e^{c v_\rho{}^{-1}}.$$

According to (8.27a) and (8.27b), this implies that

$$\begin{aligned} I_1^{(K)} &\underset{\rho\to\infty}{\sim} \frac{K}{24^K}\int_{(\mathbf{W}_\rho\times\mathbf{R}_+\times\mathbf{S}^3)^K}\prod_{i\le K} e^{-2\pi\cdot r_i}a\left(u_i^{(1:3)}\right)\mathbb{1}_{r_i>v_\rho} \\ &\qquad\qquad \times \mathbb{1}_{r_1=\max_{j\le K} r_j}\,\mathbb{1}_{|z_i-z_j|>r_1^3 \text{ for } j\neq i}\,dz_{1:K}dr_{1:K}\sigma(du_{1:K}^{(1:3)}) \\ &= \frac{K\tau^K}{(24\pi)^K}\int_{(\mathbf{W}_1\times\mathbf{R}_+\times\mathbf{S}^3)^K}\prod_{i\le K} e^{-2\pi\cdot r_i}a\left(u_i^{(1:3)}\right) \\ &\qquad\qquad \times \mathbb{1}_{r_1=\max_{j\le K} r_j\mathbb{1}_{|z_i-z_j|>\rho^{-1/2}r_1^3 \text{ for } j\neq i}}\,dz_{1:K}dr_{1:K}\sigma(du_{1:K}^{(1:3)}), \end{aligned}$$

where the last equality comes from (8.26), the change of variables $z_i' = \rho^{-1/2}z_i$ and $r_i' = r_i - v_\rho$. It follows from the monotone convergence theorem that

$$\begin{aligned} I_1^{(K)}(\rho) &\underset{\rho\to\infty}{\sim} \frac{K\tau^K}{(24\pi)^K}\int_{(\mathbf{W}_1\times\mathbf{R}_+\times\mathbf{S}^3)^K}\prod_{i\le K} e^{-2\pi\cdot r_i}a\left(u_i^{(1:3)}\right)\mathbb{1}_{r_1=\max_{j\le K} r_j} \\ &\qquad\qquad \times dz_{1:K}dr_{1:K}\sigma(du_{1:K}^{(1:3)}) \\ &= \frac{\tau^K}{(24\pi)^K}\left(\int_{(\mathbf{W}_1\times\mathbf{R}_+\times\mathbf{S}^3)^K} a(u_{1:3})e^{-2\pi r}dzdr\sigma(du_{1:3})\right)^K. \end{aligned}$$

Integrating over z, r and $u_{1:3}$ and using the fact that $\lambda_2(\mathbf{W}_1) = 1$ and

$$\int_{\mathbf{S}^3} a(u_{1:3})\sigma(du_{1:3}) = 48\pi^2,$$

we obtain that $I_1^{(K)}(\rho) \underset{\rho\to\infty}{\longrightarrow} \tau^K$. □

Proof of Proposition 8.19 (b). Beginning in the same way as for the proof of (a), we have

$$\begin{aligned} I_2^{(K)}(\rho) = \frac{K}{24^K}\int_{(\mathbf{W}_\rho\times\mathbf{R}_+\times\mathbf{S}^3)^K}\prod_{i\le K} e^{-\phi\left(\cup_{i\le K}B(z_i,r_i)\right)}a\left(u_i^{(1:3)}\right)\mathbb{1}_{E_{B_{1:K}}}\mathbb{1}_{\left(E^\star_{B_{1:K}}\right)^c} \\ \times \mathbb{1}_{E^\circ_{B_{1:K}}}\,dz_{1:K}dr_{1:K}\sigma\left(du_{1:K}^{(1:3)}\right). \end{aligned}$$

Here and subsequently, we consider the event $E^\circ_{B_{1:K}}$ to specify that the inballs of the cells are disjoint. Integrating over $u_{1:K}^{(1:3)}$ and summing over all configurations $n_{1:K} = (n_1,\dots,n_K) \neq (K,0\dots,0)$ such that $\sum_{k=1}^K kn_k = K$, we get

$$\begin{aligned} I_2^{(K)}(\rho) &= c\cdot\sum_{n_{1:K}}\int_{(\mathbf{W}_\rho\times\mathbf{R}_+)^K}\prod_{i\le K} e^{-\phi\left(\cup_{i\le K}B(z_i,r_i)\right)}\mathbb{1}_{E_{B_{1:K}}}\mathbb{1}_{E^\circ_{B_{1:K}}}\mathbb{1}_{n_{1:K}(z_{1:K},r_1)=n_{1:K}}\,dz_{1:K}dr_{1:K} \\ &= I_{2a,\epsilon}(\rho) + I_{2b,\epsilon}(\rho), \end{aligned}$$

where, for any $\epsilon > 0$, the terms $I_{2a,\epsilon}(\rho)$ and $I_{2b,\epsilon}(\rho)$ are defined as the term of the first line when we add the indicator that r_1 is larger than v_ρ in the integral and the indicator for the complement respectively. We provide below a suitable upper bound for these two terms.

For $I_{2a,\epsilon}(\rho)$, we obtain from Proposition 8.15 (b1) that

$$I_{2a,\epsilon}(\rho) \leq c \cdot \sum_{n_{1:K}} \int_{(\mathbf{W}_\rho \times \mathbf{R}_+)^K} e^{-\left(2\pi r_1 + \left(\sum_{k=1}^K n_k - 1\right) 2\pi v_\rho - c \cdot v_\rho^{-1}\right)} \times \mathbb{1}_{r_1 = \max_{j \leq K} r_j} \, \mathbb{1}_{n_{1:K}(z_{1:K}, r_1) = n_{1:K}} \, \mathbb{1}_{r_1 > (1+\epsilon) v_\rho} \, dz_{1:K} \, dr_{1:K}.$$

Integrating over $r_{2:K}$ and $z_{1:K}$, we obtain

$$I_{2a,\epsilon}(\rho) \leq c \cdot \sum_{n_{1:K}} \int_{(1+\epsilon)v_\rho}^\infty r_1^{K-1} e^{-\left(2\pi r_1 + \left(\sum_{k=1}^K n_k - 1\right) 2\pi v_\rho\right)} \lambda_{dK}\left(\{z_{1:K} \in \mathbf{W}_\rho^K : n_{1:K}(z_{1:K}, r_1) = n_{1:K}\}\right) dr_1. \tag{8.31}$$

For each $n_{1:K}$ such that $\sum_{k=1}^K k n_k = K$, we can easily prove that

$$\lambda_{dK}\left(\{z_{1:K} \in \mathbf{W}_\rho^K : n_{1:K}(z_{1:K}, r_1) = n_{1:K}\}\right) \leq c \cdot \rho^{\sum_{k=1}^K n_k} \cdot r_1^{6\left(K - \sum_{k=1}^K n_k\right)} \tag{8.32}$$

since the number of connected components of $\bigcup_{i=1}^K B(z_i, r_1^3)$ equals $\sum_{k=1}^K n_k$. It follows from (8.31) and (8.32) that there exists a constant $c(K)$ such that

$$\begin{aligned} I_{2a,\epsilon}(\rho) &\leq c \cdot \sum_{n_{1:K}} \left(\rho e^{-2\pi v_\rho}\right)^{\left(\sum_{k=1}^K n_k\right)} e^{2\pi v_\rho} \int_{(1+\epsilon)v_\rho}^\infty r_1^{c(K)} e^{-2\pi r_1} \\ &= O\left((\log \rho)^{c(K)} \rho^{-\epsilon}\right) \end{aligned}$$

according to (8.26). For $I_{2b,\epsilon}^{(K)}$, we proceed exactly as for $I_{2a}^{(K)}$, but this time we apply the bound given in Proposition 8.15 (b2). We obtain

$$\begin{aligned} I_{2b,\epsilon}^{(K)} &\leq c \cdot \sum_{n_{1:K}} \left(\rho \, e^{-2\pi v_\rho}\right)^{\left(\sum_{k=1}^K n_k\right)} e^{2\pi v_\rho - \sum_{k=2}^K n_k (4 - \epsilon\pi) v_\rho} \int_{v_\rho}^{(1+\epsilon)v_\rho} r_1^{c(K)} e^{-2\pi r_1} \, dr_1 \\ &= O\left(\rho^{-\frac{4-\epsilon\pi}{2\pi}} \log \rho\right) \end{aligned}$$

since there exists a $2 \leq k \leq K$ such that n_k is non-zero. Choosing $\epsilon < \frac{4}{\pi}$ proves that $I_{2b,\epsilon}^{(K)}$ converges to 0 as ρ goes to infinity. □

Proof of Proposition 8.19 (c). Given a bipartite graph $\mathbf{G} = \mathbf{G}(V_L, V_C, E)$ as in page 116, we denote by $B_i(\mathbf{G})$ the inball of $\Delta_i \in V_C$ for each $1 \leq i \leq K$. With each line $H_i \in \hat{\mathbf{X}}$ and each triangle $\triangle_i$, we associate a unique vertex in V_L and a unique vertex in V_C, so that $\mathbf{G}$ now defines the dependence structure between the cells and the lines. We may now re-write (8.30c) as follows:

$$\begin{aligned} I_3^{(K)} &= K \sum_{\substack{\mathbf{G} \in \Lambda_K, \\ |V_L| < 3K}} \mathbb{E}\left[\sum_{H_{1:|V_L|} \in \mathbf{X}_{\neq}^{|V_L|}} \mathbb{1}_{\left\{\hat{\mathbf{X}} \setminus \cup_{i \leq |V_L|} H_i\right\} \cap \{\cup_{i \leq K} B_i(\mathbf{G})\} = \varnothing} \, \mathbb{1}_{E_{B_{1:K}(\mathbf{G})}} \, \mathbb{1}_{E^\circ_{B_{1:K}(\mathbf{G})}} \, \mathbb{1}_{z(B_{1:K}(\mathbf{G})) \in \mathbf{W}_\rho^K}\right] \\ &= K \sum_{\substack{\mathbf{G} \in \Lambda_K, \\ |V_L| < 3K}} \int_{\mathcal{A}^{|V_L|}} e^{-\phi\left(\cup_{i \leq K} B_i(\mathbf{G})\right)} \mathbb{1}_{E_{B_{1:K}(\mathbf{G})}} \, \mathbb{1}_{E^\circ_{B_{1:K}(\mathbf{G})}} \, \mathbb{1}_{z(B_{1:K}(\mathbf{G})) \in \mathbf{W}_\rho^K} \, \mu(dH_{1:|V_L|}). \end{aligned}$$

Recalling the definition of the connected components in (8.20), we let $\kappa(C; C_{1:K})$ be the indicator that the inball of C is the largest in its associated connected component in the graph over $C_{1:K}$, noting that $\kappa(\triangle_1) = 1$ by definition. We then apply the Slivnyak-Mecke formula, and sum over configurations of clusters to give

$$\leq \sum_{\substack{\mathbf{G}\in\Lambda_K,\\ |V_L|<3K}} \sum_{x\in(0,1)^{K-1}} K \int_{\mathcal{A}(1)^{|V_L|}} e^{-\phi\left(\cup_{i\leq K} B(\triangle_i)\right)} \mathbb{1}_{E_{\triangle_{1:K}}} \mathbb{1}_{\forall i\leq K,\, \left\{z(\triangle_i)\in\mathbf{W}_\rho,\, \kappa(\triangle_i)=x_i\right\}} \mu\left(dH_{1:|V_L|}\right)$$

As in the proof for (b), we partition the integral in to two parts, $I_{3a}^{(K)}(\rho)$ and $I_{3b}^{(K)}(\rho)$ using the event $r(\triangle_1) \geq (1+\varepsilon)v_\rho$. Applying Proposition 8.15 (b1) and observing that $\sum_{i=1}^K n_k - 1 = \sum_{i=2}^K x_i$ gives

$$I_{3a}^{(K)} \leq \sum_{\substack{\mathbf{G}\in\Lambda_K,\\ |V_L|<3K}} \sum_{x\in(0,1)^{K-1}} K \int_{\mathcal{A}(1)^{|V_L|}} e^{-2\pi\left(r(\triangle_1)+v_\rho\sum_{i=2}^K x_i\right)} \mathbb{1}_{E_{\triangle_{1:K}}} \mathbb{1}_{z(1:K)\in\mathbf{W}_\rho^K} \mathbb{1}_{\kappa(\triangle_{i:K})=x_{i:K}}$$
$$\times \mathbb{1}_{r(\triangle_1)\geq(1+\varepsilon)v_\rho} \mu\left(dH_{1:|V_L|}\right)$$

integrating over the centres of the largest ball in each cluster, and the first $K-1$ radii gives

$$\begin{aligned}
&\leq \sum_{\substack{\mathbf{G}\in\Lambda_K,\\ |V_L|<3K}} \sum_{x\in(0,1)^{K-1}} c\cdot e^{\left(-2\pi v_\rho\left(\sum_{k=1}^K n_k-1\right)\right)} \rho^{\left(\sum_{i=1}^K x_i\right)} \int_{(1+\varepsilon)v_\rho}^\infty r^{c\cdot K} e^{-2\pi r} dr \\
&\leq \sum_{\substack{\mathbf{G}\in\Lambda_K,\\ |V_L|<3K}} \sum_{x\in(0,1)^{K-1}} c\cdot\left(\rho\exp(-2\pi v_\rho)\right)^{\left(\sum_{i=2}^K x_i\right)} \rho \int_{(1+\varepsilon)v_\rho}^\infty r^{c\cdot K} e^{-2\pi r} dr \\
&= O\left(\tau^K\cdot(\log\rho)^{c\cdot K}\cdot\rho^{-\varepsilon}\right).
\end{aligned}$$

In the following, we compute the integrals over the contributions of each cell. For each cell satisfying $x_i = 1$ and sharing no lines with cells $j = 1,\ldots,i-1$, we allow the centre to fall anywhere within the window, giving a contribution of ρ to the integral. For those cells with $x_i = 1$ that *do* share a line with a previous cell, the centre must lie within a distance r_1 of one of the previous lines, so the contribution of the integral is reduced to $c\cdot\rho^{1/2}\cdot r$. To formalise this, we construct a set of cells satisfying this second property, let

$$D_{x_{1:k},\mathbf{G}} := \left\{ i : x_i = 1,\, \text{neighbours}(\triangle_i) \cap \bigcup_{\substack{j<i,\\ x_j=1}} \text{neighbours}(\triangle_i) \neq \varnothing \right\}. \tag{8.33}$$

Recalling that neighbours$(\triangle_i)$ is defined to be the lines associated with the cell $\triangle_i$. Thus,

the integral for the second part becomes

$$
\begin{aligned}
I_{3b}^{(K)} &\leq \sum_{\substack{\mathbf{G}\in\Lambda_K,\\ |V_L|<3K}} \sum_{x\in(0,1)^{K-1}} c\cdot e^{\left(-2\pi v_\rho\left(\sum_{k=1}^K n_k-1\right)-\sum_{k=2}^K n_k(4-\varepsilon\pi)v_\rho\right)}\rho^{\left(\sum_{i=1}^K x_i\right)}\rho^{-D/2}\int_{v_\rho}^{(1+\varepsilon)v_\rho} r^{c\cdot K}e^{-2\pi r}dr \\
&\leq \sum_{\substack{\mathbf{G}\in\Lambda_K,\\ |V_L|<3K}} \sum_{x\in(0,1)^{K-1}} c\cdot\left(\rho\exp(-2\pi v_\rho)\right)^{\left(\sum_{i=2}^K x_i\right)} e^{-\left(\sum_{k=2}^K n_k(4-\varepsilon\pi)v_\rho\right)}\rho^{-D/2}\rho e^{-2\pi v_\rho}\log(\rho)^{c\cdot K} \\
&\leq \sum_{\substack{\mathbf{G}\in\Lambda_K,\\ |V_L|<3K}} \sum_{x\in(0,1)^{K-1}} c\cdot\tau^K\cdot e^{-\left(\sum_{k=2}^K n_k(4-\varepsilon\pi)v_\rho\right)}\rho^{-D/2}\log(\rho)^{c\cdot K} \\
&= O\left(\log(\rho)^{c\cdot K}\rho^{-1/2}\right)
\end{aligned}
$$

The last line follows by considering two cases. Suppose that there exists at least one cluster with two cells, in which case there exists a $k > 1$ such that $n_k \geq 1$, and we can choose an ε such that the result follows. For the second case, if there is no cluster with two cells, this implies that every x_i must be the only cell within each cluster, so $x_i = 1$ for every $i = 1,\ldots,K$. Since we also have that $|V_L| < 3K$, this means that at least one of the cells i, shares a line with a cell $j < i$, so that $|D| > 1$. □

Appendices

Appendix A

Software for investigating the properties of random triangulations

As part of this thesis, a piece of software was developed to investigate the behaviour of random Delaunay triangulations, Voronoi diagrams, and random line tessellations. In particular, the software enables a user to generate and manipulate triangulations with approximately 2×10^7 points in real time. This is achieved by using a quad tree to consistently sample points from the process up to an approximate target density given the current area being viewed. A screenshot of the application is given in Figure A.2.

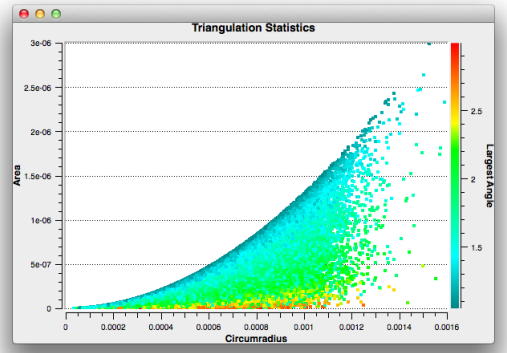
A.1 Plotting

The application allows subsets of the points to be selected, with an option to generate plots of different characteristics of the Triangulation (or Voronoi Diagram.) These plots can be made with between one and three variables, using either a histogram, a scatter plot, or a scatter plot with coloured markers.

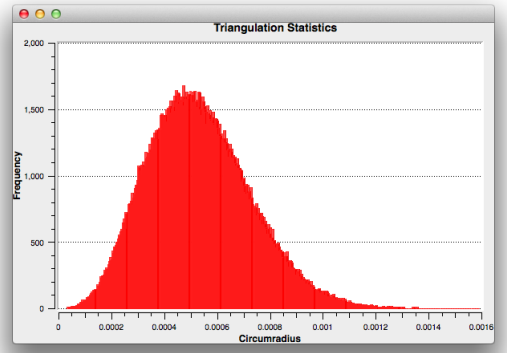
When the selected region is sufficiently large, the application can be used to obtain quite reasonable estimates for the distributions of functions for different properties, and gives good intuition for understanding the dependence between various properties of Delaunay triangulations. Some example plots are given in Figure A.1.

A.2 Visualising order statistics

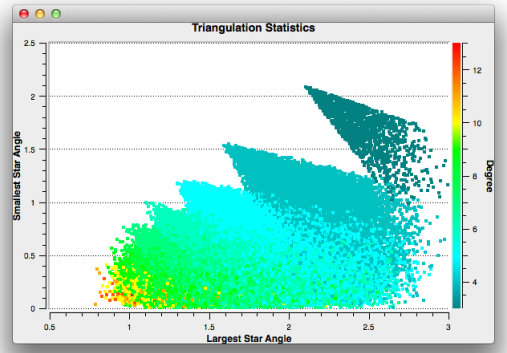
Another useful feature, which is of great use for investigating the extremal index (defined in Section 7.1), is the ability to iterate through the order statistics for properties of cells, vertices or lines in the triangulation (or its dual.) An additional feature enables the high-



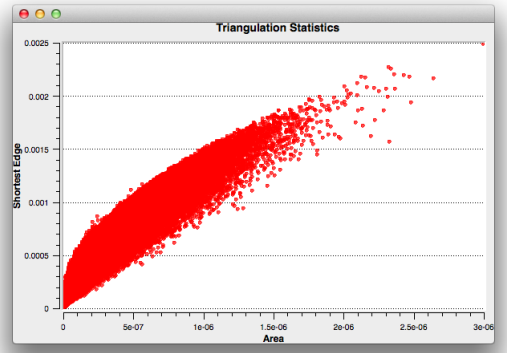
(a) Circumradius against area.



(b) Histogram of circumradii.



(c) Min. vs max. star angle.



(d) Area against shortest edge.

Figure A.1 – The above plots are examples of output generated by the visualiser application. Figure A.1a is a plot of triangle area against circumradius, with the colour representing the magnitude of the largest angle in the triangle. Figure A.1b is a histogram of the circumradii. Figure A.1c is the *smallest star angle* plotted against the *maximum star angle* for each vertex, with the colour representing the degree of each vertex. Figure A.1d is a scatter plot of the area of the triangles against the shortest edge lengths.

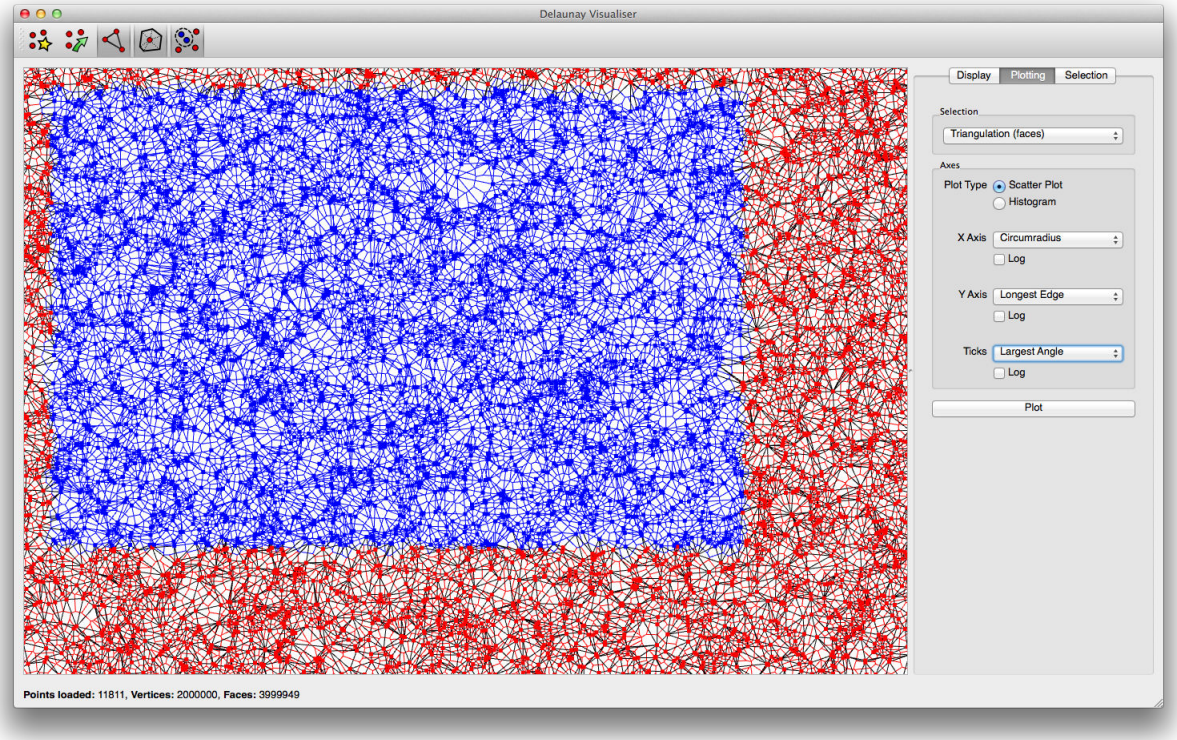


Figure A.2 – The visualiser application running, displaying a the Delaunay triangulation (and Voronoi diagram) of two million random points. The region selected is the region used for the plots in Figure A.1.

lighting of all faces, lines or vertices such that a given property falls below (or above) a given threshold.

A.3 Integration into CGAL

The software developed is highly dependent on the CGAL library for all geometric algorithms; however it quickly became apparent that CGAL does not provide many functions in its API to compute approximations to commonly used properties of geometric structures, or functions to iterate over containers using STL-like functionality. We thus provide the utilities used in the development of this application as a package for CGAL, which we call 'Property_generator'. At the time of writing, this package is currently under review, though it should be released in CGAL 4.6

The given package comprises of two parts. The first being a selection of STL- like functions which act on iterator ranges and unary functions. The second part is a group of function objects operating on different parts of Delauany triangulations (though the framework has been designed to be appropriate for any geometric container.) A challenge in the design of the API was to enable the 'statistics functions' to operate on ranges of

iterators which may be seen as both geometric objects *and* 'handles' to the objects. This is made necessary by a feature in the CGAL API which identifies iterators with their handle types. In practice, this is achieved using template meta-programming techniques using the Boost MPL library.

The visualisation software is distributed as free and open source software, and will be included with the Property Generator package as a demo when it is released.

Nomenclature

$\mathbb{E}$ The expected value, page 11

$\mathbb{P}$ A probability measure, page 12

$\mathbb{P}_x$ The Palm measure, page 14

$\mathbf{E}$ A general locally compact space, page 13

$\mathbf{E}$ A general space, page 7

$\mathbf{Z}$ The ring of integers, page 7

$\mathbf{N}$ Non-negative integers, page 7

$\mathbf{R}$ The real numbers, page 7

$\mathbf{S}$ The unit sphere in $\mathbf{R}^2$, page 7

$\mathbf{S}^d$ The unit sphere in $\mathbf{R}^d$, page 7

$\mathbf{W}(\mathbf{X}, q)$ The walk graph, page 30

$\mathbf{W}_x$ The restriction of the walk graph to the star of x, page 82

$\mathbf{W}_\rho$ A window scaled by $\rho^{1/d}$, page 95

$\mathbf{W}$ A (usually compact convex) set to be used as a window, page 95

$\mathbf{X}$ A point process, page 13

κ_d Volume of the unit ball in $\mathbf{R}^d$, page 7

χ The maximum degree of a dependency graph., page 18

$\delta_{\mathbf{X}}(x)$ The degree of x in $\mathbf{X}$, page 37

δ_x The Dirac measure, page 13

γ The intensity of a homogeneous Poisson process, page 97

$\gamma_{\mathfrak{m}}$ The intensity of a random tessellation, page 93

λ_d The d-dimensional Lebesgue measure, page 7

μ The intensity measure of a point process, page 14

η A counting measure, page 13

η_s A simple counting measure, page 13

Ω The space of outcomes in a probability space, page 12

ω_n A sequence growing faster than $\log n$, page 53

$\phi(\cdot)$ The measure of affine lines intersecting a region, page 111

$\sigma(\cdot)$ The uniform Lebesgue measure on the sphere, page 97

σ_d The uniform measure on the d-sphere, page 7

$\mathcal{A}$ The space of affine lines, page 97

$\mathcal{B}(\mathbf{E})$ The Borel sets of $\mathbf{E}$, page 7

$\mathscr{C}$ The typical cell in a tessellation, page 94

$\mathcal{K}_2$ The convex bodies in $\mathbf{R}^2$, page 94

$\mathcal{F}$ A sigma algebra, page 12

$\mathfrak{m}$ A general tessellation, page 97

$\mathfrak{m}_{\text{PHT}}$ A Poisson line tessellation, page 97

$\mathrm{T}(\mathbf{X}$ A generic triangulation, page 8

$B(x_0, x_2, \ldots, x_d)$ The d-circumball touching the points $x_{0:d}$, page 36

$B(z, r)$ The closed ball in $\mathbf{R}^d$, page 7

$B_k^{(j)}[\ell]$ The ℓth largest ball in the jth comonent of size k, page 112

$\mathrm{Cone}(z, q, r)$ A Cone region in Cone Walk, page 45

$\mathrm{Disc}(zq, r)$ A step disc in Cone Walk, page 45

d The dimension of the ambient space, page 7

$d(x, y)$ The distance between two points in a metric space, page 9

$\widetilde{G}(z)$ Poisson transform, page 17

$H(u,t)$ The affine line through ut, page 97

M_n The maximum over a stochastic sequence, page 94

n_k The number of connected components of size k, page 112

$r_k^{(j)}[\ell]$ The radius of the ℓth largest ball in the jth comonent of size k, page 112

v_ρ A threshold, page 95

v_ρ A threshold, page 110

$r(C)$ The circumradius of the circumball for the compact convex set C, page 110

$z(C)$ The circumcentre of the circumball for the compact convex set C, if it is unique, page 110

$z_k^{(j)}[\ell]$ The centre of the ℓth largest ball in the jth comonent of size k, page 112

0 The origin in Euclidean spaces, page 97

$\angle xyz$ The angle between the points x, y and z, page 28

$\langle \cdot, \cdot \rangle$ The scalar product product., page 97

∂A The boundary of the set A., page 38

$\left\{ {n \atop k} \right\}$ Stirling number of the second kind, page 111

Conv($\mathbf{X}$) The convex hull of $\mathbf{X}$, page 8

Del($\cdot$) The Delaunay triangulation of a set of points, page 8

n_f Reserved for the number of faces in a tessellation., page 82

neighbours$_k(v)$ The neighbours of v in a graph within k hops, page 26

Po(λ) A Poisson-distributed random variable of rate λ., page 18

star(x) The star of a vertex in a triangulation, page 82

supp($\cdot$) The support of a measure, page 13

Vor($\cdot$) The Voronoi diagram of a set of points, page 8

Bibliography

[1] A. Aggarwal, L. J. Guibas, J. Saxe, and P. W. Shor. A linear-time algorithm for computing the Voronoi diagram of a convex polygon. *Discrete & Computational Geometry*, 4(1):591–604, 1989.

[2] F. Aurenhammer and O. Schwarzkopf. A simple on-line randomized incremental algorithm for computing higher order Voronoi diagrams. *International Journal of Computational Geometry & Applications*, 2(04):363–381, 1992.

[3] A. Baddeley. Spatial point processes and their applications. In W. Weil, editor, *Stochastic Geometry*, volume 1892 of *Lecture Notes in Mathematics*, pages 1–75. Springer Berlin Heidelberg, 2007.

[4] M. Beermann, C. Redenbach, and C. Thäle. Asymptotic shape of small cells. *Mathematische Nachrichten*, 287:737–747, 2014.

[5] M. Ben Chen, S. J. Gortler, C. Gotsman, and C. Wormser. Distributed computation of virtual coordinates for greedy routing in sensor networks. *Discrete Applied Mathematics*, 159(7):544–560, 2011.

[6] M. Bern, D. Eppstein, and F. Yao. The expected extremes in a delaunay triangulation. *International Journal of Computational Geometry & Applications*, 1(01):79–91, 1991.

[7] M. Bogdanov, O. Devillers, and M. Teillaud. Hyperbolic Delaunay complexes and Voronoi diagrams made practical. In *Proceedings of the twenty-ninth Annual Symposium on Computational Geometry*, pages 67–76. ACM, 2013.

[8] J.-D. Boissonnat and M. Teillaud. The hierarchical representation of objects: the Delaunay tree. In *Proceedings of the Second Annual Symposium on Computational Geometry*, pages 260–268. ACM, 1986.

[9] J.-D. Boissonnat and M. Teillaud. On the randomized construction of the Delaunay tree. *Theoretical Computer Science*, 112(2):339–354, 1993.

[10] J.-D. Boissonnat, O. Devillers, and S. Hornus. Incremental construction of the Delaunay triangulation and the Delaunay graph in medium dimension. In *Proceedings of the twenty-fifth annual Symposium on Computational Geometry*, pages 208–216. ACM, 2009.

[11] N. Bonichon and J.-F. Marckert. Asymptotics of geometrical navigation on a random set of points in the plane. *Advances in Applied Probability*, 43:899–942, 2011.

[12] N. Bonichon, C. Gavoille, N. Hanusse, and D. Ilcinkas. Connections between theta-graphs, Delaunay triangulations, and orthogonal surfaces. In *Graph Theoretic Concepts in Computer Science*, pages 266–278. Springer, 2010.

[13] C. Bordenave. Navigation on a Poisson point process. *The Annals of Applied Probability*, 18:708–746, 2008.

[14] P. Bose and L. Devroye. Intersections with random geometric objects. *Computational Geometry*, 10(3):139–154, 1998.

[15] P. Bose and L. Devroye. On the stabbing number of a random Delaunay triangulation. *Computational Geometry: Theory and Applications*, 36:89–105, 2006.

[16] P. Bose and P. Morin. Online routing in triangulations. *SIAM Journal on Computing*, 33:937–951, 2004.

[17] P. Bose and P. Morin. Competitive online routing in geometric graphs. *Theoretical Compututer Science*, 324(2-3):273–288, 2004.

[18] P. Bose, P. Morin, I. Stojmenović, and J. Urrutia. Routing with guaranteed delivery in ad hoc wireless networks. *Wireless networks*, 7(6):609–616, 2001.

[19] P. Bose, A. Brodnik, S. Carlsson, E. Demaine, R. Fleischer, A. López-Ortiz, P. Morin, and J. Munro. Online routing in convex subdivisions. *International Journal of Computational Geometry & Applications*, 12:283–295, 2002.

[20] P. Bose, R. Fagerberg, A. van Renssen, and S. Verdonschot. Optimal local routing on Delaunay triangulations defined by empty equilateral triangles. *arXiv preprint arXiv:1409.6397*, 2014.

[21] S. Boucheron, G. Lugosi, and P. Massart. *Concentration Inequalities - A nonasymptotic theory of independence*. Clarendon Press, Oxford, 2012.

[22] A. Bowyer. Computing Dirichlet tessellations. *The Computer Journal*, 24(2):162–166, 1981.

[23] N. Broutin, O. Devillers, and R. Hemsley. Efficiently navigating a random Delaunay triangulation. *Analysis of Algorithms*, 2014.

[24] N. Broutin, O. Devillers, and R. Hemsley. Efficiently navigating a random Delaunay triangulation. *arXiv preprint arXiv:1402.6148*, 2014.

[25] P. Calka. Tessellations. In *New perspectives in stochastic geometry*. Oxford University Press, 2010.

[26] P. Calka and N. Chenavier. Extreme values for characteristic radii of a Poisson-Voronoi tessellation. *Extremes*, pages 1–27.

[27] M. Caroli and M. Teillaud. Computing 3d periodic triangulations. In *Algorithms-ESA 2009*, pages 59–70. Springer, 2009.

[28] D. Chen, L. Devroye, V. Dujmović, and P. Morin. Memoryless routing in convex subdivisions: Random walks are optimal. *Computational Geometry*, 45(4):178 – 185, 2012. ISSN 0925-7721. doi:10.1016/j.comgeo.2011.12.005.

[29] N. Chenavier. *Valeurs etrêmes de mosaïques aléatoires*. PhD thesis, Université de Rouen, December 2013.

[30] N. Chenavier. A general study of extremes of stationary tessellations with examples. *Stochastic Processes and their Applications*, 124(9):2917 – 2953, 2014. ISSN 0304-4149.

[31] S. Cheng. A crash course on the Lebesgue integral and measure theory, 2008.

[32] K. L. Clarkson and P. W. Shor. Applications of random sampling in computational geometry, ii. *Discrete & Computational Geometry*, 4(1):387–421, 1989.

[33] K. L. Clarkson, K. Mehlhorn, and R. Seidel. Four results on randomized incremental constructions. *Computational Geometry*, 3(4):185–212, 1993.

[34] T. H. Cormen, C. E. Leiserson, et al. *Introduction to Algorithms*, volume 2. MIT press Cambridge, 2001.

[35] J. T. Cox, A. Gandolfi, P. S. Griffin, and H. Kesten. Greedy lattice animals i: Upper bounds. *The Annals of Applied Probability*, pages 1151–1169, 1993.

[36] M. de Berg, M. van Kreveld, M. Overmars, and O. Schwarzkopf. *Computational Geometry: Algorithms and Applications*. Springer-Verlag, Berlin, 1997.

[37] J.-L. De Carufel, C. Dillabaugh, and A. Maheshwari. Point location in well-shaped meshes using jump-and-walk. In *Canadian Conference on Computational Geometry*, 2011.

[38] P. M. M. De Castro and O. Devillers. Walking Faster in a Triangulation. RR-7322 RR-7322.

[39] P. M. M. De Castro and O. Devillers. Practical distribution-sensitive point location in triangulations. *Computer Aided Geometric Design*, 30(5):431–450, 2013.

[40] L. De Floriani, B. Falcidieno, G. Nagy, and C. Pienovi. On sorting triangles in a Delaunay tessellation. *Algorithmica*, 6:522–532, 1991.

[41] L. de Haan and A. Ferreira. *Extreme value theory. An introduction*. Springer Series in Operations Research and Financial Engineering. Springer, New York, 2006.

[42] J. Dean and L. A. Barroso. The tail at scale. *Communications of the ACM*, 56(2): 74–80, Feb. 2013. ISSN 0001-0782.

[43] B. Delaunay. Sur la sphere vide. *Bulletin of Academy of Sciences of the USSR*, 7 (793-800):1–2, 1934.

[44] E. D. Demaine, J. S. Mitchell, and J. O'Rourke. The open problems project: Problem 13.

[45] O. Devillers. The Delaunay hierarchy. *International Journal of Foundations of Computer Science*, 13(02):163–180, 2002.

[46] O. Devillers and S. Pion. Efficient exact geometric predicates for Delaunay triangulations. In *Proceedings of the 5th Workshop on Algorithm Engineering & Experiments*, pages 37–44, 2003.

[47] O. Devillers, S. Meiser, and M. Teillaud. Fully dynamic Delaunay triangulation in logarithmic expected time per operation. *Computational Geometry*, 2(2):55–80, 1992.

[48] O. Devillers, S. Pion, and M. Teillaud. Walking in a triangulation. *International Journal of Foundations of Compututer Science*, 13:181–199, 2002.

[49] L. Devroye, E. Mücke, and B. Zhu. A note on point location in Delaunay triangulations of random points. *Algorithmica*, 22:477–482, 1998.

[50] L. Devroye, C. Lemaire, and J.-M. Moreau. Expected time analysis for Delaunay point location. *Computational Geometry: Theory and Applications*, 29:61–89, 2004.

[51] D. Dubhashi and A. Panconesi. *Concentration of Measure for the Analysis of Randomized Algorithms*. Cambridge University Press, Cambridge, UK, 2009.

[52] R. A. Dwyer. Higher-dimensional Voronoi diagrams in linear expected time. *Discrete & Computational Geometry*, 6(1):343–367, 1991.

[53] H. Edelsbrunner. An acyclicity theorem for cell complexes in d dimensions. *Combinatorica*, 10(3):251–260, 1990.

[54] C. A. Ferro and J. Segers. Inference for clusters of extreme values. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 65(2):545–556, 2003.

[55] M. Franceschetti and R. Meester. Navigation in small-world networks: A scale-free continuum model. *Journal of Applied Probability*, pages 1173–1180, 2006.

[56] J. Gao, L. J. Guibas, J. Hershberger, L. Zhang, and A. Zhu. Geometric spanners for routing in mobile networks. *IEEE Journal on Selected Areas in Communications*, 23 (1):174–185, 2005.

[57] K. Georgoulas and Y. Kotidis. Random hyperplane projection using derived dimensions. In *Proceedings of the Ninth ACM International Workshop on Data Engineering for Wireless and Mobile Access*, pages 25–32. ACM, 2010.

[58] S. Giordano and I. Stojmenovic. Position based routing algorithms for ad hoc networks: A taxonomy. In *Ad hoc wireless networking*, pages 103–136. Springer, 2004.

[59] B. Gnedenko. Sur la distribution limite du terme maximum d'une serie aleatoire. *Annals of mathematics*, pages 423–453, 1943.

[60] S. Goudsmit. Random distribution of lines in a plane. *Reviews of Modern Physics*, 17: 321–322, 1945. ISSN 0034-6861.

[61] P. J. Green and R. Sibson. Computing Dirichlet tessellations in the plane. *The Computer Journal*, 21(2):168–173, 1978.

[62] L. Guibas and J. Stolfi. Primitives for the manipulation of general subdivisions and the computation of Voronoi. *ACM Transactions on Graphics (TOG)*, 4(2):74–123, 1985.

[63] L. J. Guibas, D. E. Knuth, and M. Sharir. Randomized incremental construction of Delaunay and Voronoi diagrams. *Algorithmica*, 7(1-6):381–413, 1992.

[64] M. Haenggi. *Stochastic geometry for wireless networks*. Cambridge University Press, 2012.

[65] M. Haenggi, J. Andrews, F. Baccelli, O. Dousse, and M. Franceschetti. Stochastic geometry and random graphs for the analysis and design of wireless networks. *Selected Areas in Communications, IEEE Journal on*, 27(7):1029–1046, September 2009. ISSN 0733-8716.

[66] L. Heinrich, H. Schmidt, and V. Schmidt. Limit theorems for stationary tessellations with random inner cell structures. *Advances in Applied Probability*, 37(1):25–47, 2005. ISSN 0001-8678.

[67] N. Henze. The limit distribution for maxima of "weighted" rth-nearest-neighbour distances. *The Journal of Applied Probability*, 19(2):344–354, 1982. ISSN 0021-9002.

[68] W. Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58:13–30, 1963.

[69] D. Hug, M. Reitzner, and R. Schneider. The limit shape of the zero cell in a stationary Poisson hyperplane tessellation. *The Annals of Probability*, 32(1B):1140–1167, 2004. ISSN 0091-1798.

[70] S. Janson. Large deviation for sums of partially dependent random variables. *Random Structures and Algorithms*, 24(3):234–248, 2004.

[71] B. Karp and H.-T. Kung. Gpsr: Greedy perimeter stateless routing for wireless networks. In *Proceedings of the 6th annual international conference on Mobile computing and networking*, pages 243–254. ACM, 2000.

[72] D. Kirkpatrick. Optimal search in planar subdivisions. *SIAM Journal on Computing*, 12(1):28–35, 1983.

[73] J. Kleinberg. Navigation in a small world. *Nature*, 406(6798):845–845, 2000.

[74] J. Kleinberg. The small-world phenomenon: An algorithmic perspective. In *Proceedings of the thirty-second annual ACM symposium on Theory of computing*, pages 163–170. ACM, 2000.

[75] R. Kleinberg. Geographic routing using hyperbolic space. In *INFOCOM 2007. 26th IEEE International Conference on Computer Communications. IEEE*, pages 1902–1909. IEEE, 2007.

[76] Y.-B. Ko and N. H. Vaidya. Location-aided routing (LAR) in mobile ad hoc networks. *Wireless Networks*, 6(4):307–321, 2000.

[77] I. Kolingerová. A small improvement in the walking algorithm for point location in a triangulation. In *22nd European workshop on computational geometry*, pages 221–224. Citeseer, 2006.

[78] G. Kozma, Z. Lotker, M. Sharir, and G. Stupp. Geometrically aware communication in random wireless networks. In *Proceedings of the Twenty-third Annual ACM Symposium on Principles of Distributed Computing*, PODC '04, pages 310–319, New York, NY, USA, 2004. ACM.

[79] E. Kranakis, H. Singh, and J. Urrutia. Compass routing on geometric networks. In *Proceedings of the 11th Canadian Conference on Computational Geometry*. Citeseer, 1999.

[80] F. Kuhn, R. Wattenhofer, and A. Zollinger. Asymptotically optimal geometric mobile ad-hoc routing. In *Proceedings of the 6th international workshop on Discrete algorithms and methods for mobile computing and communications*, pages 24–33. ACM, 2002.

[81] F. Kuhn, R. Wattenhofer, and A. Zollinger. Worst-case optimal and average-case efficient geometric ad-hoc routing. In *Proceedings of the 4th ACM international symposium on Mobile ad hoc networking & computing*, pages 267–278. ACM, 2003.

[82] C. L. Lawson. Software for C^1 surface interpolation. In J. R. Rice, editor, *Mathematical Software III*, pages 161–194. Academic Press, New York, NY, 1977.

[83] M. Leadbetter. On extreme values in stationary sequences. *Journal für Probability Theory and Related Fields*, 28(4):289–303, 1974.

[84] M. Leadbetter. Extremes and local dependence in stationary sequences. *Probability Theory and Related Fields*, 65(2):291–306, 1983.

[85] M. R. Leadbetter, G. Lindgren, and H. Rootzén. *Extremes and related properties of random sequences and processes*. Springer Series in Statistics. Springer-Verlag, New York, 1983.

[86] J. Matoušek. *Lectures on discrete geometry*. Springer, New York, 2002.

[87] M. Mauve, J. Widmer, and H. Hartenstein. A survey on position-based routing in mobile ad hoc networks. *Network, IEEE*, 15(6):30–39, 2001.

[88] C. J. H. McDiarmid. Concentration. In Habib et al., editor, *Probabilistic Methods in Algorithmic Discrete Mathematics*, pages 195–248. Springer-Verlag, 1998.

[89] R. E. Miles. Random polygons determined by random lines in a plane. *Proc. Nat. Acad. Sci. U.S.A.*, 52:901–907, 1964. ISSN 0027-8424.

[90] R. E. Miles. Random polygons determined by random lines in a plane. II. *Proceedings of the National Academy of Sciences of the United States of America*, 52:1157–1160, 1964. ISSN 0027-8424.

[91] P. R. Morin. *Online routing in geometric graphs*. PhD thesis, Carleton University, 2001.

[92] E. P. Mücke, I. Saias, and B. Zhu. Fast randomized point location without preprocessing in two-and three-dimensional Delaunay triangulations. 12:274–283, 1996. doi:10.1.1.6.6917.

[93] K. Mulmuley and S. Sen. Dynamic point location in arrangements of hyperplanes. In *Symposium on Computational Geometry*, pages 132–141, 1991.

[94] A. Okabe, B. Boots, K. Sugihara, and S. Chiu. Spatial tessellations: Concepts and applications of Voronoi diagrams. 2000.

[95] C. H. Papadimitriou and D. Ratajczak. On a conjecture related to geometric routing. *Theoretical Computer Science*, 344(1):3–14, 2005.

[96] M. Penrose. *Random Geometric Graphs*. Oxford scholarship online. Oxford University Press, 2003.

[97] Y. Plan and R. Vershynin. One-bit compressed sensing by linear programming. *Communications on Pure and Applied Mathematics*, 66(8):1275–1297, 2013.

[98] Y. Plan and R. Vershynin. Dimension reduction by random hyperplane tessellations. *Discrete & Computational Geometry*, 51(2):438–461, 2014.

[99] F. Preparata. Planar point location revisited. *International Journal of Foundations of Compututer Science*, 1, 1990.

[100] A. Rao, S. Ratnasamy, C. Papadimitriou, S. Shenker, and I. Stoica. Geographic routing without location information. In *Proceedings of the 9th annual international conference on Mobile computing and networking*, pages 96–108. ACM, 2003.

[101] S. I. Resnick. *Extreme values, regular variation, and point processes*, volume 4 of *Applied Probability. A Series of the Applied Probability Trust*. Springer-Verlag, New York, 1987.

[102] C. Y. Robert, J. Segers, C. A. Ferro, et al. A sliding blocks estimator for the extremal index. *Electronic Journal of Statistics*, 3:993–1020, 2009.

[103] R. Rossignol and L. Pimentel. Greedy polyominoes and first-passage times on random Voronoi tilings. *Electronic Journal of Probability*, 17:no. 12, 1–31, 2012. ISSN 1083-6489.

[104] R. Schneider and W. Weil. *Stochastic and Integral Geometry*. Probability and Its Applications. Springer, 2008.

[105] M. Schulte and C. Thäle. The scaling limit of poisson-driven order statistics with applications in geometric probability. *Stochastic Processes and their Applications*, 122 (12):4096–4120, 2012.

[106] R. Seidel. The upper bound theorem for polytopes: an easy proof of its asymptotic version. *Computational Geometry*, 5(2):115–116, 1995.

[107] D. A. Spielman and S.-H. Teng. Smoothed analysis of algorithms: Why the simplex algorithm usually takes polynomial time. *Journal of the ACM (JACM)*, 51(3):385–463, 2004.

[108] W. Szpankowski. *Analytic Poissonization and Depoissonization*, pages 442–519. John Wiley & Sons, Inc., 2001.

[109] The Cgal Project. *Cgal Reference Manual*. Cgal Editorial Board, 4.4 edition, 2014.

[110] R. Vershynin. Estimation in high dimensions: a geometric perspective. *arXiv preprint arXiv:1405.5103*, 2014.

[111] I. Weissman and S. Y. Novak. On blocks and runs estimators of the extremal index. *Journal of Statistical Planning and Inference*, 66(2):281–288, 1998.

[112] G. Xia. Improved upper bound on the stretch factor of Delaunay triangulations. In *Proceedings of the 27th annual ACM symposium on Computational geometry*, pages 264–273. ACM, 2011.

[113] G. Xia and L. Zhang. Toward the tight bound of the stretch factor of Delaunay triangulations. In *Proceedings of the Canadian Conference of Computational Geometry (CCCG)*, 2011.

[114] C. Yap. Towards exact geometric computation. *Computational Geometry: Theory and Applications*, 7(1):3–23, 1997.

[115] M. Yvinec. 2D triangulations. In *Cgal User and Reference Manual*. Cgal Editorial Board, 4.4 edition, 2014.

[116] B. Zhu. On Lawson's oriented walk in random Delaunay triangulations. In *Fundamentals of Computation Theory*, volume 2751 of *Lecture Notes Computer Science*, pages 222–233. Springer-Verlag, 2003.