University of Liverpool

Department of Computer Science

Geometric and Topological Methods for Applications to Materials and Data Skeletonisation

Thesis submitted in accordance with the requirements of the University of Liverpool for the degree of Doctor in Philosophy by Philip Smith

Supervisors: Dr. Vitaliy Kurlin and Prof. Igor Potapov

May 2021

"How much better to get wisdom than gold,
to choose understanding rather than silver!"

(Proverbs 16:16, New International Version.)

"Listen to advice and accept instruction,
that you may gain wisdom in the future."

(Proverbs 19:20, English Standard Version.)

"The fear of the LORD is the beginning of wisdom,
and knowledge of the Holy One is understanding."

(Proverbs 9:10, New International Version.)

Abstract

Geometric and Topological Methods for Applications to Materials and Data Skeletonisation

Philip Smith

Crystal Structure Prediction (CSP) aims to speed up functional materials discovery by using supercomputers to predict whether an input molecule can form stable crystal structures with desirable properties. The process produces large datasets where each entry is a simulated arrangement of copies of the input molecule to form a crystal. However, these datasets have little structure themselves, and it is the aim of this thesis to contribute towards simplifying and analysing such datasets.

Crystals are unbounded collections of atoms or molecules, extending infinitely in the space they lie within. As such, rigorously quantifying the geometric similarity of crystal structures, and even just identifying identical structures, is a challenging problem. To solve it, we seek a continuous, complete, isometry classification of crystals. Consequently, by modelling crystals as periodic point sets, we introduce the density fingerprint, which is invariant under isometries, Lipschitz continuous, and complete for an open and dense space of crystal structures. Such a classification will be able to identify and remove near-duplicates from these large CSP datasets, and potentially even guide future searches.

We describe how this fingerprint can be computed using periodic higher Voronoi zones. This geometric concept of concentric regions around a fixed centre characterises relative positions of points from the centre in a periodic point set. We present an algorithm to compute these zones in addition to proving key structural properties.

We later discuss research into skeletonisation algorithms, proving theoretical guarantees of the homological persistent skeleton (HoPeS), subsequently formulating and performing an experimental comparison of HoPeS with other relevant algorithms. Such algorithms, if effectively used, can be applied to large datasets including those produced by CSP to reveal the shape of the data, helping to highlight regions of interest and branches that merit further study.

Acknowledgements

I am primarily thankful to Vitaliy Kurlin for his supervision throughout the PhD, consistently imparting on me valuable help, advice, direction, opportunities and encouragement, all of which I am extremely grateful for. Moreover, I would like to thank all members of the Data Science Theory and Applications group for our useful discussions, in particular over our Friday lunches and weekly seminars, as well as for the friendships that have been formed over the years.

I am very grateful to Teresa Heiss, Herbert Edelsbrunner, Mathijs Wintraecken and also Janos Pach for our enjoyable, fruitful and insightful collaboration, which was a pleasure to be a part of and from which I learnt much.

I would like to thank the Leverhulme Research Centre for giving me the opportunity to undertake a PhD and for all of their support, particularly in providing me with opportunities to network and attend conferences.

To those that I have shared an office with during the PhD, I am very grateful for our conversations and friendships that were welcome distractions from the highs and lows of the PhD. Those informal conversations have been one of the things I've missed most during the pandemic. Finally, I will always be indebted to my friends and family, who have constantly been a source of encouragement and support, in ways both related and unrelated to the challenges of a PhD.

Contents

Chapter 1

Introduction

Solid periodic crystalline materials (crystals) are fundamental to our current way of life, with wide-ranging applications from molecular electronics [9] and photomechanics [63] to gas storage [49] and pharmaceuticals [50]. Much of the rich space of crystal structures remains unsurveyed, and so being able to efficiently explore this space may lead to significant breakthroughs in some of the most prominent current challenges; for example in carbon capture, producing superconducting materials, or improving solid-state batteries such as those found in electric vehicles.

The emerging field of Crystal Structure Prediction (CSP) aims to speed up the discovery of functional materials by running complex algorithms that take as input a molecular structure and output a large dataset of simulated crystals, where each entry consists of a particular arrangement of copies of the input molecule. These datasets can then be analysed to predict if the input molecule can form a stable crystal with desirable properties for real-world applications.

However, CSP is a slow process, producing datasets with little structure and often with many near-duplicates – entries of the dataset that are effectively identical – which lead to unnecessary repetitions of computationally expensive calculations. Yet, due to discontinuities in crystallographic groups and ambiguities in unit cell representation, rigorously describing the geometric similarity of crystal structures remains an unsolved problem.

Hence, adding structure to these somewhat disorganised datasets is the very issue that underpins this thesis, which if solved may not only speed up functional materials discovery but could even guide future research directions to focus on certain regions of crystal space.

As part of this thesis, we present two methods where geometric and topological techniques can be used to speed up the discovery of new functional materials. Firstly, we introduce the geometric concept of Voronoi zones and apply them to our research towards a continuous, complete, isometry classification of crystal structures based on geometric invariants. Secondly, we discuss our contributions to skeletonisation algorithms – algorithms that take as input a point cloud and output a skeleton where points or clusters of points

are connected by simplices. Such outputs can be used to visualise the global structure of a dataset, highlighting regions or branches of related points. This research on skeletonisation algorithms can be applied more widely to any datasets that can be represented as point clouds, not just to datasets of crystal structures.

The outline of the thesis is as follows. In the rest of this introduction, we state in Sections 1.1 - 1.3 basic definitions and concepts that will be used throughout the thesis. The introduction finishes in Section 1.4 with a formal description of the problems this thesis attempts to solve. In Chapter 2, we present and discuss a new algorithm for computing Voronoi zones of a periodic point set. This algorithm is then used in Chapter 3 in the computations of the density fingerprint – a classification of periodic point sets with proven properties. In Chapter 4, we describe our research into skeletonisation algorithms, before summarising the thesis with some concluding remarks in Chapter 5. Appendix A gives a short description of separate work on superpixels, and Appendix B lists all notations used in this thesis.

1.1 Periodic Point Sets Model Crystals

We begin with the notion of a periodic point set, which is important in Chapters 2 and 3 as we seek to classify crystals up to isometry. Periodic point sets can be used to represent crystals as formal mathematical objects. Each point corresponds to an atom or molecular centre in a crystal, and we can add labels such as chemical elements or other physical properties if needed. Periodic point sets generalise the notion of a lattice which is a set of points defined by n linearly independent vectors in $\mathbb{R}^n$.

Notation 1.1 (Point p, Vector $\vec{p}$)**.** Any point $p \in \mathbb{R}^n$ can be represented by a vector $\vec{p}$ from $0 \in \mathbb{R}^n$ to p, which has a length and a direction. We use the notation $\vec{p}$ with an arrow above for the vector to distinguish it from the point p.

Definition 1.2 (Lattice Λ, Unit Cell U)**.** For a basis of n linearly independent vectors $\vec{v_1}, \ldots, \vec{v_n} \in \mathbb{R}^n$, the integer combinations of these basis vectors form a *lattice*, $\Lambda = \{\sum_{i=1}^{n} c_i v_i \mid c_i \in \mathbb{Z}\}$. By taking linear combinations of the basis vectors with coefficients in the interval $[0, 1)$, we obtain a *unit cell* U for Λ, $U = \{\sum_{i=1}^{n} c_i v_i \mid c_i \in [0, 1)\}$. See Figure 1.1 for an example.

For dimensions $n \geq 2$, there are infinitely many bases that define the same lattice. As such, there have been several algorithms designed to deterministically select a 'good' basis for a given lattice, usually consisting of basis vectors that are short and close to orthogonal. One such basis is the Minkowski-reduced basis.

Definition 1.3 (Minkowski-reduced Basis of a Lattice)**.** Let $\Lambda \subset \mathbb{R}^n$ be a lattice with basis vectors $\vec{v_1}, \ldots, \vec{v_n}$. We say that $\{\vec{v_1}, \ldots, \vec{v_n}\}$ is a *Minkowski-reduced basis* if, for all $1 \leq i \leq n$, there is no lattice vector $\vec{v}$ with norm less than that of $\vec{v_i}$ such that the vectors $\vec{v_1}, \ldots, \vec{v_{i-1}}, \vec{v}$ can be extended to a basis of Λ.

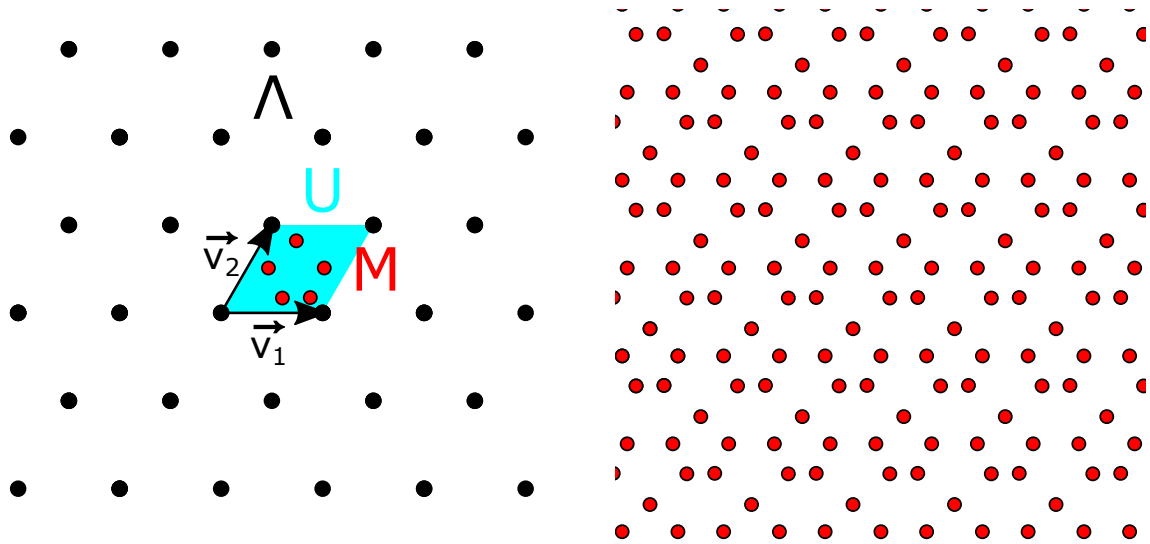

Figure 1.1: In the left image, the lattice Λ consisting of black zero-dimensional points is formed by taking the integer combinations of the basis vectors $\vec{v_1}, \vec{v_2}$. The blue unit cell contains a motif of five red zero-dimensional points. By translating each red motif point by all integer combinations of the basis vectors, we obtain the periodic set on the right, which extends infinitely throughout the plane.

Periodic point sets are formed by translating a finite subset of the unit cell by all of the lattice vectors. This finite subset is called a motif.

Definition 1.4 (Motif M)**.** Given a unit cell U of a lattice $\Lambda \subset \mathbb{R}^n$, a *motif* $M \subset U$ is a finite subset of U. The cardinality of M, $|M|$, is the number of points in the motif, and is denoted by $m = |M|$.

Definition 1.5 (Periodic Point Set A)**.** A *periodic point set* A (or periodic set for brevity) of a lattice $\Lambda \subset \mathbb{R}^n$ with motif M is defined to be the Minkowski sum $M + \Lambda$, $A = \{a + \vec{v} \mid a \in M, v \in \Lambda\}$, see Figure 1.1.

We note here the important observation of how a periodic set generalises a lattice: while a lattice will always have just one point in its motif, a periodic set can have any finite number of motif points.

In Chapter 2, it is helpful to consider a finite, bounded extension of a unit cell of a periodic set.

Definition 1.6 (k-Extended Unit Cell kU)**.** Let $A = M + \Lambda \subset \mathbb{R}^n$ be a periodic set with lattice Λ and motif M. If the basis vectors of the lattice Λ are $\vec{v_1}, \ldots, \vec{v_n}$, then $U = \{\sum_{i=1}^{n} c_i v_i \mid c_i \in [0, 1)\}$. For an integer $k \geq 1$, we define kU to be the *k-extended unit*

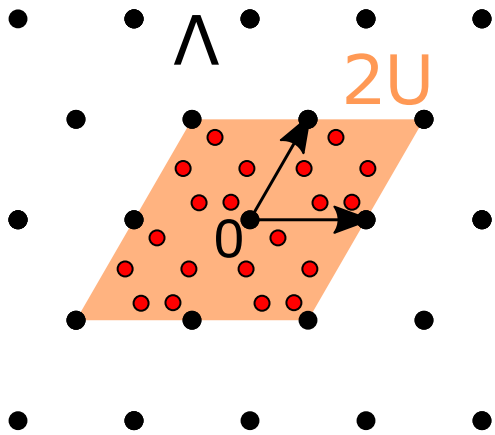

Figure 1.2: The 2-extended unit cell $2U$ of the unit cell U from Figure 1.1, extended symmetrically around the origin. Since $|M| = 5$ in Figure 1.1, there are $2^2 \cdot 5 = 20$ points of the periodic set that are contained within $2U$.

cell, $kU = \{\sum_{i=1}^{n} c_i v_i \mid c_i \in [0, k)\}$. If $m = |M|$, the number of points of A within kU is $|A \cap kU| = k^n m$. It is often more convenient, if k is even and so $k = 2j$ for some $j \in \mathbb{Z}_{\geq 1}$, to let $kU = \{\sum_{i=1}^{n} c_i v_i \mid c_i \in [-j, j)\}$, see Figure 1.2.

Periodic sets are preferable for modelling crystal structures, since atomic centres are much better defined than chemical bonds, especially bonds between molecules. However, additional information such as chemical elements or bonds can be added to the periodic set if required.

Definition 1.7 (Crystal)**.** For the purposes of this thesis, we will mathematically define a *crystal* to be a solid material that can be represented as a periodic set, for instance by placing points at the atomic locations.

Crystals, and similarly periodic sets, are invariant under all translations by lattice vectors. For a periodic set $A = M + \Lambda$, if there are no other translations that map A onto itself, then we call the unit cell U formed by the basis vectors that generate Λ a *primitive unit cell* of A. Crystals are also invariant under rigid motions, which are orientation-preserving transformations.

Definition 1.8 (Orientation of a Transformation)**.** A non-singular transformation $T\colon \mathbb{R}^n \to \mathbb{R}^n$ is *orientation-preserving* if its determinant is positive. If its determinant is negative, then we say that the transformation is *orientation-reversing*.

Definition 1.9 (Rigid Motion)**.** A *rigid motion* of $\mathbb{R}^n$ is a transformation of the space that preserves distances (see Definition 1.14) and orientation. Any rigid motion can be described as the composition of rotations and translations. For $\mathbb{R}^2$, it is possible to describe every rigid motion as the composition of just a single translation and rotation.

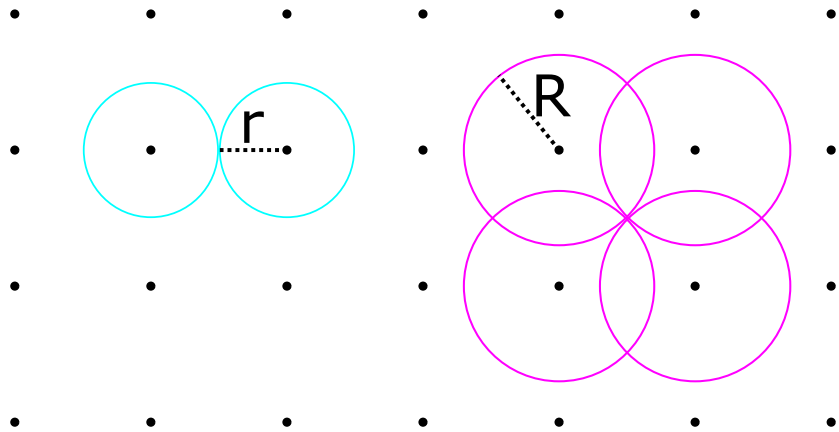

Figure 1.3: The packing and covering radii for the unit square lattice with points at (x, y), $x, y \in \mathbb{Z}$. The packing radius r of 0.5 is illustrated on the left, whilst the covering radius R of $\sqrt{2}/2$ is illustrated on the right.

Rigid motions are a subset of isometries, and for simplicity it is these transformations that we wish to classify periodic sets (and hence crystals) up to.

Definition 1.10 (Isometry)**.** An *isometry* of $\mathbb{R}^n$ is a transformation of the space that preserves distances, but not necessarily orientation. Isometric transformations include reflections in addition to translations and rotations.

For some statements relating to periodic sets, it is necessary to define certain conditions that are satisfied by the periodic sets. Often we desire periodic sets in which points cannot be arbitrarily close to each other, and where the points are spread evenly throughout the space. Such conditions are intrinsically related to the packing and covering radii.

Definition 1.11 (Packing Radius r)**.** Let $A \subset \mathbb{R}^n$ be a periodic set. The *packing radius* r of A is the largest radius such that every open ball of radius r and centre $p \in \mathbb{R}^n$ contains at most one point of the periodic set A. In other words, the packing radius is equal to half the minimum distance between two points in A (see Figure 1.3).

Definition 1.12 (Covering Radius R)**.** Let $A \subset \mathbb{R}^n$ be a periodic set. The *covering radius* R of A is the smallest radius such that every closed ball of radius R and centre $p \in \mathbb{R}^n$ contains at least one point of the periodic set A. In other words, there is no point in $\mathbb{R}^n$ that is further than R from a point in A (see Figure 1.3).

A periodic set that has packing radius $r > 0$ and covering radius $R < \infty$ is called a Delone set [16]. For example, in $\mathbb{R}^2$, the unit square lattice is a Delone set since it has $r = 0.5$ and $R = \sqrt{2}/2$, see Figure 1.3.

1.2 Metric Spaces, Point Clouds and Graphs

Definition 1.13 (Metric)**.** For a set M, a function $d\colon M \times M \to \mathbb{R}$ is called a *metric* if it satisfies, for any $x, y, z \in M$, the following three conditions:

1. Positivity: $d(x, y) \geq 0$ with equality if and only if $x = y$.
2. Symmetry: $d(x, y) = d(y, x)$.
3. The triangle inequality: $d(x, y) + d(y, z) \geq d(x, z)$.

Definition 1.14 (Metric Space (M, d))**.** A *metric space* (M, d) is any set M that is equipped with a metric d on M.

Sometimes, we will drop the metric d in the notation and simply refer to a metric space (M, d) as M. The set M can be infinite in size (for example $\mathbb{R}^2$ with the Euclidean distance) but it can also be finite.

Definition 1.15 (Point Cloud C)**.** For a metric space M, we define a *point cloud* $C \subset M$ to be a finite subset of M.

The set of points in a metric space M that are equidistant from two points in a point cloud $C \subset M$ form the bisector.

Definition 1.16 (Bisector)**.** The *bisector* between points $p, q \in \mathbb{R}^n$ is the $\mathbb{R}^{n-1}$-dimensional subspace composed of all points of $\mathbb{R}^n$ that are equidistant from p and q. We note that the vector $\vec{p} - \vec{q}$ is normal to the bisecting subspace.

A graph can be turned into a metric space by assigning appropriate weights to its edges.

Definition 1.17 (Metric and Neighbourhood Graphs)**.** A *graph* G is a finite set of vertices and edges, where an edge is simply an unordered pair of vertices. A *metric graph* is a graph that has a length or weight assigned to each edge, and the distance between two vertices is the minimum total length of any path from one vertex to another (if one exists). A *neighbourhood graph* $N(C; \epsilon)$ of a point cloud C with threshold ϵ is an example of a metric graph. C is its vertex set, and it has an edge between vertices $p, q \in C$ if the distance $d(p, q)$ between p and q in the metric space is no more than ϵ (and the length of this edge will be $d(p, q)$).

Graphs can contain cycles – a sequence of vertices, where each vertex is connected to the previous vertex by an edge, the first and last vertices are the same, and otherwise no vertex is repeated. An edge that is not part of any cycle is called a bridge.

Definition 1.18 (Bridge)**.** Let G be a graph. An open edge $e \subseteq G$ is called a *bridge* if the removal of the edge increases the number of connected components of the graph G. An edge that is not a bridge belongs to a cycle of G.

1.3 Simplices, Filtrations and Homology

Definition 1.19 (Simplex Δ, Simplicial Complex Q)**.** Let $n \geq 1$. An *n-dimensional simplex* Δ^n is the convex hull of $n+1$ linearly independent vertices and, for a standard n-simplex, whose vertices are the $n+1$ standard unit vectors, has the geometric realisation

$$\Delta^n = \{(x_0, \ldots, x_n) \in \mathbb{R}^{n+1} \mid x_0 + \ldots + x_n \leq 1, x_i \geq 0\},$$

whilst a 0-dimensional simplex is just a point. A *simplicial complex* Q with a finite vertex set C is a collection of simplices with vertices $\{v_0, \ldots, v_k\} \subseteq C$ such that: any subset (face) of a simplex, which is just a lower-dimensional simplex, is included in Q; and any pair of simplices in the complex only intersect along common faces.

Given a point cloud C in a metric space M, there are several methods of obtaining simplicial complexes with vertex set C. These include the Čech complex, the Vietoris-Rips complex, the Delone triangulation and the α-complex.

Definition 1.20 (Čech Complex $\check{\mathrm{C}}\mathrm{h}(C, M; \alpha)$, Vietoris-Rips Complex $\mathrm{VR}(C, M; \alpha)$)**.** Let $C \subset M$ be a point cloud in a metric space M. The *Čech complex* $\check{\mathrm{C}}\mathrm{h}(C, M; \alpha)$ is a simplicial complex that includes the simplex $\{v_0, \ldots, v_k \mid v_i \in C\}$ if and only if the full intersection of the set of $k+1$ balls with radius α and centres at $v_0, \ldots, v_k$ is non-empty [12, Section 4.2.3]. Similarly yet subtly different, the *Vietoris-Rips complex* $\mathrm{VR}(C, M; \alpha)$ is defined to include the simplex $\{v_0, \ldots, v_k \mid v_i \in C\}$ if and only if each pairwise intersection of the corresponding balls with radius α and centres at $v_0, \ldots, v_k$ is non-empty.

Definition 1.21 (Delone Triangulation $\mathrm{Del}(C)$, α-complex $C(\alpha)$)**.** For a point cloud $C \subset \mathbb{R}^n$, the *Delone triangulation* consists of simplices with vertices at points in C whose minimal open circumballs do not contain any other points in C. For any scale $\alpha \geq 0$, the *α-complex*, $C(\alpha) \subseteq \mathrm{Del}(C)$, consists of all simplices in the Delone triangulation of C whose circumradii are at most α.

Often, particularly in Chapter 4, to capture the shape of a point cloud, balls are 'grown' simultaneously around each point. The union of all balls at a given radius α is called the α-offset.

Definition 1.22 (α-offset X^α)**.** Let $X \subset M$ be a subspace of a metric space M. We define the α-offset of X, X^α, to be the set of all points in M that are within a distance α from a point in X. Namely, $X^\alpha = \{m \in M \mid d(m, x) \leq \alpha \text{ for some } x \in X\}$.

The following lemma from topology is fundamental to the field of Topological Data Analysis, and is equally essential to the research in Chapter 4. It states that the rather complex α-offset of a point cloud C can be replaced by certain combinatorially simpler complexes without changing the homotopy type. Definitions of basic topological notions used below can be found in [31].

Lemma 1.23 (Nerve Lemma)**.** *[17, Theorem 3.2] Let $C \subset \mathbb{R}^n$ be a finite point cloud. Then any α-offset $C^\alpha \subseteq \mathbb{R}^n$ is homotopy equivalent to both the Čech complex $\check{\mathrm{C}}\mathrm{h}(C, \mathbb{R}^n; \alpha)$ and equally the α-complex $C(\alpha) \subseteq \mathbb{R}^n$.*

Sometimes we wish to consider a family of nested simplicial complexes, for example the family of α-complexes of a point cloud C at increasing values of α. Such a family is called a filtration.

Definition 1.24 (Filtration of Simplicial Complexes)**.** A *filtration* of simplicial complexes is an indexed family of complexes $\{Q_i\}_{i \in I}$ where the index i runs over a totally ordered index set I. The family is subject to the condition that for $i, j \in I$, if $i \leq j$, then $Q_i \subseteq Q_j$. A filtration $\{Q(C; \alpha)\}$ of simplicial complexes on a point cloud C starts with $Q(C; 0) = C$ and then adds simplices of dimension at least one as α increases.

Any simplicial complex has topological features such as connected components or cycles. The homology groups algebraically encode this information.

Definition 1.25 (Homology Groups of a Complex, $H_k(Q)$)**.** Cycles of a complex Q can be algebraically written as finite linear combinations of edges (with coefficients in $\mathbb{Z}_2$) and generate the vector space Z_1 of all cycles. Meanwhile, boundaries of 2-dimensional simplices in Q are cycles of 3 edges and generate the subspace $B_1 \subseteq Z_1$. The *first homology group*, $H_1(Q)$, of a complex Q is the quotient group Z_1/B_1, with the addition of cycles as its operation and the empty cycle as its identity element. Homology groups of other dimensions can be defined similarly.

The rank of a homology group provides information on the number of a particular feature that are present in a complex Q. This is of particular importance in Chapter 4 as we investigate reconstructing graphs from point cloud samples that have the same number of cycles – i.e. the rank of the first homology group, or equally the first Betti number – as the underlying graph.

Definition 1.26 (k-th Betti number)**.** The *k-th Betti number* of a simplicial complex Q is the rank of its k-th homology group $H_k(Q)$. For example, the rank of $H_0(Q)$, namely the zeroth Betti number of the complex Q, is equal to the number of connected components present in Q.

1.4 Formal Problems and Key Contributions

Having defined some of the basic definitions that establish a foundation for the rest of this thesis, we can now introduce the two main problems we will go on to tackle. Firstly, we desire a classification of crystals up to isometries, which we formulate in terms of periodic sets which model crystals.

Problem 1 (A Continuous, Complete, Isometry Classification of Periodic Sets)**.** We desire a classification of periodic sets that satisfies the following three conditions:

I *Invariance under isometries:* any two periodic sets that are isometric should be classified identically.

II *Continuous under perturbations:* if a periodic set undergoes a small perturbation, the distance between the old and new classifications should be small.

III *Completeness:* no two non-isometric periodic sets should have the same classification.

We prove in Lemma 3.4, Theorem 3.10 and Theorem 3.14 that the density fingerprint map (Definition 3.5) satisfies Conditions I and II of the above problem, and is complete for an open and dense space of periodic sets. Voronoi zones – the subject matter of Chapter 2 – enable us to compute the density fingerprint as described in Section 3.5.

Secondly, it is the data skeletonisation problem that desires guarantees on the reconstruction of a graph G from a point cloud C that has been sampled from it.

Problem 2 (Data Skeletonisation Problem)**.** Given a noisy point cloud C sampled from a graph G in a metric space M, can you find conditions on G and C such that the reconstructed graph G' has the same first homology group as G ($H_1(G') \cong H_1(G)$) and geometrically approximates G in the sense that $G' \subset G^\alpha$ and $G \subset (G')^\alpha$ for a suitable parameter α depending on G and C?

Theorems 4.21, 4.28 and 4.32 state optimality and reconstruction guarantees of the skeleotisation algorithm HoPeS, whilst Sections 4.7 - 4.9 describe a detailed comparison of three skeletonisation algorithms on real and synthetic datasets.

Chapter 2

Voronoi Zones of a Periodic Set

(This chapter is based on the paper "A practical algorithm for higher Voronoi zones of periodic point sets" authored by P.S. and V. Kurlin and is currently under review.)

The focus of this chapter is on the geometric concept of Voronoi zones (Definition 2.4) of a periodic set (Definition 1.5). These structures characterise the relative positions of points from a fixed centre in a periodic set, including points that are more distant in addition to the nearest neighbours. Since periodic sets can be used to model all crystals (Definition 1.7), this research is applicable to the study of crystals as properties such as a crystal's energy are similarly dependent on interatomic distances between both distant atoms and close neighbours. A detailed explanation of precisely how Voronoi zones can be used to assist the discovery of new functional materials is described in Section 3.5 of Chapter 3.

To introduce Voronoi zones, we must start with the related classical concept of a Voronoi domain introduced by Georges Voronoi in 1908 [61].

Definition 2.1 (Voronoi Domain $V(C;p)$)**.** Let $C \subset \mathbb{R}^n$ be a point cloud (Definition 1.15). The *Voronoi domain* of a point $p \in C$, $V(C;p)$, is defined to be the set of all points $x \in \mathbb{R}^n$ that are no closer to any other point of C than p. Namely, $V(C;p) = \{x \in \mathbb{R}^n \mid d(x,p) \leq d(x,q) \text{ for all } q \in C, q \neq p\}$.

Although any distance can be used in the definitions of a Voronoi domain and its related concepts, throughout this chapter we will simply be using the Euclidean distance. As such, we note that the Voronoi domain of a point is closed and convex. Of course, it is straightforward to extend Voronoi domains to k-th degree Voronoi domains.

Definition 2.2 (k-th Degree Voronoi Domain $V_k(C;p)$)**.** Let $C \subset \mathbb{R}^n$ be a point cloud. The *k-th degree Voronoi domain* of a point $p \in C$, $V_k(C;p)$, is defined to be the set of all points $x \in \mathbb{R}^n$ that have no more than $k-1$ points of C closer to x than p. Namely, $V_k(C;p) = \{x \in \mathbb{R}^n \mid \|x-q\|_2 < \|x-p\|_2 \text{ for at most } k-1 \text{ points } q \in C\}$. We set $V_0(C;p) = \emptyset$ for convenience, and we note that $V_1(C;p) = V(C;p)$.

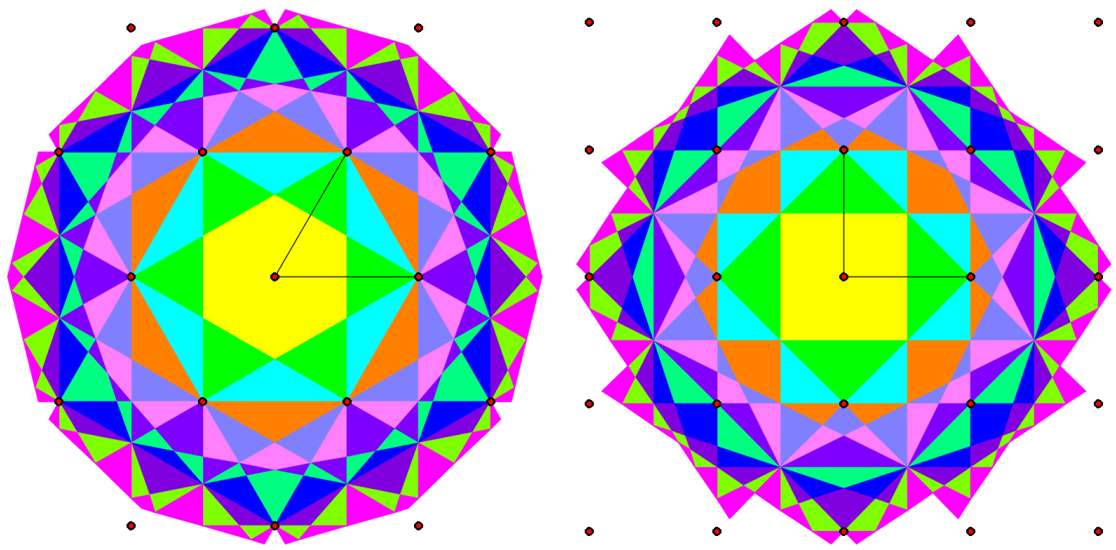

Figure 2.1: The first 12 Voronoi zones of the origin in the hexagonal lattice (left) and square lattice (right), where lattice points are portrayed as small red dots. The k-th Voronoi zone is represented by polygons of the same colour.

Unlike k-th order Voronoi domains of a point cloud C (which partition $\mathbb{R}^n$ so that points $x, y \in \mathbb{R}^n$ are in the same k-th order Voronoi domain if their corresponding sets of k nearest neighbours in C match) which tile $\mathbb{R}^n$ [20], k-th degree Voronoi domains overlap and so do not form a tiling. k-th degree Voronoi domains are again closed, but they are not necessarily convex. However, they are star convex.

Definition 2.3 (Star Convexity)**.** A set $X \subset \mathbb{R}^n$ is *star convex* if there exists a point $x_0 \in X$ such that any line segment from x_0 to a point $x \in X$ is contained within X.

Indeed, for any $x \in V_k(C;p)$, any point on the line segment from p to x must also have p among its k-th nearest neighbours in C, and so is contained within $V_k(C;p)$. The k-th degree Voronoi domain contains the $(k-1)$-th degree Voronoi domain, and it is by taking the relative complement of successive k-th degree Voronoi domains that we obtain Voronoi zones.

Definition 2.4 (k-th Voronoi Zone $Z_k(C;p)$)**.** Let $C \subset \mathbb{R}^n$ be a point cloud. For $k \in \mathbb{Z}_{\geq 1}$, the *$k$-th Voronoi zone* of a point $p \in C$, $Z_k(C;p)$, is defined to be the relative complement $V_k(C;p) \setminus V_{k-1}(C;p)$. We set $Z_0(C;p) = \emptyset$, and note that $Z_1(C;p) = V_1(C;p) = V(C;p)$.

A point $x \in \mathbb{R}^n$ has p as its unique k-th nearest neighbour if and only if x is in the interior of $Z_k(C;p)$. We also note that, for $k \geq 2$, $Z_k(C;p)$ is neither open nor closed, since its interior faces (faces such that any point on the face can be connected to p by a line

segment whose interior does not intersect $Z_k(C;p)$) are open whilst the remaining exterior faces are closed. Our work focuses on higher Voronoi zones of periodic sets (Definition 1.5), an example of which can be seen in Figure 2.1.

2.0.1 Review of Related Work

The first algorithm to compute Voronoi domains for general periodic sets of points was described in [15], but did not consider higher degree Voronoi domains. The algorithm, which computed their dual – periodic Delone triangulations or mosaics – was recently improved in [46].

Previously, higher Voronoi zones of order $k \geq 2$ were considered in the periodic setting only for lattices [14], referring to them as Brillouin zones. This is because they are applied to reciprocal lattices of a crystal, which are obtained from the crystal's lattice (in direct space) via a Fourier transform, and can be seen in the crystal's powder diffraction pattern. Such a pattern is primarily impacted by the crystal's periodicity, and so it is only ever lattices and not more general periodic sets that are produced. It is the extension of Voronoi zones to periodic sets in direct space, and an algorithm to compute them, that is the novel work of this chapter.

An algorithm visualising higher Voronoi zones of a lattice was introduced by Andrew et al. [3]. The algorithm simply assigns each point in a square or cubical grid to the appropriate k-th Voronoi zone. We substantially improve on this work in two ways: firstly, as previously mentioned, we generalise the input so that higher Voronoi zones of general periodic sets with motifs of more than one point can be computed; and secondly we compute precisely the polytopes that comprise each zone, enabling our outputs to be used not just for visualisations but also for accurate computations (for example in the computations of $\varphi_k^A(t)$ in Theorem 3.21 of Chapter 3).

2.0.2 Contributions and Chapter Outline

The main contributions of this chapter are as follows:

- For any periodic set $A \subset \mathbb{R}^n$, the k-th Voronoi zone is introduced in Definition 2.4 and structurally described in Theorem 2.6.
- The sum of the volumes of the k-th Voronoi zones $Z_k(A;a)$ over all points a in the motif M of a periodic set $A \subset \mathbb{R}^n$ is independent of k, as stated and proved in Theorem 2.12.
- The practical algorithm of Section 2.2 computes all Voronoi zones up to a given order k of a point a in a periodic set $A \subset \mathbb{R}^n$, $n \leq 3$. By Theorem 2.18, it has polynomial complexity in the number $m = |M|$ of points in the motif M of A, and has been implemented in C++ [54].

- Experimental analysis of the algorithm in Section 2.3 illustrates the time complexity bound, as well as showing the dependence of geometric features like the number of polytopes that comprise a k-th Voronoi zone on the order k or the motif size m.

Chapter outline: Section 2.1 states and proves structural properties of k-th Voronoi zones, in particular Theorems 2.6 and 2.12. Section 2.2 provides a practical algorithm to compute the k-th Voronoi zones up to any finite k for a periodic set A. The algorithm is experimentally analysed in Section 2.3, which also includes visualisations of higher Voronoi zones for several periodic sets. Section 2.4 discusses novelty and open problems relating to Voronoi zones.

2.1 The Geometric Structure of Voronoi Zones

The main results in this section are Theorem 2.6 describing the spherical nature of the k-th Voronoi zone around its centre, and Theorem 2.12 that states that the volume of the k-th Voronoi zone (summed over motif points) is independent of the order k. Hence, all coloured regions in Figure 2.1 have the same area, which might seem surprising at first glance.

2.1.1 Spherical Projection

In order to prove Theorem 2.6 about the structure of a Voronoi zone, it is helpful to know which zone a point belongs to. This is the zone index of a point.

Definition 2.5 (Zone Index $\mathrm{ind}(x; C; p)$)**.** Let $C \subset \mathbb{R}^n$ be a point cloud, and for $p \in C$, let $b(C; p)$ be the set of all bisectors (Definition 1.16) between p and every other point of C. For any $x \in \mathbb{R}^n$, consider the half-open line segment $[p, x)$ joining p to x (but not including x). Then the *zone index* of x relative to C and p is $\mathrm{ind}(x; C; p) = i + 1$, where i is the number of bisectors in $b(C; p)$ that intersect the line segment $[p, x)$.

For any point x in the closed Voronoi domain $V(C; p)$, the half-open segment $[p, x)$ belongs to the interior of $V(C; p)$ and hence does not intersect any of the bisectors of $b(C; p)$. Therefore, all points $x \in V(C; p)$ have zone index $\mathrm{ind}(x; C; p) = 1$. Each time we cross a bisector of $b(C; p)$ as we move radially further away from the central point p, the zone index jumps by at least one. This fact can be used to prove the following spherical structural description of Voronoi zones of periodic sets.

Theorem 2.6 (Structure of Voronoi Zones)**.** *For any point a in a periodic set $A \subset \mathbb{R}^n$, the closure of the k-th Voronoi zone $Z_k(A; a)$ is the union of convex polytopes whose interior points have zone index k. Moreover, this closure is spherical in the sense that the radial projection* $\mathrm{closure}\{Z_k(A; a)\} \to S^{n-1}$ *(where S^{n-1} is centred at a) is surjective.*

Proof. First, we prove that any point $x \in \mathbb{R}^n$ that has a as its unique k-th nearest neighbour has zone index $\text{ind}(x; A; a) = k$. Consider the line segment $l(s)$ from a to x parameterised by s so that we have $l(0) = a$ and $l(1) = x$. While $l(s)$ lies strictly within $V_1(A; a)$, the point $l(s)$ has a as its unique closest neighbour, and $\text{ind}(l(s); A; a) = 1$.

As we increase the parameter s, the zone index $\text{ind}(l(s); A; a)$ increases by one each time we intersect a bisector separating a from another point in A. If we cross i bisectors, $\text{ind}(l(s); A; a)$ would equal $i + 1$. As $l(1) = x$ has a as its unique k-th nearest neighbour, $l(s)$ must intersect $k - 1$ bisectors as s increases from 0 to 1. Therefore, $\text{ind}(x; A; a) = k$.

Hence the closure of $Z_k(A; a)$ is a finite union of polytopes whose interior points have zone index k. Since the polytopes' faces are bisectors between the central point a and other points in the periodic set A, each individual polytope is convex. Interior faces of these polytopes consist of points with zone index at most $k - 1$, whilst the remaining exterior faces have zone index k.

To prove the second clause of the theorem, note that any straight ray R emanating from a either contains points of zone index k, and thus intersects $Z_k(A; a)$, or R must jump from zone index $k' < k$ to zone index $k'' > k$ as we move along R away from a. This can only happen if R passes through an intersection of multiple bisectors. However, at this intersection point w, any small neighbourhood of w contains points of all intermediate indices from k' to k'', including k, since no two bisectors can coincide. Hence w is contained in the closure of $Z_k(A; a)$, and we can conclude that the image of the closure of $Z_k(A; a)$ under the radial projection covers the whole sphere S^{n-1}. □

2.1.2 Constant Volume

Theorem 2.12 states that the sum of the volumes of the k-th Voronoi zones over all points in the motif of a periodic set is independent of the order k. The definitions, statements and proofs in this subsection follow a similar pattern to that found in [19], where they prove the same result for lattices. The key difference here is that we have extended the result to more general periodic sets. The proof works by finding a bijection (Lemma 2.11) between regions of the Voronoi domain of the origin in the lattice Λ of a periodic set A (the k-th Voronoi subdomain) and regions of the k-th Voronoi zones (k-th Voronoi subzones) of motif points of A.

Definition 2.7 (k-th Voronoi subdomain $V^{(k)}(A; 0)$)**.** Let $A \subset \mathbb{R}^n$ be a periodic set with lattice Λ. Then the *k-th Voronoi subdomain*, $V^{(k)}(A; 0)$, is the open subdomain strictly within $V(\Lambda; 0)$ (and so has a unique closest point in Λ) consisting of all points that have a unique k-th nearest neighbour in the periodic set A.

Definition 2.8 (k-th Voronoi subzone $Z^{(k)}(A; a)$)**.** Let $A \subset \mathbb{R}^n$ be a periodic set with lattice Λ. Then the *k-th Voronoi subzone*, $Z^{(k)}(A; a)$, is the open subdomain strictly within $Z_k(A; a)$ (and so has a unique k-th closest point in A) consisting of all points that have a unique closest lattice point $v \in \Lambda$.

The proof of Theorem 2.12 requires that each point $a \in A$ should be identified with just one lattice point. The natural thought is to identify a point $a \in A$ with the lattice point v whose Voronoi domain $V(\Lambda; v)$ it lies within. Yet Voronoi domains are closed and can overlap at their boundaries. Hence a point that is equidistant from its two closest lattice points will be contained in both of the corresponding Voronoi domains. To overcome this, we define half-open Voronoi domains that tile $\mathbb{R}^n$ without any overlap.

Definition 2.9 (Half-open Voronoi domain $V^h(\Lambda; 0)$)**.** For a lattice $\Lambda \subset \mathbb{R}^n$, we define the *half-open Voronoi domain*, $V^h(\Lambda; 0)$, to be the union of the interior of $V(\Lambda; 0)$ with one representative of each orbit of the points on the boundary of $V(\Lambda; 0)$, where two points are equivalent if they are related to each other by a translation by a lattice vector.

By using half-open Voronoi domains in the place of classic Voronoi domains, the following piecewise shift function is well-defined.

Definition 2.10 (Piecewise shift $f_k(x)$)**.** Let $A \subset \mathbb{R}^n$ be a periodic set with lattice Λ. By Definition 2.7, any point $x \in V^{(k)}(A; 0)$ has a unique k-th nearest neighbour $a_k \in A$. We have that $a_k \in V^h(\Lambda; v_k)$ for a unique lattice point $v_k \in \Lambda$, and so we define the *piecewise shift* to be $f_k(x) = x - \vec{v_k}$.

Lemma 2.11. *The function*

$$f_k : V^{(k)}(A; 0) \to \bigcup_{a \in A \cap V^h(\Lambda; 0)} Z^{(k)}(A; a)$$

from Definition 2.10 is a bijection.

Proof. We first show that the image of $V^{(k)}(A; 0)$ under the function f_k is in the union of k-th Voronoi subzones $\bigcup_{a \in A \cap V^h(\Lambda; 0)} Z^{(k)}(A; a)$. Any $x \in V^{(k)}(A; 0)$ has a unique k-th nearest neighbour $a_k \in A$ by Definition 2.7. a_k is covered by a unique half-open Voronoi domain $V^h(\Lambda; v_k)$ for some $v_k \in \Lambda$, since the half-open Voronoi domains tile $\mathbb{R}^n$ without overlap. So we have that $f_k(x) = x - \vec{v_k}$. Since $x \in V^{(k)}(A; 0)$ has a unique closest lattice point and a unique k-th closest point in A, so does $f_k(x)$ as its a translation of x by a lattice vector. The unique k-th nearest neighbour of $f_k(x)$ among points in A is $a_k - \vec{v_k}$, which is in the half-open Voronoi domain $V^h(\Lambda; v_k - \vec{v_k}) = V^h(\Lambda; 0)$, and so $f_k(x) \in \bigcup_{a \in A \cap V^h(\Lambda; 0)} Z^{(k)}(A; a)$.

It remains to prove that f_k is injective and surjective. To prove that it is injective, let $x, x' \in V^{(k)}(A; 0)$ have unique k-th nearest neighbours $a_k, a'_k \in A$ that lie in half-open Voronoi domains $V^h(\Lambda, v_k)$ and $V^h(\Lambda, v'_k)$ respectively. If $v_k = v'_k$, then $f_k(x) - \overrightarrow{f_k(x')} = x - \vec{x'}$, which implies $f_k(x) = f_k(x')$ if and only if $x = x'$. If $v_k \neq v'_k$, then $f_k(x)$ and $f_k(x')$ lie in different half-open Voronoi domains, and since these regions do not overlap, $f_k(x) \neq f_k(x')$. Hence f_k is injective.

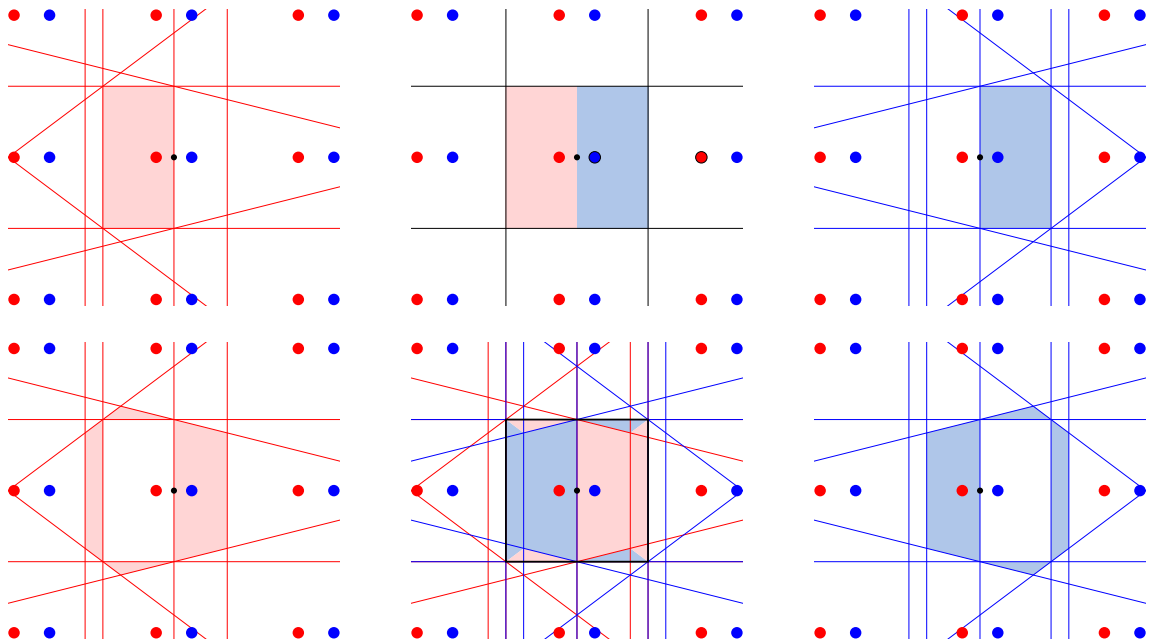

Figure 2.2: A periodic set with a two-point motif (blue and red points). The origin is depicted by a small black dot. **Top left:** the first Voronoi zone of the red point. **Top middle:** both first Voronoi zones of the red and blue points form the Voronoi domain $V(\Lambda;0)$. **Top right:** the first Voronoi zone of the blue point. **Bottom left:** the second Voronoi zone of the red point. **Bottom middle:** both second Voronoi zones of the red and blue points form the Voronoi domain $V(\Lambda;0)$ after applying lattice vector translations to the polygons that form the second Voronoi zones, see Lemma 2.11 and Theorem 2.12. **Bottom right:** the second Voronoi zone of the blue point.

To prove that f_k is surjective, let $x \in Z^{(k)}(A;a)$ for some $a \in A \cap V^h(\Lambda;0)$. Then x has a as its unique k-th nearest neighbour in A, and has some $v_k \in \Lambda$ as its unique closest lattice point. Consider the point $x - \vec{v_k}$. This point has a unique k-th nearest neighbour $a - \vec{v_k} \in A$, and has $v_k - \vec{v_k} = 0$ as its unique closest lattice point. Hence $x - \vec{v_k} \in V^{(k)}(A;0)$. Since $a - \vec{v_k}$ has $-v_k$ as its unique closest lattice point, $f_k(x - \vec{v_k}) = x - \vec{v_k} + \vec{v_k} = x$, and so f_k is surjective. □

It is Lemma 2.11 that does the majority of the heavy lifting in the proof of Theorem 2.12.

Theorem 2.12 (Voronoi Zone Volumes)**.** *Let $A \subset \mathbb{R}^n$ be a periodic set such that $A = M + \Lambda$ for a motif M and a lattice Λ. Then the sum of the volumes of the k-th Voronoi zones $Z_k(A;a)$ over all motif points $a \in M$ is independent of the order k.*

Proof. The statement is trivially true for $k = 0$, so let $k \geq 1$. Since Lemma 2.11 says that the k-th Voronoi subdomain, $V^{(k)}(A;0)$, and the union of k-th Voronoi subzones over

points of A in the half-open Voronoi domain $V^h(\Lambda;0)$, $\bigcup_{a\in A\cap V^h(\Lambda;0)} Z^{(k)}(A;a)$, are related by a bijection consisting of piecewise translations of nonzero measure regions, we can conclude that these two domains have the same volume. Since these two domains differ from the Voronoi domain $V(\Lambda;0)$ and the union of k-th Voronoi zones over points of A in the half-open Voronoi domain by measure zero sets, it also holds that these two latter domains have equal volume too. Motif points and points in the set $A \cap V^h(\Lambda;0)$ each contain one representative of each orbit of A. Hence we have

$$\mathrm{Vol}[V(\Lambda;0)] = \sum_{a\in A\cap V^h(\Lambda;0)} \mathrm{Vol}[Z_k(A;a)] = \sum_{a\in M} \mathrm{Vol}[Z_k(A;a)],$$

showing that the sum of the volumes of the k-th Voronoi zones over all motif points $a \in M$ is independent of the order k, see Figure 2.2. □

For a periodic set $A = M + \Lambda \subset \mathbb{R}^n$, we can generalise Theorem 2.12 to any integrable function $\mu : \mathbb{R}^n \to \mathbb{R}$ that is Λ-periodic, i.e. $\mu(x + \vec{v}) = \mu(x)$ for all $x \in \mathbb{R}^n$, $v \in \Lambda$.

Corollary 2.13. *Let $A = M + \Lambda \subset \mathbb{R}^n$. For any integrable function $\mu : \mathbb{R}^n \to \mathbb{R}$ that is Λ-periodic, the integral over the union of the k-th Voronoi zones $Z_k(A;a)$ of all motif points $a \in A$ is independent of the order k.*

Proof. We can follow the exact same logic as in the proof of Theorem 2.12. In fact, Theorem 2.12 is just the specific case when, for $x \in \mathbb{R}^n$, $\mu(x) = 1$. □

2.2 A Practical Algorithm to Compute Voronoi Zones

We describe here a practical algorithm to compute all Voronoi zones up to a given order k for a point a in a periodic set $A \subset \mathbb{R}^n$, where the dimension $n = 2$ or 3. The **input** of the algorithm consists of the following, where all coordinates are rational to allow for practical computations:

- A lattice $\Lambda \subset \mathbb{R}^n$ given by a basis $\{\vec{v}_1, \ldots, \vec{v}_n\} \in \mathbb{Q}^n$.
- A finite motif $M \subset U$, where U is the unit cell of Λ, consisting of $m = |M|$ points given by their fractional coordinates (coefficients in the basis of Λ).
- A point $a \in M$ selected to be the centre of the Voronoi zone $Z_k(A;a)$, where A is the periodic set formed by the Minkowski sum $A = M + \Lambda$. Since we are interested in applications to crystals which are invariant under rigid motions, we similarly study periodic sets up to rigid motions, and so we can perform a translation on the periodic set A so that the point a is at the origin $0 \in \mathbb{R}^n$.
- An integer $k \geq 1$ which determines the highest order Voronoi zone to be computed.

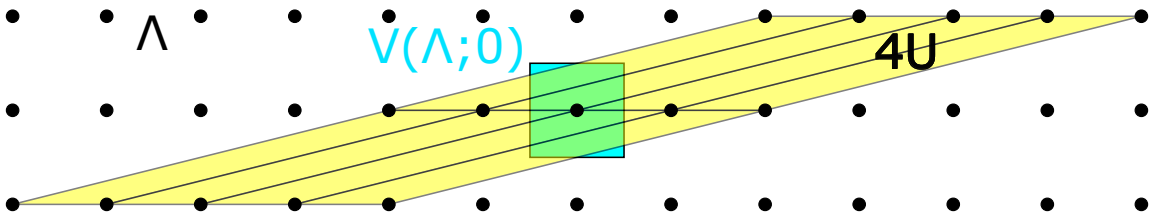

Figure 2.3: When Λ is the square lattice generated by the basis vectors $\vec{v_1} = (1, 0)$, $\vec{v_2} = (4, 1)$, the 4-extended unit cell $4U$ (yellow) is insufficient to cover the Voronoi domain $V(\Lambda; 0)$ (blue), see Lemma 2.14.

The **output** is the first k Voronoi zones of the origin in the periodic set A, $Z_i(A; 0)$, $1 \leq i \leq k$. Each Voronoi zone is the union of a set of polytopes defined by:

- Vertices: arbitrarily ordered points in $\mathbb{R}^n$.
- Edges: unordered pairs of vertices indexed above.
- Two-dimensional faces (for $n = 3$): cyclically ordered lists of edges indexed above.

2.2.1 Justification of the Minkowski Basis Reduction

In the first stage of the algorithm, a basis reduction is performed. This is needed due to Lemma 2.14 which states that for any lattice there is always a unit cell such that no fixed k-extension of the unit cell (Definition 1.6) covers even the Voronoi domain $V(\Lambda; 0)$, let alone higher Voronoi zones.

Lemma 2.14. *For any fixed $k \geq 1$ and for any lattice $\Lambda \subset \mathbb{R}^n$, there always exists a set of basis vectors such that no k-extension of the corresponding unit cell U covers the Voronoi domain $V(\Lambda; 0)$, see Figure 2.3.*

Proof. One can choose a basis $\{\vec{v_1}, \ldots, \vec{v_n}\}$ of Λ in such a way that the nearest neighbour of the origin $0 \in \mathbb{R}^n$ in $\Lambda \setminus \{0\}$ is the point v_1. Let w be the midpoint of the line segment from the origin 0 to v_1, which is on the boundary of $V(\Lambda; 0)$, and so would have to be covered by the k-extension of any unit cell. Consider the basis $\{\vec{v}_1 + (2k + 1)\vec{v}_2, \vec{v_2}, \ldots, \vec{v_n}\}$. The k-extension kU of the unit cell U spanned by this new basis does not cover w. Indeed, U must be extended by $-(k + 1)\vec{v_2}$ in order to cover w. Hence at least the $(k + 1)$-extension of U is needed. □

In [30], they prove Lemma 2.15 stating that the 2-extension of the unit cell of a Minkowski-reduced basis (Definition 1.3) covers the Voronoi domain $V(\Lambda; 0)$ of the origin. We have used this lemma to prove Lemma 2.16, which is in turn used in the proof of Lemma 2.17 extending Lemma 2.15 to any higher order Voronoi zone. This justifies the

use of the Minkowski basis reduction in Stage 1 of the algorithm computing higher Voronoi zones of periodic sets.

Lemma 2.15. *[30, Appendix A.1] For a unit cell U of a lattice $\Lambda \subset \mathbb{R}^n$, $n \leq 3$, with a Minkowski-reduced basis $\{\vec{v}_1, \ldots, \vec{v}_n\}$, the 2-extended unit cell $2U$ strictly contains the Voronoi domain $V(\Lambda; 0)$.*

Lemma 2.16. *For a unit cell U of a lattice $\Lambda \subset \mathbb{R}^n$, $n \leq 3$, with a Minkowski-reduced basis $\{\vec{v}_1, \ldots, \vec{v}_n\}$, let Λ_i, $i \geq 1$ an integer, be the set of all points in Λ on the boundary of the $2i$-extended unit cell $2iU$ (extended symmetrically around the origin). Then any point $x \in \mathbb{R}^n \setminus 2iU$ is closer to at least one point of Λ_i than to $0 \in \mathbb{R}^n$.*

Proof. Set $i = 1$. By Lemma 2.15, the Voronoi domain $V(\Lambda; 0)$ is strictly within $2U$. Consider a point x on the boundary of $2U$. x belongs to the Voronoi domain $V(\Lambda; v)$ of a lattice point $v \in \Lambda \setminus \{0\}$. $2U + v$ must strictly contain $V(\Lambda; v)$, and as $x \in V(\Lambda; v)$, $2U + v$ must strictly contain x. Since x is also on the boundary of $2U$, for $2U + v$ to strictly contain x, we must also have $0 \in 2U + v$. From this we can deduce that $v \in \Lambda_1$. Therefore, any point on the boundary of $2U$ is closer to at least one point of Λ_1 than to 0, which implies that any point $x \in \mathbb{R}^n \setminus 2U$ is similarly closer to at least one point of Λ_1 than to 0. For $i > 1$, consider the lattice $i\Lambda$ with a Minkowski-reduced basis $\{i\vec{v}_1, \ldots, i\vec{v}_n\}$ and unit cell iU. The above argument still holds for this new lattice, meaning that any point $x \in \mathbb{R}^n \setminus 2iU$ is closer to at least one point of $i\Lambda_1$ than to 0. But since $i\Lambda_1 \subset \Lambda_i$, the result follows. □

Lemma 2.17. *For a unit cell U of a lattice $\Lambda \subset \mathbb{R}^n$, $n \leq 3$, with a Minkowski-reduced basis $\{\vec{v_1}, \ldots, \vec{v_n}\}$, its $2k$-extension $2kU$ (symmetrically extended around $0 \in \mathbb{R}^n$) covers the k-th Voronoi zone $Z_k(A; 0)$ for any periodic set $A = M + \Lambda$.*

Proof. Since $Z_k(A; 0) \subseteq V_k(A; 0)$, it suffices to prove that $V_k(A; 0) \subseteq 2kU$ when A is a lattice (and so has just one point in its motif M) since increasing the number of points in M can only make $V_k(A; 0)$ smaller. Consider any point $x \in \mathbb{R}^n \setminus 2kU$. By applying Lemma 2.16 for $1 \leq i \leq k$, we can conclude that there are at least k points of $\Lambda \setminus \{0\}$ that are closer to x than to 0. Then x must be outside the k-th degree Voronoi domain $V_k(\Lambda; 0)$, implying that $\mathbb{R}^n \setminus 2kU \subseteq \mathbb{R}^n \setminus V_k(\Lambda; 0)$. Hence $V_k(A; 0) \subseteq 2kU$. □

2.2.2 The Stages of the Algorithm in Dimension Two

The stages of the algorithm computing the first k Voronoi zones of a point in a periodic set A are outlined below for dimension $n = 2$, while we discuss how this can be extended to dimension $n = 3$ in Subsection 2.2.3.

Stage 1: reduction of the unit cell. The given basis of the lattice Λ is reduced to a Minkowski-reduced basis.

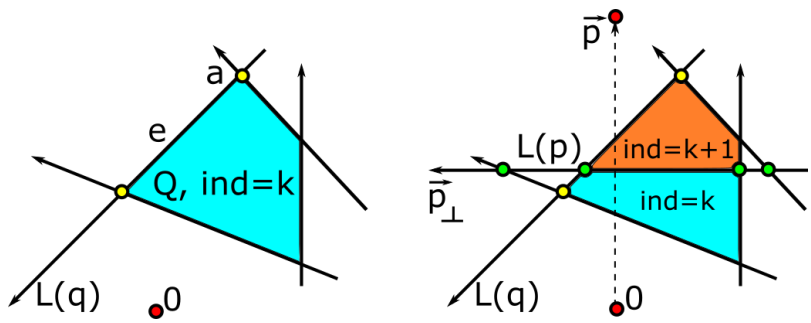

Figure 2.4: **Left:** the blue polygon Q is a convex piece of $\mathbb{R}^2$ with boundaries corresponding to bisectors between the origin and points in the periodic set. **Right:** a new bisector $L(p)$ intersects Q, in particular creating two new intersection points on the boundary of Q. The polygon Q is split into two smaller polygons, with the zone index of the polygon further from the origin increasing by one.

Stage 2: sorting points from the extended motif. By Lemma 2.17, the k-th Voronoi zone is covered by the $2k$-extended unit cell $2kU$ of the unit cell U of the Minkowski-reduced basis. If the motif $M \subset U$ has $m = |M|$ points, then the number of points in the extended motif $M_{2k} = A \cap 2kU$ is $|M_{2k}| = (2k)^n m$. These points are inserted into a balanced binary tree whose key for comparison is the distance of each point to the origin. This tree and subsequent trees in the algorithm are implemented using the multimap structure in C++ for fast searching and insertions.

Stage 3 forms the main loop processing all points of the balanced binary tree (except the origin) from Stage 2 in increasing order of their distance to the origin $0 \in \mathbb{R}^2$.

Stage 3: looping over points. For a point $p \in \mathbb{R}^2$, the point $0.5p$ is at the midpoint of the line segment $[0, p] \subset \mathbb{R}^2$. The bisecting line (Definition 1.16) $L(p)$ between 0 and p has parametric equation $0.5\vec{p} + t_p\vec{p}_\perp$ where $t_p \in \mathbb{R}$ and $\vec{p}_\perp$ is a vector orthogonal to $\vec{p}$ and anticlockwisely oriented relative to the origin.

For the first time through Stage 3, we start with the polygon $2kU$, and take the first point p_1 of the binary tree. The bisector $L(p_1)$ intersects the boundary of $2kU$ in two locations a and b, splitting the polygon $2kU$ into two. Points a and b are placed into a balanced binary tree $T(p_1)$ whose key for comparison is the parameter t_{p_1} of the bisector $L(p_1)$. The two polygons that $2kU$ is split into are stored as additional information relating to the edge $e \subset L(p_1)$ from a to b, assigning this information to a if $a < b$ in $T(p_1)$, otherwise to b. The polygon containing the origin is given a zone index of 1, whilst the polygon not containing the origin is given a zone index of 2.

For any subsequent point p in the balanced binary tree from Stage 2, the intersection points (that lie within $2kU$) of the bisector $L(p)$ with all previous bisectors $L(q)$ as well as the polygon $2kU$ are found. After finding a new intersection point a of $L(p)$ with a previous

bisector $L(q)$, we insert the intersection point a into the balanced binary trees $T(p)$ and $T(q)$ according to the parameters t_p, t_q of the equations of $L(p), L(q)$ respectively. a subdivides an edge $e \subset L(q)$ and so we mark the two polygons attached to e. The intersection points of $L(p)$ with $2kU$ are added to the binary tree $T(p)$.

Each marked polygon Q is split into two smaller polygons by the line $L(p)$. We add one to the zone index of the polygon on the opposite side of $L(p)$ compared with the origin, whilst the zone index of the polygon on the same side of $L(p)$ as the origin remains the same, see Figure 2.4. Every edge that was previously associated to the polygon Q is appropriately reassigned one of the smaller polygons.

After completing Stage 3 for all points in the extended motif M_{2k} (bar the origin), we end up with a splitting of the polygon $2kU$ into smaller polygons where each polygon has been assigned a zone index. The union of all polygons of zone index k is equal to the closure of the k-th Voronoi zone $Z_k(A; 0)$.

2.2.3 Extension to Dimension Three

We now discuss our implemented extension of the algorithm in Subsection 2.2.2 to $\mathbb{R}^3$. Stages 1 and 2 are identical to the two-dimensional case. In Stage 3, bisectors of points in $\mathbb{R}^3$ are now two-dimensional planes that intersect with other bisectors in one-dimensional lines. For any bisecting plane, we choose a normal vector oriented away from the origin. For any two points p, p' in the balanced binary tree, the line l of intersection (if there is one) of their corresponding bisecting planes can be indexed by p, p' (or indices of these points). l has direction equal to the vector product of the normal vectors of the bisecting planes. Any intersection point of three bisecting planes belongs to three lines of intersection (one for each distinct pair of the three planes). Since such intersection points are naturally ordered along each line l of intersection, we again keep them in balanced binary trees $T(l(p, p'))$.

For every point p in the loop of Stage 3, its corresponding bisecting plane is intersected with the bisecting planes of every previous point q, obtaining the line of intersection $l(p, q)$. If $l(p, q)$ intersects $l(q, q')$, where q' is another previous point in the loop, we insert the intersection point a into the binary trees $T(l(p, q))$, $T(l(p, q'))$ and $T(l(q, q'))$.

For a line of intersection l and for each oriented edge $e \subset l$ between successive intersection points, we maintain a cyclic order of all polyhedra attached to e, which is again kept as an extra attribute of the lower vertex of e in the binary tree $T(l)$. If a new intersection point enters $T(l)$ within the edge $e \subset l$, then all of the polyhedra attached to e are marked.

We split the marked polyhedra by using the intersection function in the CGAL library. Each of the new smaller polyhedra will belong to a list of cyclically ordered polyhedra attached to edges within the intersection lines $l(p, q)$, $l(p, q')$ and $l(q, q')$.

Again, the output of the algorithm will be a splitting of the parallelepiped $2kU$ into smaller polyhedra where each polyhedron has been assigned a zone index. The union of all polyhedra of zone index k is equal to the closure of the k-th Voronoi zone $Z_k(A; 0)$.

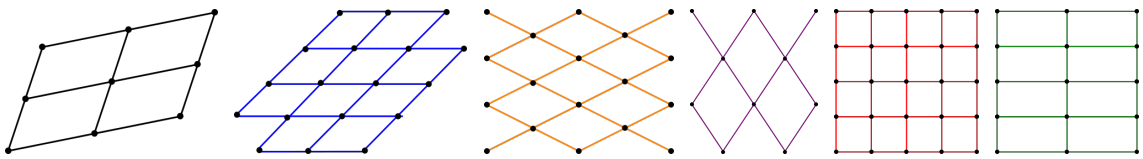

Figure 2.5: The six 2D lattices used in the experiments of Section 2.3. From left to right: a (black) generic lattice with basis $\{(1.25, 0.25), (0.25, 0.75)\}$; a (blue) hexagonal lattice with basis $\{(1, 0), (0.5, \sqrt{3}/2)\}$; an (orange) rhombic lattice with basis $\{(1, 0.5), (1, -0.5)\}$; a (purple) rhombic lattice with basis $\{(1, 1.5), (1, -1.5)\}$; a (red) square lattice with basis $\{(1, 0), (0, 1)\}$; a (green) rectangular lattice with basis $\{(2, 0), (0, 1)\}$.

2.2.4 The Complexity of the Voronoi Zones Algorithm

Theorem 2.18 says that the first k Voronoi zones can be computed in polynomial time in the number m of motif points and equally the order k. The polynomial dependence on m and k seems inevitable, because in general position $m(2k)^n$ bisectors between a fixed centre p and its neighbours in a $2k$-extended unit cell can intersect each other.

Theorem 2.18 (Algorithm Complexity)**.** *Let a periodic set $A \subset \mathbb{R}^n$, $n \leq 3$, have a motif of m points in a Minkowski-reduced basis. Then the time complexity to compute the first k Voronoi zones $Z_i(A; 0)$, $1 \leq i \leq k$, is $\mathcal{O}(m^n(2k)^{n^2}(n \log(m) + n^2 \log(2k)))$.*

Proof. Starting from a reduced basis in Stage 1, the $2k$-extended unit cell consists of $m(2k)^n$ points. Since sorting N points takes $\mathcal{O}(N \log(N))$ time, sorting the $m(2k)^n$ points according to their distance from the origin in Stage 2 takes $\mathcal{O}(m(2k)^n(\log(m) + n \log(2k)))$ time. Stage 3 computes all n-fold intersections of $m(2k)^n$ bisectors, of which there are $\mathcal{O}((m(2k)^n)^n)$, and so takes $\mathcal{O}((m(2k)^n)^n)$ time. Inserting the $\mathcal{O}(m^n(2k)^{n^2})$ intersection points into binary trees and marking polyhedra requires $\mathcal{O}(N \log(N))$ time for N points, and so takes $\mathcal{O}(m^n(2k)^{n^2}(n \log(m) + n^2 \log(2k)))$ time. Splitting the $\mathcal{O}(m^n(2k)^{n^2})$ polytopes depends linearly on the number of intersection points. Therefore, the algorithm's time complexity is $\mathcal{O}(m^n(2k)^{n^2}(n \log(m) + n^2 \log(2k)))$, since this is the greatest time complexity for a single stage of the algorithm. □

The complexity to compute a Minkowski-reduced basis is quadratic in logarithms of the lengths of initial basis vectors for dimensions $n \leq 3$ (exact bounds can be found in [44, Theorems 4.2.1 and 5.0.4]). Although the dependence of the time estimate on the dimension n is exponential, the experiments of Section 2.3 show that the algorithm is very fast in practice.

2.3 Experimental Analysis of the Voronoi Zones Algorithm

The complexity bound in Theorem 2.18 has been experimentally illustrated in dimensions two and three as follows. In dimension $n = 2$, we chose six different lattices which are described and visualised in Figure 2.5. Given one of these lattices Λ and a fixed integer $m \in [1, 50]$, we randomly generated a motif M of m points to get a periodic set $A = M + \Lambda$. Repeating this random generation 100 times for each of the six lattices, we obtain 600 periodic sets for every $m \in [1, 50]$ (see Figure 2.6 for a selection of periodic sets with $m = 1, 2$ along with their Voronoi zones).

In the graphs of Figures 2.7-2.13, each cross represents the mean result, such as runtime in milliseconds, over the 600 periodic sets of each number m of motif points considered. All experiments were performed on a MacBook Pro with 2.3 GHz and 8GB RAM.

Figure 2.7 indicates that starting from about $m = 10$, the runtime increases almost linearly with respect to the number m of motif points as expected by Theorem 2.18. Meanwhile, Figure 2.8 indicates that in dimension two, the runtime follows a slow quadratic increase with respect to the order k of Voronoi zones.

Although higher Voronoi zones are more complicated, Figures 2.9-2.10 show that the number of vertices and polygons increase roughly linearly in dimension two as the order k increases. Hence, their total numbers up to an order k grows quadratically as k increases. Indeed, a linear number of bisectors are expected to produce a quadratic number of intersections in $\mathbb{R}^2$.

Figure 2.11 shows that the number of polygons (similarly for vertices and edges) present in a Voronoi zone depends on the number m of motif points in a different way. For a fixed k the figure shows that the number of polygons plateaus as m increases (stabilising at about 168 for $k = 8$ in Figure 2.11). The dependence of this stabilising number on k can be investigated in future research.

We also computed the perimeter (the total length of the boundary) of k-th Voronoi zones. For a fixed k, Figure 2.12 shows that the perimeter naturally decreases as the number m of motif points increases, because the polygons that comprise each zone tend to become 'rounder'. If we fix the number m of motif points and increase k, the total perimeter length seems to grow logarithmically in Figure 2.13.

The experiments in three dimensions were for periodic sets with m motif points randomly generated within the cubic lattice. Figures 2.14-2.15 illustrate the time analysis of Theorem 2.18 for dimension three. The number of polyhedra grows quadratically in k (Figure 2.16) and stabilises in m (Figure 2.17) which is similar to dimension two. Three-dimensional Voronoi zones for the cubic lattice can be seen in Figure 2.18, whilst Figure 2.19 shows the 5-th Voronoi zones for the FCC (face-centred cubic) and BCC (body-centred cubic) lattices as well as for HCP (hexagonal close packing).

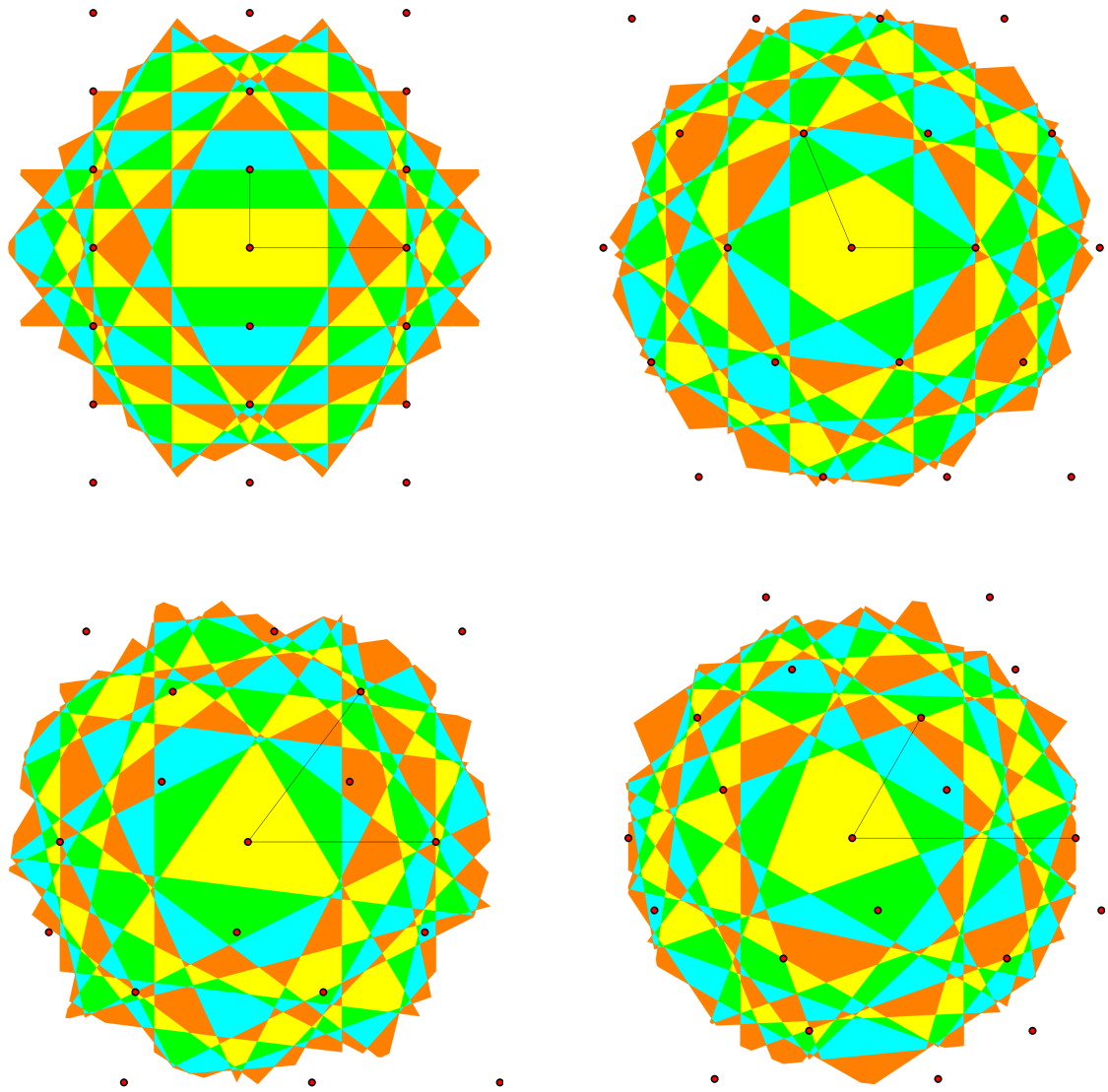

Figure 2.6: The first 12 Voronoi zones of $0 \in \mathbb{R}^2$ for: **Top left:** the rectangular lattice; **Top right:** the lattice with basis $\{(1, 1.5), (1, -1.5)\}$; **Bottom left:** a periodic set with a two-point motif and basis $\{(1, 0.5), (1, -0.5)\}$; **Bottom right:** a periodic set with a two-point motif and basis $\{(1.25, 0.25), (0.25, 0.75)\}$. In each image, the basis vectors are shown by thin black lines, and the image is rotated so that the first basis vector is horizontal.

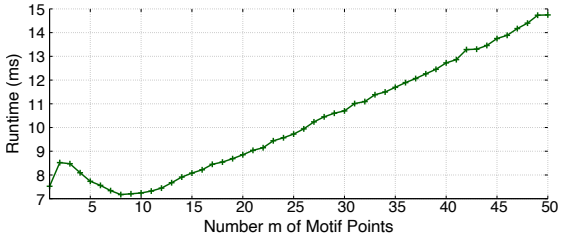

Figure 2.7: The runtime to compute the first $k = 8$ Voronoi zones as the number of motif points takes values $m = 1, \ldots, 50$, averaged over 600 2D periodic sets.

Figure 2.8: The runtime to compute the first k Voronoi zones for $k = 1, \ldots, 30$, averaged over 3000 2D periodic sets with values of m between 1 and 5.

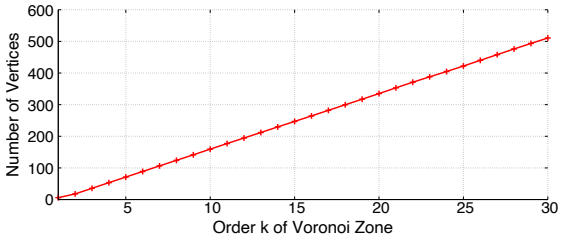

Figure 2.9: The number of vertices in the k-th Voronoi zone for $k = 1, \ldots, 30$, averaged over 6000 2D periodic sets with values of m between 1 and 10.

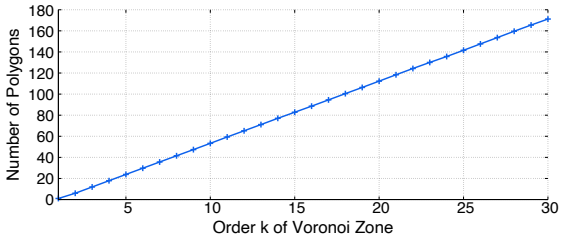

Figure 2.10: The number of polygons that form the k-th Voronoi zone for $k = 1, \ldots, 30$, averaged over 6000 2D periodic sets with values of m between 1 and 10.

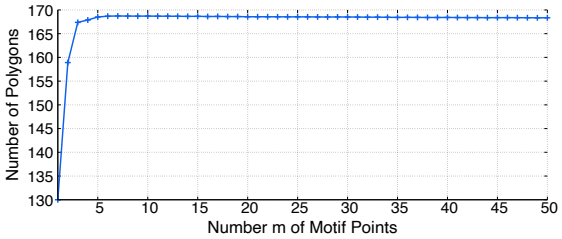

Figure 2.11: The number of polygons that form the 8-th Voronoi zone as the number of motif points takes values $m = 1, \ldots, 50$, averaged over 600 2D periodic sets.

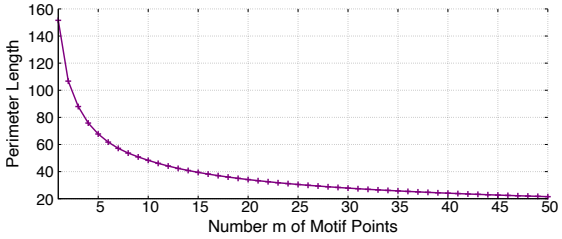

Figure 2.12: The total perimeter length of the 8-th Voronoi zone as the number of motif points takes values $m = 1, \ldots, 50$, averaged over 600 2D periodic sets.

Figure 2.13: The total perimeter length of k-th Voronoi zones for $k = 1, \ldots, 30$, averaged over 6000 2D periodic sets with values of m between 1 and 10.

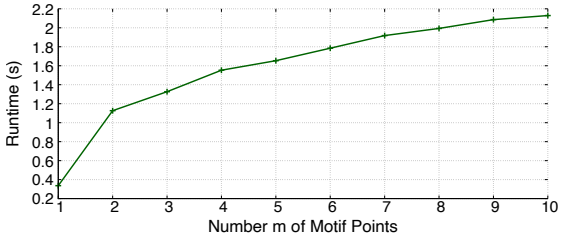

Figure 2.14: The runtime to compute the first $k = 5$ Voronoi zones as the number of motif points takes values $m = 1, \ldots, 10$, averaged over 10 3D periodic sets.

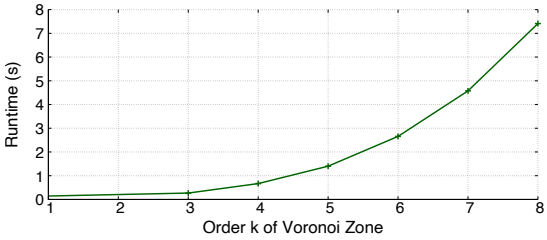

Figure 2.15: The runtime to compute the first k Voronoi zones for $k = 1, \ldots, 8$, averaged over 50 3D periodic sets with values of m between 1 and 5.

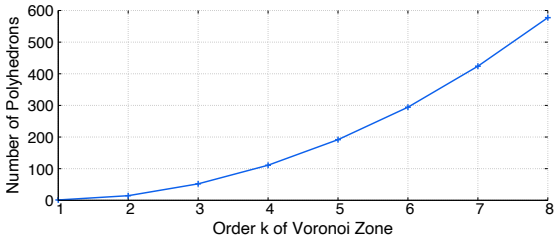

Figure 2.16: The number of polyhedra that form the k-th Voronoi zone for $k = 1, \ldots, 10$, averaged over 50 3D periodic sets with values of m between 1 and 5.

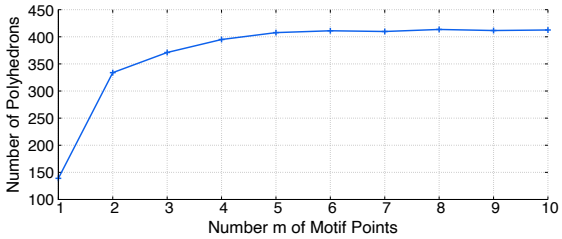

Figure 2.17: The number of polyhedra that form the 5-th Voronoi zone as the number of motif points takes values $m = 1, \ldots, 10$, averaged over 10 3D periodic sets.

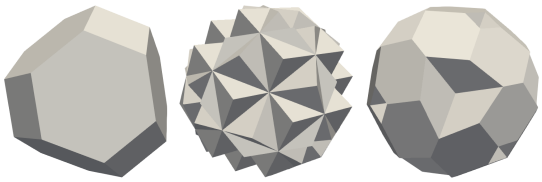

Figure 2.18: Voronoi zones $Z_k(\Lambda; 0)$ for $k = 4, 5, 6$ respectively in the cubic lattice Λ.

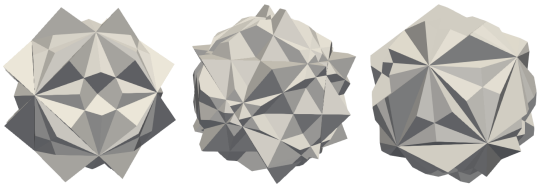

Figure 2.19: 5-th Voronoi zones for FCC, BCC and HCP respectively.

2.4 Conclusion and Open Problems

Computing and visualising Voronoi zones of periodic sets beyond classical Voronoi domains is important for understanding crystals whose distant interatomic interactions determine key physical properties.

The main novelty of this chapter is an algorithm to compute the first k Voronoi zones of a periodic set for any k in dimensions two and three, which has polynomial time in the key parameters k and m for a fixed dimension n.

The new generalisations extending work relating to lattices required overcoming several roadblocks. Most importantly, an explicit estimate for an extension of a suitably reduced cell to guarantee the covering of the k-th Voronoi zone in Lemma 2.17 is provided.

We finish with some open problems relating to Voronoi zones of periodic sets:

- Can the upper bound of the $2k$-extension in Lemma 2.17 be improved? An improved bound will enable a better asymptotic complexity in Theorem 2.18, and may be possible considering that the square lattice (Figure 2.20) only requires the 2-extended unit cell to cover the third Voronoi zone.

- Higher Voronoi zones reveal symmetries of a periodic set, for example the six-fold rotational symmetry of the hexagonal lattice can clearly be seen in Figure 2.21. One can try to use other geometric properties of higher Voronoi zones to enable a finer classification of crystals than by classical crystallographic groups.

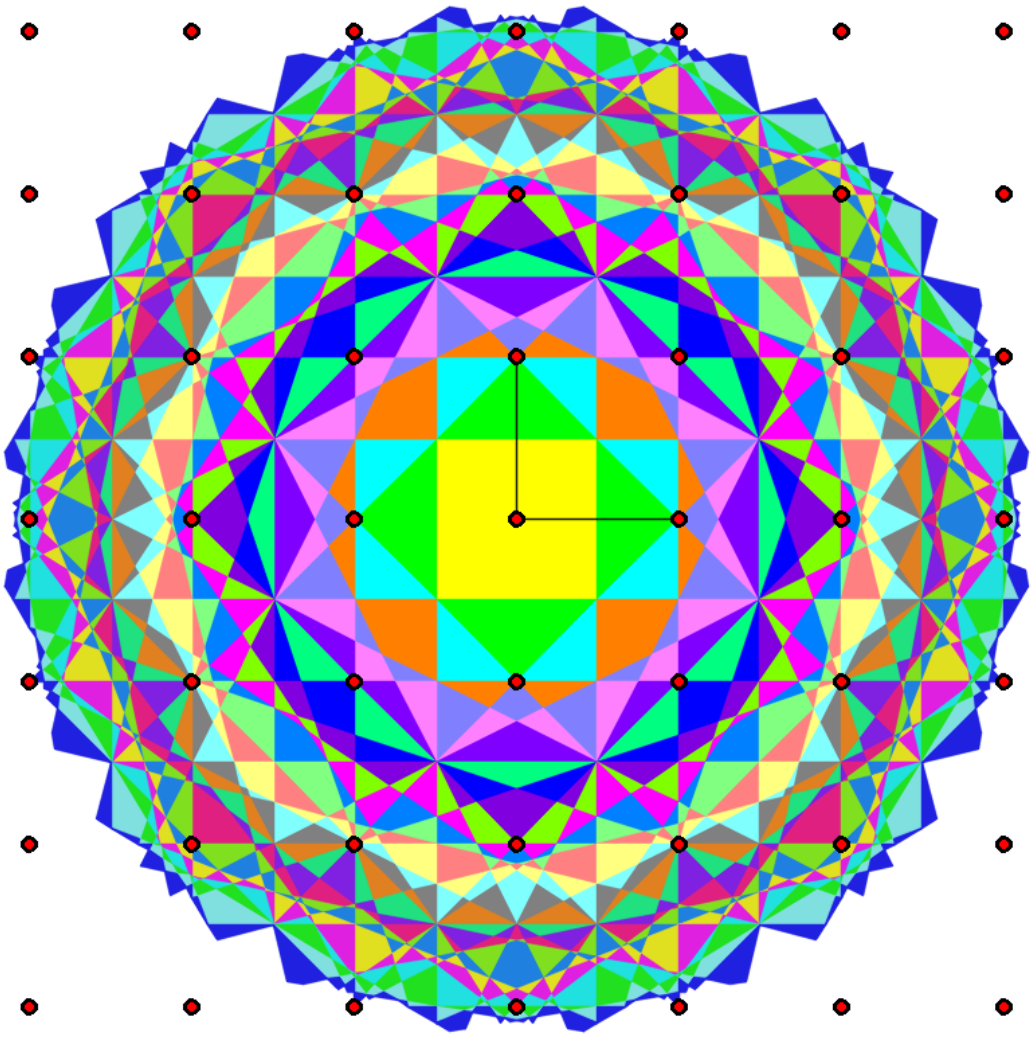

Figure 2.20: The first 30 Voronoi zones of the origin in the square lattice (red dots).

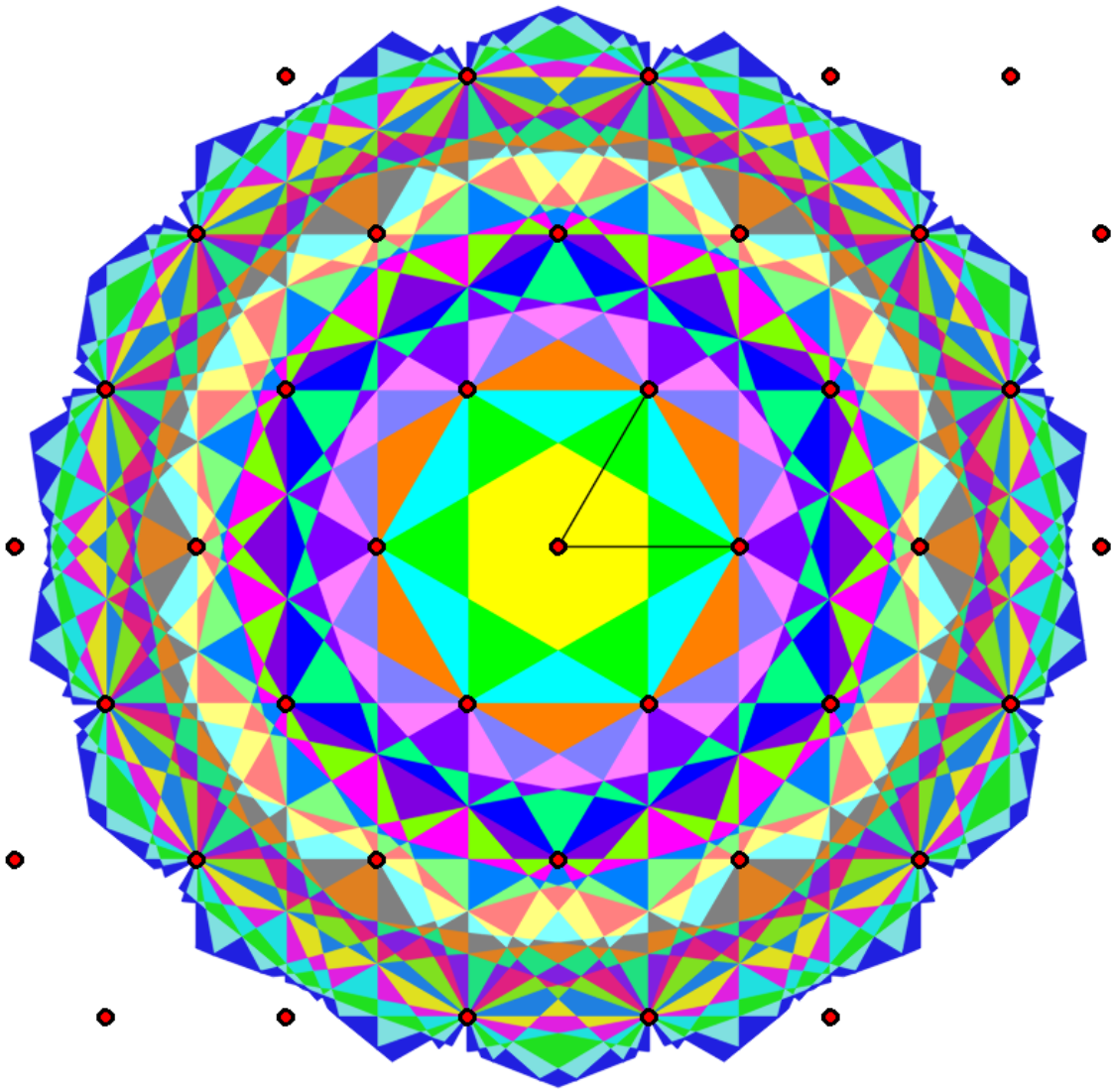

Figure 2.21: The first 30 Voronoi zones of the origin in the hexagonal lattice (red dots).

Chapter 3

The Density Fingerprint

(This chapter is based on the paper "The density fingerprint of a periodic point set" authored by H. Edelsbrunner, T. Heiss, V. Kurlin, P.S. and M. Wintraecken, and accepted for publication in the proceedings of the 37th Symposium on Computational Geometry, 2021 [21].)

In recent years there has been a substantial change in the process by which new functional crystal structures are discovered. The exponential increase in computing power has made the ambitious idea of being able to identify useful crystal structures without setting foot inside a laboratory a reality. In particular, the field of Crystal Structure Prediction (CSP) seeks to predict properties of a crystal simply from its composition and geometric structure. In novel work outlined in [49], given a molecule of fixed composition, material scientists generate a large dataset of simulated crystal structures, with each entry representing a local minimum of a complicated energy function. By computing the properties of each entry, a small subset can be identified as promising candidates to synthesise in the laboratory. Hence useful, functional materials can be discovered that perhaps previously would have required the impossible task of synthesising thousands or even millions of crystals. This process could revolutionise functional materials discovery, yet in key ways it can still be streamlined.

These large datasets often contain many near-duplicate entries that slow down the process since costly computational calculations are unnecessarily performed on these doppelgängers. Hence removing excess entries in the datasets could significantly speed up the process. Moreover, the algorithm used to generate these datasets starts with a (broadly-speaking) random arrangement of the specified molecule, and then modifies this arrangement until the structure lies within a threshold of a perceived local minimum of the energy function, in a similar approach to gradient descent. This is not too dissimilar from the old-fashioned approach of literally shaking models of molecules in a box to see what arrangement it settles in! What if we can more elegantly guide the search for new functional materials in crystal space by efficiently identifying deep stable local minima of the energy

function that correspond to the most stable and typically most useful structures? When trying to find the source of a river, a sensible approach once you've encountered the river is simply to head upstream. Similarly, can we better describe the space of crystals so optimal structures are found more quickly? We believe both these potential improvements can be obtained with the implementation of a sufficiently descriptive classification of crystal structures, and it is in this chapter that we describe the progress we have made towards this goal.

3.0.1 Contributions and Chapter Outline

The main contributions of this chapter are as follows:

- We formulate the classification problem of periodic sets in Section 3.1, and then introduce the density fingerprint in Definition 3.5, consisting of an infinite sequence of density functions (Definition 3.2), as a classification.
- We prove that, in $\mathbb{R}^3$, the density fingerprint is Lipschitz continuous in Theorem 3.10 and complete for an open and dense set of periodic sets in Theorem 3.14.
- We describe how the density fingerprint can be computed in Theorem 3.21, linking the density fingerprint to the Voronoi zones of Chapter 2. An implementation in dimensions one, two and three is available in C++ [53].

Chapter outline: Section 3.1 formulates the problem and describes the drawbacks of previous approaches. The density fingerprint is introduced in Section 3.2, for which we go on to discuss continuity and completeness in Sections 3.3 and 3.4 respectively. Computing the density fingerprint is tackled in Section 3.5, whilst an application to CSP is described in Section 3.6. We conclude with a discussion in Section 3.7.

3.1 A Continuous, Complete, Isometry Classification

As described above, we desire a classification of crystals that will improve the novel and groundbreaking approach of CSP. The first step is to model crystal structures as a rigorously defined mathematical object, which we achieve by representing crystals as periodic sets (Definition 1.5). We therefore desire a classification of periodic sets, and hence crystal structures, that can firstly determine whether two sets are identical or not, and secondly, if the two sets do not match, quantify their similarity. Hence, with this in mind, we are looking for a solution to the following problem.

Problem 1 (A Continuous, Complete, Isometry Classification of Periodic Sets)**.** We desire a classification of periodic sets that satisfies the following three conditions:

I *Invariance under isometries:* any two periodic sets that are isometric (Definition 1.10) should be classified identically.

II *Continuous under perturbations:* if a periodic set undergoes a small perturbation, the distance between the old and new classifications should be small.

III *Completeness:* no two non-isometric periodic sets should have the same classification.

Condition I is required since crystals are rigid bodies and so are invariant under rigid motions. Therefore, periodic sets that are related by a rigid motion should be deemed equivalent. We use the slightly larger group of isometries for simplicity. Condition I combined with Condition III implies that two structures will be identified as identical if and only if they are isometric to each other, whilst Condition II will enable us to rigorously quantify how similar two non-isometric structures are to each other.

3.1.1 Drawbacks of Previous Approaches

Ultimately, the question being considered here is deceptively simple: given two periodic sets in $\mathbb{R}^3$, how close are they to being isometric to each other? Yet previous approaches and existing tools available to tackle this problem are not without their drawbacks.

Ambiguity of Unit Cells

Typically, crystals, which are conceptually infinite, tend to be represented by a finite region or building block which determines the entire infinite crystal. These building blocks are called unit cells (Definition 1.2). Reducing an object of infinite size to that of something that is finite seems a very sensible method of simplifying the problem. However, there is much ambiguity regarding unit cells.

In Definition 1.2, a unit cell is defined to be the set of points that can be expressed as a linearly combination of the basis vectors of the lattice with coefficients in the interval $[0, 1)$. Yet, in dimensions $n \geq 2$, all lattices have infinitely many bases, and thus infinitely many unit cells, see Figure 3.1. Because of this, there have been several attempts to define algorithms that output a unique basis (and therefore a unique unit cell) for a given lattice. Such a basis is called a *reduced basis* (and yield a corresponding reduced unit cell), and tends to be composed of vectors that are reasonably short and almost orthogonal. Finding good reduced bases has been widely studied, especially given its relevance to other fields outside of crystallography such as cryptology. Some of the most well known reduced bases include those by Hermite, Minkowski (Definition 1.3), Lagrange and Niggli [44].

The issue with all reduced bases is that they are not stable under perturbations of the periodic set [4]. As an example, the Niggli reduced cell [45], for certain configurations, requires a choice of basis vectors. However, if the periodic set is only slightly perturbed, a significantly different set of basis vectors is outputted. This lack of stability poses significant problems when establishing a continuous classification of crystal structures. Any invariants

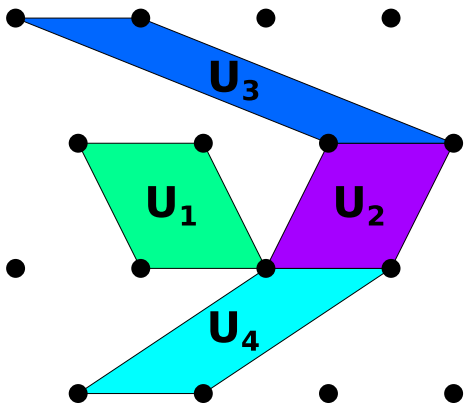

Figure 3.1: All four unit cells correspond to a set of basis vectors that generate the hexagonal lattice.

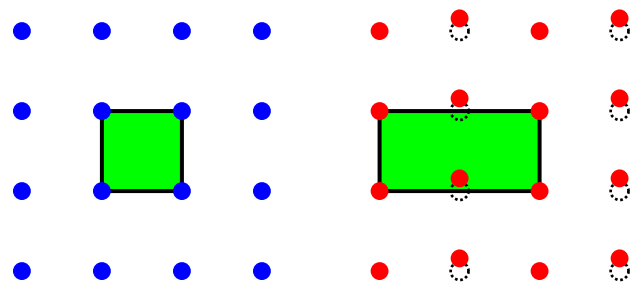

Figure 3.2: On the left, we have the square lattice where the reduced unit cell is highlighted in green. If every other lattice point in the x-direction is perturbed slightly, as on the right, the reduced unit cell doubles in area, and the number of motif points in the unit cell also doubles from one to two.

that are based on the unit cell, such as the parameters of the parallelepiped or fractional coordinates of atoms within the unit cell, cannot be stable under perturbations of the periodic set. As another example, the unit square lattice generated by the vectors $(1, 0)$ and $(0, 1)$ has a square unit cell. However, if every point in the lattice with odd x-coordinate is only sightly perturbed, the reduced basis doubles in volume, see Figure 3.2. These discontinuities under small perturbations makes it impossible to use compact unit cells as the foundation of a continuous classification of crystals.

Current Tools Require Tolerances

There are various software systems currently used in practice to quantify the similarity between crystals, which include Compack [13], Mercury [40] and Compstru [25]. These tools can be of great help when comparing crystals. However, they employ heuristics like cut-offs and tolerances. For example, by default, Mercury seeks to match up to 15 molecules, where two molecules are said to be matched if they lie within some distance and angle tolerances that are set by the user. Then, considering just the matched molecules, the root mean square deviation is computed and this output is used to assess how similar the structures are. In some scenarios, this does give an accurate description of the similarity, whilst in others it can be misleading. Moreover, changing the tolerances can lead to a different value of the root mean square deviation, and of course if you let the number of molecules compared go to infinity, it is likely that the root mean square deviation will also tend to infinity.

Space Groups are a Discrete Classification

It is common practice for crystallographers to focus on the space group of a structure, and with good reason. This contains important information particularly relating to the structure's symmetry. There are 230 space groups in three dimensions, obtained from the Bravais lattices by including rotations, screw axes, mirror and glide planes, plus points of inversion. These space groups can be thought of as a stratification of the space of isometry classes of crystals, yet the stratum a structure belongs to is not a continuous property. For example, a cubic lattice has 3 axes of 4-fold rotation. Yet if we extend one of the basis vectors of the cubic lattice, even by the smallest amount, we obtain a tetragonal lattice with only 1 axis of 4-fold rotation.

3.2 Density Functions and the Density Fingerprint

In light of the review of previous methods and tools to geometrically compare crystal structures in Subsection 3.1.1, we must desire geometric invariants of crystals that are continuous and independent of the unit cell in order to solve Problem 1. Motivated by the single value density of a crystal, which is defined to be the molecular weight of the atoms within a unit cell divided by the unit cell volume, we introduce the density fingerprint that somewhat extends the concept of density to an infinite family of continuous functions, which we call density functions.

Notation 3.1 ($B(C;r)$)**.** Let C be a set of points in $\mathbb{R}^n$. The set of balls with centres at the points of C and radii t is denoted by $B(C;t)$.

Definition 3.2 (k-th Density Function $\psi_k^A(t)$)**.** Let $A \subset \mathbb{R}^n$ be a periodic set with unit cell U. The *k-th density function*, $\psi_k^A(t)$, is defined to be the fractional volume of the unit cell U that is covered by exactly k balls of $B(A;t)$,

$$\psi_k^A(t) = \frac{\mathrm{Vol}[\{p \text{ in exactly } k \text{ balls of } B(A;t) \mid p \in U\}]}{\mathrm{Vol}[U]}.$$

We illustrate how the density functions grow and diminish in a simple case in Example 3.3, where the periodic set considered is the unit square lattice generated by the basis vectors $\vec{v_1} = (1,0)$ and $\vec{v_2} = (0,1)$.

Example 3.3. Let the periodic set A be the unit square lattice. We describe here some key milestones of the first nine density functions of A as the radii of balls of $B(A;t)$ increases, as seen in the top set of images in Figure 3.3. In the top-left of Figure 3.3, we see snapshots of the growing balls around the lattice points at radii $t = 0.25, 0.55, 0.75, 1$. In particular, we note that green double intersections (which will be born at radius 0.5) appear in the second image ($t = 0.55$), orange and red triple and quadruple intersections (which will be

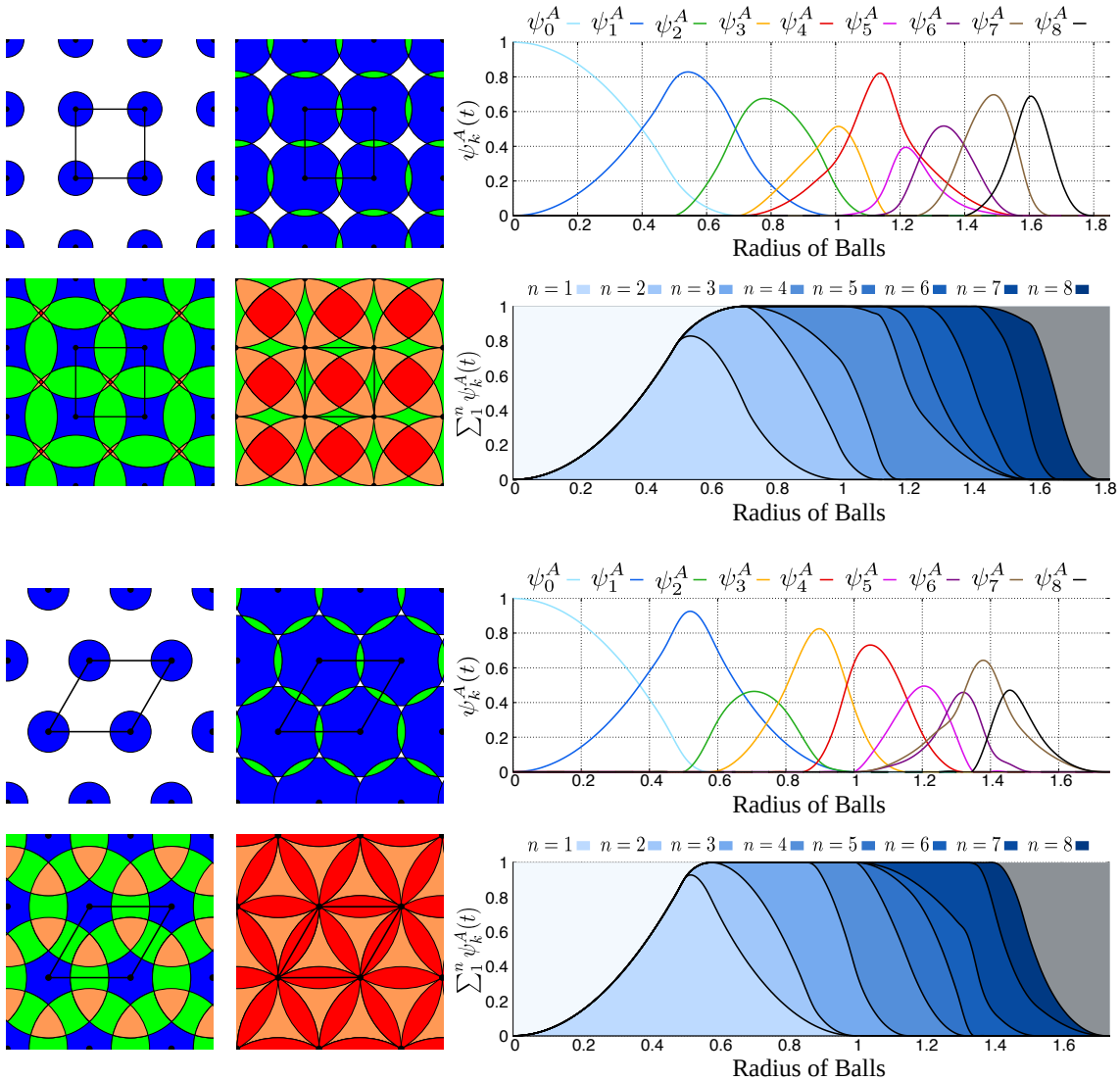

Figure 3.3: The density functions of the square lattice (top set of images) and, for comparison, the hexagonal lattice (bottom set of images). **Left:** the k-fold covers of the lattice for four different radii: $t = 0.25, 0.55, 0.75, 1.00$. **Right:** the graph of the first nine density functions above the corresponding densigram, in which the zeroth density function can be seen upside-down and the remaining density functions are accumulated from left to right.

born at radius $\sqrt{2}/2$) appear in the third image ($t = 0.75$), and the blue regions covered by exactly one ball have vanished in the final image ($t = 1$).

The plot of the first nine density functions shown to the right of these snapshots shows how the values of the functions grow and decrease as the radius increases. In particular, we note in Table 3.1 the milestone radii at which each function becomes nonzero (is born) and vanishes (dies).

k	0	1	2	3	4	5	6	7	8
Birth radius	0	0	0.5	$\frac{\sqrt{2}}{2}$	$\frac{\sqrt{2}}{2}$	1	$\frac{5\sqrt{2}}{6}$	1.25	$\sqrt{2}$
Death radius	$\frac{\sqrt{2}}{2}$	1	$\frac{\sqrt{5}}{2}$	$\frac{5\sqrt{2}}{6}$	$\frac{\sqrt{10}}{2}$	$\frac{\sqrt{10}}{2}$	$\frac{\sqrt{10}}{2}$	$\frac{5}{3}$	$\frac{\sqrt{13}}{2}$

Table 3.1: The birth and death radii of the first nine density functions of the square lattice.

We have stressed that we are looking for geometric invariants of periodic sets that are invariant under isometries and independent of the unit cell. Density functions fulfil both of these criteria, despite there being a choice of unit cell in Definition 3.2.

Lemma 3.4. *The k-th density function ψ_k^A of a periodic set A is invariant under isometries of A, and independent of the unit cell chosen in Definition 3.2.*

Proof. By definition, isometries preserve distances and volumes. In particular, any point of $\mathbb{R}^n$ that is within a distance t of exactly k points of A will again be within a distance t of exactly k points of A after an isometry has been applied. Therefore, any region that contributes towards the k-th density function will again contribute towards the k-th density function after an isometry.

To show that density functions are independent of the unit cell chosen, consider partitioning $\mathbb{R}^n$ into its orbits generated by the lattice translations. Any unit cell of minimal volume will have one representative of each orbit. As every point in the same orbit will have the same number k of points in A within a distance t, any unit cell of minimal volume will have the same fractional volume of points covered by exactly k balls of $B(A;t)$. Any unit cell of greater volume will have m representatives of each orbit, but then its volume will be m times greater, so the fractional volume of points covered by exactly k balls of $B(A;t)$ remains the same. □

Having introduced the concept of a density function, we wish to associate to a periodic set A its infinite family of density functions, which we call its density fingerprint, $\Psi(A)$.

Definition 3.5 (Density Fingerprint $\Psi(A)$)**.** Let $A \subset \mathbb{R}^n$ be a periodic set. The *density fingerprint* of A is the infinite family of functions $\Psi(A) = \{\psi_0^A, \psi_1^A, \ldots\}$. The function $\Psi : A \to \{\psi_0^A, \psi_1^A, \ldots\}$ is called the *density fingerprint map.*

In Sections 3.3 and 3.4, we prove that, in dimension $n = 3$, the density fingerprint is continuous under perturbations and complete for generic periodic sets, conjecturing that it is complete for all periodic sets in dimensions $n \geq 2$.

3.3 Continuity of the Density Fingerprint Map

In this section, we will prove that, in dimension $n = 3$, the density fingerprint map from periodic sets to density fingerprints is Lipschitz continuous with respect to small perturbations of the periodic set.

Definition 3.6 (Lipschitz Continuity)**.** Given two metric spaces $(X, d_X), (Y, d_Y)$, a function $f\colon X \to Y$ is called *Lipschitz continuous* if there exists a constant $C \in \mathbb{R}_{\geq 0}$ such that for all $x_1, x_2 \in X$, we have that

$$d_Y(f(x_1), f(x_2)) \leq C \cdot d_X(x_1, x_2). \tag{3.1}$$

Any such C is called a *Lipschitz constant*, and the smallest Lipschitz constant is called *the (best) Lipschitz constant.*

To show that the density fingerprint map Ψ is Lipschitz continuous, we must first define the distances we will use between pairs of periodic sets and between pairs of density fingerprints.

Definition 3.7 (Bottleneck Distance $d_B(A, Q)$)**.** Let A, Q be point sets of equal cardinality. The *(Euclidean) bottleneck distance*, $d_B(A, Q)$, is defined to be the infimum over all bijections $\gamma\colon A \to Q$ of the supremum of the (Euclidean) distances between all pairs of points $a \in A$ and $\gamma(a) \in Q$ (see Figure 3.4). Namely,

$$d_B(A, Q) = \inf_{\gamma\colon A \to Q} \sup_{a \in A} \|a - \gamma(a)\|_2 . \tag{3.2}$$

The bottleneck distance is perhaps theoretically the best way of comparing periodic sets. However, it is not practical since it involves finding the best bijection between infinitely many points. This is why we need more appropriate geometric invariants to describe periodic sets with distance functions that are easier to compute. For example, it is easy to (approximately) compute the d_∞ distance between density fingerprints.

Definition 3.8 ($d_\infty(\Psi(A), \Psi(Q))$)**.** Let A, Q be periodic sets with density fingerprints $\Psi(A), \Psi(Q)$ respectively. We define the distance between $\Psi(A)$ and $\Psi(Q)$, $d_\infty(\Psi(A), \Psi(Q))$, to be the supremum over $k \geq 0$ of the weighted infinity norm (see Figure 3.4). That is,

$$d_\infty(\Psi(A), \Psi(Q)) = \sup_{k \geq 0} \frac{1}{\sqrt[3]{k+1}^2} \left\| \psi_k^A - \psi_k^Q \right\|_\infty . \tag{3.3}$$

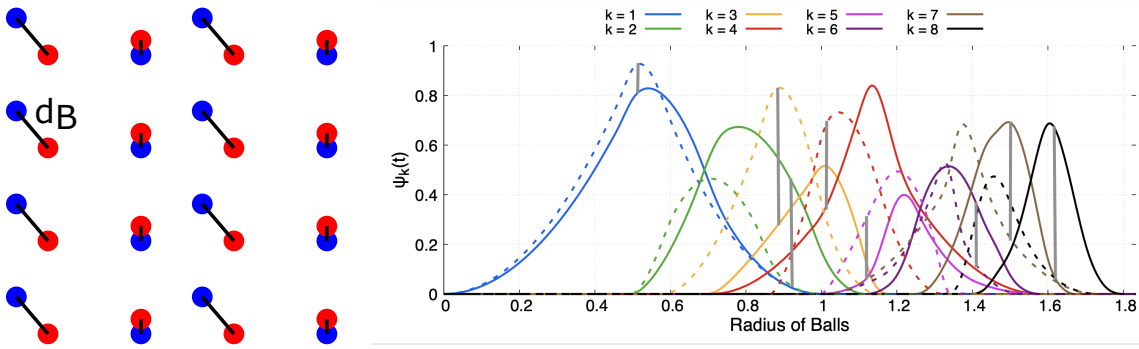

Figure 3.4: **Left:** the bottleneck distance d_B between the red and blue periodic sets. The black lines between pairs of points denotes the optimal bijection. **Right:** the L_∞-distances between the k-th density function, $1 \leq k \leq 8$, of the square (solid lines) and hexagonal (dashed lines) lattices are equal to the lengths of the grey vertical bars.

The reason for the damping coefficient, $\frac{1}{\sqrt[3]{k+1}^2}$, is technical and is drawn out in the proof of Theorem 3.10, but in summary it compensates for the fact that higher density functions will be more sensitive to perturbations than lower ones.

Theorem 3.10 makes use of Lemma 3.9 stating that there will be a common lattice of periodic sets with a small bottleneck distance between them.

Lemma 3.9 (Common Lattice)**.** *Let $A, Q \subset \mathbb{R}^3$ be periodic sets, and let $r_Q > 0$ be the packing radius (Definition 1.11) of Q. If $d_B(A, Q) < r_Q$, then there exists a lattice Λ with unit cell U such that $|A \cap U| = |Q \cap U|$ and $A = (A \cap U) + \Lambda$, $Q = (Q \cap U) + \Lambda$. Namely, A and Q can be expressed in terms of a common lattice, and the cardinality of their motifs for this lattice agree.*

Proof. We will prove this by contradiction, by assuming that there is no common lattice for A and Q. In this case, for a lattice Λ_A of A and a lattice Λ_Q of Q, $\Lambda_A \cap \Lambda_Q$ is a lattice of dimension at most two. As a result, there exists a basis vector $\vec{v}$ of Λ_A such that $n\vec{v} \in \Lambda_Q$ implies $n = 0$. Choosing a point $a \in A$, consider the line of infinitely many evenly spaced points $a(n) = a + n\vec{v}$, $n \in \mathbb{Z}$. For each $a(n)$, let $q(n) \in \Lambda_Q$ be the lattice point such that $a(n) - \overrightarrow{q(n)} \in U_Q$, where U_Q is the unit cell of the lattice Λ_Q. We denote $b(n) = a(n) - \overrightarrow{q(n)}$, where $b(n)$ is simply the translate of $a(n)$ by a lattice vector such that $b(n)$ is inside U_Q.

There must be infinitely many pairwise different points $b(n)$ in U_Q. For if $b(n) = b(m)$, then $a(n) - \overrightarrow{a(m)} = q(n) - \overrightarrow{q(m)}$ which implies that $(n - m)v = q(n) - \overrightarrow{q(m)}$. $q(n) - \overrightarrow{q(m)}$ is a lattice point of Λ_Q, and so must be 0, which implies $(n - m)v = 0$, and so $n = m$.

But these infinitely many points must all be within a distance $\delta = d_B(A, Q)$ from a point in Q. But here we have a contradiction. To see this, let $b(i)$ and $b(j)$ be at a distance less than $\epsilon = r_Q - \delta$ from each other. Two such points must exist since we have infinitely

many in a bounded region. Consider the infinite line of points $b(i) + n(\overrightarrow{b(j) - b(i)})$, $n \in \mathbb{Z}$. We note that $b(i) + n(\overrightarrow{b(j) - b(i)}) = b(i + n(j - i))$ modulo the lattice Λ_Q, as both points can be expressed as $a(i) + n(j - i)\vec{v} + \vec{q}$ for some $q \in \Lambda_Q$. Hence all of the points on the line must be within a distance $\delta = d_B(A, Q)$ from a point in Q. However, the distance between contiguous points on the line is less than ϵ, and since the gap between balls of radius δ centred at points of Q is at least 2ϵ, then at least one of the points on the line is outside all such balls, and we reach our contradiction. □

We can now prove that the density fingerprint is Lipschitz continuous by using the fact that there is necessarily a common lattice between two periodic sets that are small perturbations of each other.

Theorem 3.10 (Fingerprint Continuity)**.** *Let $A, Q \subset \mathbb{R}^3$ be periodic sets, both with packing radius $r > 0$ and covering radius $R < \infty$ (see Definitions 1.11 and 1.12). If $\delta = d_B(A, Q) < r$, then there exists a constant $C = C(r, R)$ such that $d_\infty(\Psi(A), \Psi(Q)) \leq C \cdot d_B(A, Q)$.*

Proof. (This proof was principally constructed by Herbert Edelsbrunner.) By Lemma 3.9, there is a lattice $\Lambda \subset \mathbb{R}^3$ that is common to both A and Q, and we write U for the corresponding unit cell. Let the bijection $\gamma\colon A \to Q$ be such that the supremum of the Euclidean distances between all pairs of points $a \in A$ and $\gamma(a) \in Q$ is equal to $d_B(A, Q)$. Let $k \in \mathbb{Z}_{\geq 0}$ and $t \in \mathbb{R}_{\geq 0}$. We want to find an upper bound for $d_\infty(\Psi(A), \Psi(Q)) = \sup_{k \geq 0} \frac{1}{\sqrt[3]{k+1}^2} \left\| \psi_k^A - \psi_k^Q \right\|_\infty$. Firstly, let us fix k and t and find an upper bound for $\left| \psi_k^A(t) - \psi_k^Q(t) \right|$. By letting $\cup^k B(A; t)$ be the union of all points in $\mathbb{R}^3$ that are covered by at least k balls of $B(A; t)$, and letting $A_t^k = \cup^k B(A; t) \setminus \cup^{k+1} B(A; t)$ be the union of all points in $\mathbb{R}^3$ that are covered by exactly k balls of $B(A; t)$, we have

$$\left| \psi_k^A(t) - \psi_k^Q(t) \right| = \frac{\left| \mathrm{Vol}\left[A_t^k \cap U\right] - \mathrm{Vol}\left[Q_t^k \cap U\right] \right|}{\mathrm{Vol}[U]}. \tag{3.4}$$

As a first step, let's find an upper bound for the numerator, Δ, on the right-hand side of Equation 3.4, in the case where the motif M_Q of Q differs from the motif M_A of A due to a perturbation by at most δ of a single point $a \in M_A$ to a point $q \in M_Q$. Namely, $Q = (A \setminus (a + \Lambda)) \cup (q + \Lambda)$. Let the bijection γ between A and Q be the identity except for the point $a \in M_A$ which it maps to $q = \gamma(a) \in B(a; \delta)$. A point $x \in \mathbb{R}^3$ is possibly covered by a different number of balls before and after this perturbation only if $x \in (B(a; t) \ominus B(q; t)) + \Lambda$, where $\ominus$ denotes the symmetric difference. Observing that this set is contained in $\left(B\left(\frac{a+q}{2}; t + \frac{\delta}{2}\right) \setminus B\left(\frac{a+q}{2}; t - \frac{\delta}{2}\right)\right) + \Lambda$ (see Figure 3.5), we have that

$$\Delta \leq \mathrm{Vol}[B(a; t) \ominus B(q; t)] \leq \frac{4\pi}{3}\left(\left(t + \frac{\delta}{2}\right)^3 - \left(t - \frac{\delta}{2}\right)^3\right) = \frac{4\pi}{3}\left(3t^2\delta + \frac{1}{4}\delta^3\right). \tag{3.5}$$

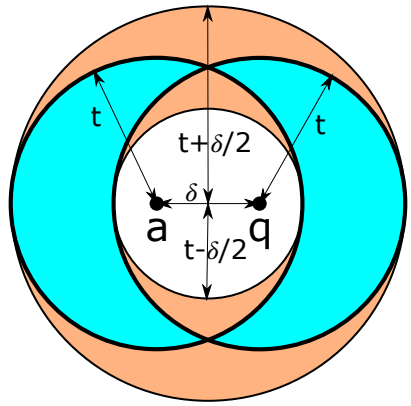

Figure 3.5: For points a and q a distance δ apart, the symmetric difference $B(a;t) \ominus B(q;t)$ is contained within $B\left(\frac{a+q}{2}; t+\frac{\delta}{2}\right) \setminus B\left(\frac{a+q}{2}; t-\frac{\delta}{2}\right)$.

Perturbing one point of M_A after another, we can bound the error each time by Equation 3.5. Setting $\rho = |M_A|/\operatorname{Vol}[U]$, we get

$$\left|\psi_k^A(t) - \psi_k^Q(t)\right| \leq \frac{4\pi}{3}\left(3t^2\delta + \frac{1}{4}\delta^3\right) \cdot \rho. \tag{3.6}$$

We now wish to eliminate the dependence on t, which can be done by noting that for each k there exists a value of t beyond which the k-th density functions of A and Q vanish. To determine an upper bound for this value, consider a point $y \in \mathbb{R}^3$. By the definition of the covering radius R, for $t \geq R$, $B(A;R)$ and $B(Q;R)$ cover $B(y;t-R)$. In fact, we need only balls of radius R centred at the points of $A \cap B(y;t)$ and similarly $Q \cap B(y;t)$ to cover $B(y;t-R)$. By using a volume argument, it follows that the number of points in the two sets $A \cap B(y;t)$ and $Q \cap B(y;t)$ is at least $\frac{\frac{4\pi}{3}(t-R)^3}{\frac{4\pi}{3}R^3} = \left(\frac{t}{R}-1\right)^3$. By setting $k+1 \leq \left(\frac{t}{R}-1\right)^3$, we see that for $t \geq R\sqrt[3]{k+1}+R$, both sets have at least $k+1$ points. Therefore, at radius t, y is covered by at least $k+1$ balls. Since this holds for any $y \in \mathbb{R}^3$, we have $\psi_k^A(t) = \psi_k^Q(t) = 0$ for $t \geq R\sqrt[3]{k+1}+R$. This also holds for the simpler inequality $t \geq 2R\sqrt[3]{k+1}$ since $\sqrt[3]{k+1} \geq 1$ for all $k \geq 0$. Replacing t in Equation 3.6 by this latter bound, we obtain

$$\begin{aligned}\left|\psi_k^A(t) - \psi_k^Q(t)\right| &\leq \frac{4\pi}{3}\left(3\left(2R\sqrt[3]{k+1}\right)^2\delta + \frac{1}{4}\delta^3\right) \cdot \rho \\ &= \frac{4\pi}{3}\left(12R^2\sqrt[3]{k+1}^2\delta + \frac{1}{4}\delta^3\right) \cdot \rho.\end{aligned} \tag{3.7}$$

Since this inequality is true for any t, then by dividing through by $\sqrt[3]{k+1}^2$, we have

$$\begin{aligned}\frac{1}{\sqrt[3]{k+1}^2}\left\|\psi_k^A-\psi_k^Q\right\|_\infty &\leq 16\pi R^2\rho\delta+\frac{\pi\rho}{3\sqrt[3]{k+1}^2}\delta^3 \leq 16\pi R^2\rho\delta+\frac{\pi\rho}{3}\delta^3 \\ &\leq \frac{12R^2}{r^3}\delta+\frac{1}{4r^3}\delta^3,\end{aligned} \tag{3.8}$$

where the last equality results from noting that due to the definition of the packing radius r, we have that $\rho\cdot\frac{4\pi}{3}r^3\leq 1$ which implies that $\rho\leq\frac{3}{4\pi r^3}$. By the constraint in the theorem, we have that $\delta^2<r^2<R^2$, and so finally we have

$$\frac{1}{\sqrt[3]{k+1}^2}\left\|\psi_k^A-\psi_k^Q\right\|_\infty \leq \frac{12R^2}{r^3}\delta+\frac{1}{4r^3}R^2\delta \leq \frac{12R^2}{r^3}\delta+\frac{R^2}{r^3}\delta=\frac{13R^2}{r^3}\delta. \tag{3.9}$$

Hence, $d_\infty(\Psi(A),\Psi(Q))\leq\frac{13R^2}{r^3}\delta$, and so Ψ is Lipschitz continuous where we can take $C=\frac{13R^2}{r^3}$ as an upper bound for the Lipschitz constant. □

Figure 3.6 illustrates Theorem 3.10 in $\mathbb{R}^2$ for a periodic set A and its perturbation Q by overlaying the first eight (undamped) density functions of the two periodic sets.

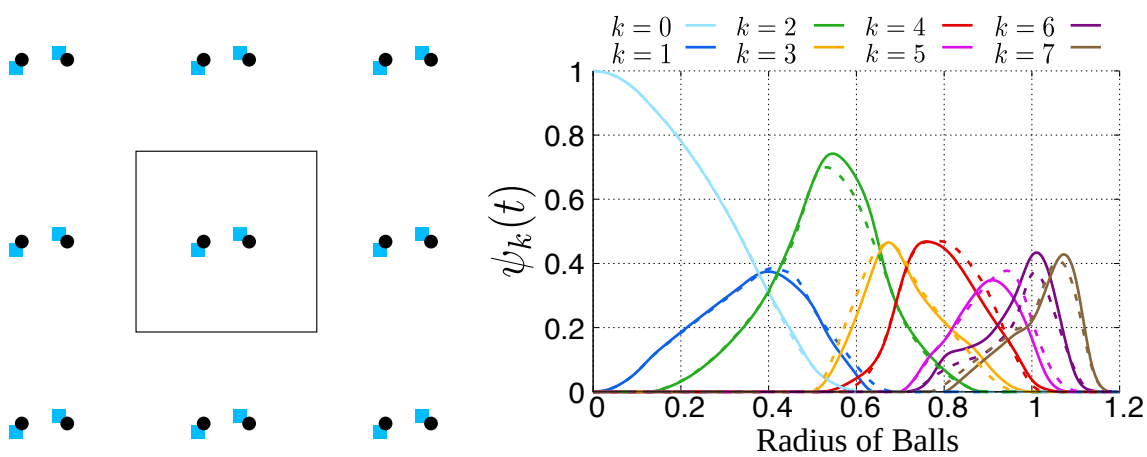

Figure 3.6: **Left:** A periodic set with two points represented by black dots in the square unit cell, and the perturbed periodic set with two points represented by blue squares in the same unit cell. **Right:** The curves of the (undamped) density functions are solid for the original periodic set and dashed for the perturbed periodic set. The small distances between corresponding density functions illustrate Theorem 3.10.

3.4 Completeness of the Density Fingerprint

One of the criteria for having an effective classification of periodic sets is that the classification is complete, meaning that no two non-isometric periodic sets are mapped to the same set of values. We conjecture that the density fingerprint map is complete in dimensions $n \geq 2$, although this remains an open problem. We have however successfully proven that the fingerprint is complete for generic periodic sets in dimension three, and also observe that, apart from in dimension one, we were unable to find a counterexample to completeness in general, despite carrying out an extensive search. We will describe our tests on prime candidates for a counterexample to completeness in Subsection 3.4.1, but first we prove completeness for generic periodic sets in dimension three, with the initial step being to define a generic periodic set. To do this, we need the definition of a circumradius.

Definition 3.11 (Circumradius)**.** The *circumradius* of a set of 2, 3, or 4 linearly independent points in $\mathbb{R}^3$ is the radius of the smallest sphere that passes through all the points in the set. We can similarly define circumradii for edges/1-simplices, triangles/2-simplices and tetrahedrons/3-simplices to be the smallest sphere that passes through all the vertices of the simplex.

For a set of two points, the circumradius is simply half the distance between the two points. For a set of three linearly independent points, the circumradius is the radius of the unique circle that passes through the 3 points. And similarly for a set of four linearly independent points, the circumradius is the radius of the unique sphere that passes through all four points.

The constraints in Definition 3.12 of a generic periodic set are formulated in terms of the circumradii of the edges, triangles and tetrahedrons spanned by subsets of two, three and four points of the periodic set. To avoid infinitely many constraints, we introduce an upper bound on the set of circumradii to consider. In particular, we denote by $L(A;\theta)$ the list of all edges/1-simplices, triangles/2-simplices and tetrahedrons/3-simplices spanned by points of the periodic set A whose circumradii are at most θ.

Definition 3.12 (Generic Periodic Set)**.** A periodic set $A \subset \mathbb{R}^3$ is said to be *generic for a constant threshold* θ if the following three conditions hold (apart from necessary violations due to the periodic nature of the set):

I No two circumradii of different simplices of $L(A;\theta)$ are equal.

II The circumradii of different edges in $L(A;\theta)$ are not related to each other by a factor of 2.

III Let $t \leq \theta$ be the circumradius of a simplex in $L(A;\theta)$. If the simplex is a tetrahedron, there is a unique set of six circumradii in $L(A;\theta)$ such that the edges with twice their lengths can be assembled to a tetrahedron with circumradius t. If the simplex is not a tetrahedron, then there is no such set of six circumradii.

We call an edge a *lattice edge* if its length is equal to the length between two lattice points. A *lattice triangle* is made up of three lattice edges, and a *lattice tetrahedron* is made up of six lattice edges. The distinguishing factor between lattice simplices and non-lattice simplices is that, for a generic periodic set, non-lattice simplices are unique up to translations by lattice vectors. This is not true for lattice simplices (if there are at least two motif points). For example, for points a, b in the motif of a generic periodic set, we can translate a lattice edge starting at a to a lattice edge starting at b, where the translation is not by a lattice vector.

Conditions I, II and III can be expressed in finitely many algebraic equations involving the vectors $\vec{a} \in M$ in the motif M of A and the lattice vectors $\vec{v_1}, \vec{v_2}, \vec{v_3}$ of the lattice Λ of A. Therefore, the set of periodic sets that are generic with threshold θ is open and dense with respect to perturbations of these vectors in the space of all periodic sets with at most $m = |M|$ motif points.

We can now prove Theorem 3.14 which states that the density fingerprint map is complete in dimension three for generic periodic sets. The value of the threshold θ used in Theorem 3.14 is dependent on the value $\mathrm{Rad}(A)$ of a periodic set A.

Definition 3.13 ($\mathrm{Rad}(A)$)**.** Let $A \subset \mathbb{R}^3$ be a periodic set with lattice Λ and unit cell U with diameter D. We define $\mathrm{Rad}(A)$ to be the largest finite circumradius of up to four points in A with pairwise distances between the points at most four times the diameter D of U.

Since the diameter D of a unit cell of a periodic set A is always equal to the distance between two points in A, there exist pairs of points in A that are a distance of $4D$ apart, and so we have that $\mathrm{Rad}(A) \geq 2D$.

Theorem 3.14 (Generic Completeness)**.** *Let $A, Q \subset \mathbb{R}^3$ be non-isometric periodic sets that are generic for the threshold $\theta = \max\{\mathrm{Rad}(A), \mathrm{Rad}(Q)\}$. Then $\Psi(A) \neq \Psi(Q)$.*

Proof. (This proof was principally constructed by Mathijs Wintraecken and Teresa Heiss.) We will show that we can uniquely reconstruct the isometry class of A, $[A]$, from the density fingerprint $\Psi(A)$, therefore showing that any periodic set Q that is non-isometric to A must have a different density fingerprint, $\Psi(Q) \neq \Psi(A)$.

Firstly, we show how we can obtain all tetrahedrons in $L(A;\theta)$ up to isometries. In [18], it is shown that each density function can be expressed as the sum of volumes of intersections of two, three or four balls centred at points in A. These volume formulae are piecewise analytic in the radius t, where the radii at which the formulae are non-analytic correspond to the circumradii of edges, triangles and tetrahedrons spanned by points in A. Therefore, the set of all radii up to θ where at least one density function is non-analytic coincides with the set of circumradii of simplices of $L(A;\theta)$.

To identify all tetrahedrons in $L(A;\theta)$ up to isometries, we treat all circumradii as if they were circumradii of edges. By taking all combinations of six circumradii and seeing if

edges with twice their lengths can be assembled to form a tetrahedron with a circumradius in $L(A;\theta)$, we can obtain all tetrahedrons of $L(A;\theta)$ up to isometries by Condition III in Definition 3.12.

To finish the proof, we want to be able to construct a periodic set in $[A]$, the isometry class of A. If we show that from the tetrahedrons of $L(A;\theta)$ this can be done uniquely, then we are done.

We start the reconstruction with the lexicographically shortest non-lattice tetrahedron from $L(A;\theta)$. If there is no non-lattice tetrahedron, then A is a lattice and can thus be reconstructed from the lexicographically shortest lattice tetrahedron from $L(A;\theta)$: taking such a tetrahedron's three shortest lattice edges that are linearly independent from each other, we obtain the Minkowski-reduced basis of the lattice.

But assuming there is a non-lattice tetrahedron, we will denote it by *abcd*, and without loss of generality assume that *ab* is a non-lattice edge. Placing *abcd* arbitrarily in space (we are only interested in the isometry class of A), we identify all tetrahedra *abce* in $L(A;\theta)$ which possess a face agreeing with *abc*. For each such tetrahedron, there are two ways, related by reflection, of glueing *abce* to *abcd* along their common face *abc*. We call the two different tip positions e_1 and e_2. However, at most only one of them is possible. The triangles *abd* and *abe* are non-lattice and therefore unique in A up to translations by lattice vectors by Condition I. Thus the tetrahedron *abde* (assuming a, b, d, e do not all lie in the same plane) is unique with a certain edge length *de* that is the distance between d and e_i for at most one of e_1 and e_2. If, for instance, this is the distance from d to e_1, we glue *abce* to *abcd* such that e is at e_1.

This glueing procedure yields all points at a distance at most four times the diameter of the unit cell from a, b, c and d by the definition of $\mathrm{Rad}(A)$, except the points that lie on a plane spanned by *abc* or *abd*. This neighbourhood is large enough such that it contains every point in the motif, as well as revealing a lattice basis identified by computing the pairwise distances between the reconstructed points and checking whether they satisfy Condition II. Repeating the reconstructed points with respect to the lattice yields the isometry class of A. As the construction was unique given the genericity conditions, we have that $\Psi(A) \neq \Psi(Q)$. □

3.4.1 Distinguishing Non-generic Periodic Sets

By Theorem 3.14, the density fingerprint map is complete for any generic periodic set $A \subset \mathbb{R}^3$. But what about for non-generic periodic sets? In general, are there any periodic sets $A, Q \subset \mathbb{R}^n$ that are non-isometric but have matching density fingerprints, $\Psi(A) = \Psi(Q)$? We have been unable to construct a proof stating that no such pair of periodic sets exists, yet neither have we found any counterexamples in dimensions $n \geq 2$. In this subsection, we consider a couple of pairs of prime candidates for a counterexample to completeness in dimension $n = 3$. Nevertheless, the density fingerprint map is able to distinguish them, motivating us to make Conjecture 3.19.

We begin by noting that a counterexample to completeness has been found in dimension one by a pair of periodic sets suggested by Morteza Saghafian.

Example 3.15 (Counterexample to Completeness in Dimension One). The following pair of periodic sets are a counterexample to completeness in dimension one. Let $U = \{0, 4, 9\}$ and $V = \{0, 1, 3\}$. It can be checked that the finite sets $U+V$ and $U-V$, and the periodic sets $A = (U+V)+15\mathbb{Z}$ and $Q = (U-V)+15\mathbb{Z}$ cannot be distinguished by the one-dimensional density fingerprint map. Specifically,

$$A = \{x + 15\vec{m} \mid m \in \mathbb{Z},\ x = 0, 1, 3, 4, 5, 7, 9, 10, 12\} \tag{3.10}$$

$$Q = \{x + 15\vec{m} \mid m \in \mathbb{Z},\ x = 0, 1, 3, 4, 6, 8, 9, 12, 14\} \tag{3.11}$$

are non-isometric periodic sets whose density fingerprints are indistinguishable. We note that the pairs of periodic sets $A \times \mathbb{Z}^{n-1}, Q \times \mathbb{Z}^{n-1} \subset \mathbb{R}^n$, $n \geq 2$, are distinguishable by the density fingerprint map, and so this example cannot be used to construct counterexamples to completeness in higher dimensions.

Our search for a counterexample to completeness in higher dimensions starts with homometric structures.

Definition 3.16 (Homometric Structures)**.** Two periodic sets $A, Q \subset \mathbb{R}^n$ are called *homometric* if their multisets of difference vectors are equal (up to isometries): $A - A = Q - Q$, where $A - A$ contains all vectors $\vec{a} - \vec{b}$ for $a, b \in A$.

As described in the proof of Theorem 3.14, density functions are non-analytic only at radii equal to the circumradius of a simplex spanned by points in the periodic set. Hence, a natural candidate for a counterexample to completeness are non-isometric periodic sets that have the same mulitset of pairwise distances between the points in the periodic set, i.e. homometric structures.

Example 3.17 (Homometric Structures). In 1930, Pauling and Shappell [48] discovered a one-parameter family of structures with crystallographic group number 206 and Wyckoff position 24d, where for the parameter μ, taking $\pm\mu$ yields pairs of homometric structures. Following [48], we set $\mu = \pm 0.03$, and refer to the corresponding periodic sets as A and Q. Table 3.2 shows the L_∞-distances between corresponding k-th density functions for $1 \leq k \leq 8$. In particular, all L_∞-distances are positive, from which we can deduce that the density fingerprints of A and Q are distinguishable.

k	1	2	3	4	5	6	7	8
$\left\lVert \psi_k^A - \psi_k^Q \right\rVert_\infty$	0.0039	0.0090	0.0247	0.0246	0.0159	0.0138	0.0245	0.0279

Table 3.2: L_∞-distances between the corresponding k-th density functions, $1 \leq k \leq 8$, of the two homometric structures A and Q from Example 3.17.

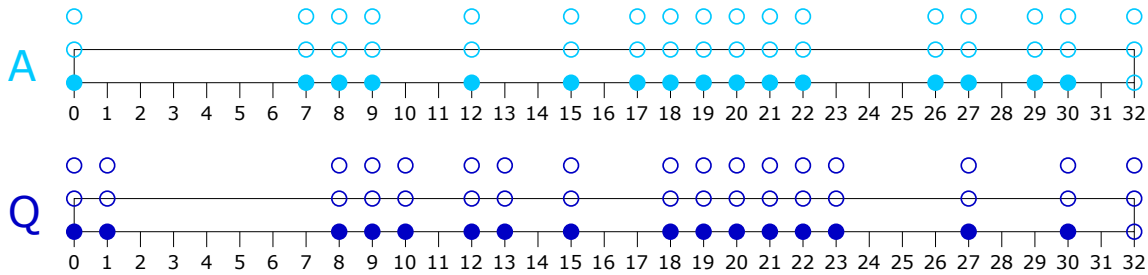

Figure 3.7: Periodic sets A and Q from Example 3.18, pictured with rectangular unit cells in dimension two for simplicity. Filled dots belong to the motifs while unfilled dots illustrate the periodicity.

Example 3.18 describes a pair of periodic sets which not only are homometric, but in fact the multisets of all triplets of points coincide too (up to isometries). Even in this case, the density fingerprints are distinguishable.

Example 3.18. Let $A_1, Q_1 \subset \mathbb{R}$ be periodic sets with periodicity 32, each with 16 points in their motifs. Explicitly,

$$A_1 = \{x + 32n \mid n \in \mathbb{Z},\ x = 0, 7, 8, 9, 12, 15, 17, 18, 19, 20, 21, 22, 26, 27, 29, 30\} \tag{3.12}$$

$$Q_1 = \{x + 32n \mid n \in \mathbb{Z},\ x = 0, 1, 8, 9, 10, 12, 13, 15, 18, 19, 20, 21, 22, 23, 27, 30\}. \tag{3.13}$$

In [29], it is shown that A_1 and Q_1 are non-isometric periodic sets that have the same multisets of pairs and triplets of points up to isometries. From A_1 and Q_1 we can obtain non-isometric periodic sets $A, Q \subset \mathbb{R}^3$ that again share this property by taking the Cartesian product of A_1 and respectively Q_1 with $\mathbb{Z}^2$, see Figure 3.7. Yet, as shown in Table 3.3, although the first four density functions of A and Q agree, there is a nonzero distance between higher density functions, and so we can conclude that the density fingerprint map distinguishes these two periodic sets. The ability of the density fingerprint map to distinguish these periodic sets may be due to the fact that it takes into account the additional information of the number of points in the periodic set that are within each circumsphere.

k	0	1	2	3	4	5	6	7	8
$\left\lVert \psi_k^A - \psi_k^Q \right\rVert_\infty$	0.000	0.000	0.000	0.000	0.005	0.007	0.013	0.022	0.007

Table 3.3: L_∞-distances between the first eight density functions of the periodic sets A and Q from Example 3.18.

Both the proof of generic completeness, and the ability of the density fingerprint map to distinguish strongly related non-generic periodic sets, leaves us hopeful that the following conjecture is true.

Conjecture 3.19 (Completeness of the Density Fingerprint)**.** *Let $A, Q \subset \mathbb{R}^n$, $n \geq 2$, be two non-isometric periodic sets. Then $\Psi(A) \neq \Psi(Q)$.*

Comparing the Density Fingerprints of FCC and HCP

Slightly tangential to this discussion on completeness, we think it is of interest to compare two related periodic sets familiar to crystallography: the face-centred cubic lattice (FCC) and the hexagonal close packing of spheres (HCP). Both periodic sets have the same packing density of spheres in $\mathbb{R}^3$, specifically the greatest packing density possible.

When we compare the density fingerprints of FCC and HCP, as we see in Table 3.4, the first and second density functions are identical, but the third density function and subsequent density functions distinguish the two periodic sets. We compare this with the persistence diagrams of the multicovers of FCC and HCP: the diagrams of the first three multicovers are the same, while we have to wait until the persistence diagram of the four-fold cover to distinguish between the two periodic sets.

k	1	2	3	4	5	6	7	8
$\left\lVert \psi_k^A - \psi_k^Q \right\rVert_\infty$	0.0000	0.0000	0.0280	0.0571	0.0946	0.0405	0.0551	0.1402

Table 3.4: The L_∞-distances between the corresponding density functions of the face-centred cubic lattice and the hexagonal close packing of spheres.

3.5 Computing Density Functions

It is in this section where the content of this chapter intersects with the Voronoi zones of Chapter 2, as we describe how the density functions that comprise the density fingerprint can be computed. In particular, it is Theorem 3.21 that relates the two chapters, from which we can deduce that, in dimension n, the challenge of computing the density fingerprint can be reduced to the challenge of computing the volume of intersection of a single solid sphere with an n-simplex. A formula for such an intersection in dimension three is given in Theorem 3.31.

Definition 3.20 ($\psi_k^A(t)$ and $\varphi_k^A(t)$)**.** In Definition 3.2, for a periodic set A and a unit cell U of A, we defined $\psi_k^A(t)$ to be the fractional volume of the unit cell that is covered by *exactly* k balls of $B(A;t)$. We define $\varphi_k^A(t)$ to be the fractional volume of the unit cell that is covered by *at least* k balls of $B(A;t)$,

$$\varphi_k^A(t) = \frac{\mathrm{Vol}[\{p \text{ in at least } k \text{ balls of } B(A;t) \mid p \in U\}]}{\mathrm{Vol}[U]}.$$

Therefore, for $k \geq 0$, we have the relation $\psi_k^A(t) = \varphi_k^A(t) - \varphi_{k+1}^A(t)$.

Theorem 3.21. *Let $A = M + \Lambda$ be a periodic set in $\mathbb{R}^n$ with lattice Λ and motif $M \subset U$, where U is the unit cell of Λ, and let $k \in \mathbb{Z}_{\geq 1}$. Then the fractional volume of the unit cell U covered by at least k balls of $B(A;t)$, $\varphi_k^A(t)$, is given by the equation*

$$\varphi_k^A(t) = \frac{1}{\mathrm{Vol}[U]} \sum_{a \in M} \mathrm{Vol}[Z_k(A;a) \cap B(a;t)]. \tag{3.14}$$

Proof. Let Z_k^M be the union over all motif points $a \in M$ of the Voronoi zones $Z_k(A;a)$. By Theorem 2.12, the volume of Z_k^M is equal to the volume of the unit cell U. Moreover, similarly to Lemma 2.11, there is a bijection up to zero measure sets between Z_k^M and the unit cell U consisting of piecewise translations by lattice vectors. Therefore, the volume of the unit cell covered by at least k balls of $B(A;t)$ is equal to the volume of Z_k^M covered by at least k balls of $B(A;t)$.

Now, consider the point $x \in Z_k^M$ that lies in the interior of $Z_k(A;a)$ for some $a \in M$. Hence x has a as its unique k-th closest point in A. Therefore, x is covered by at least k balls of $B(A;t)$ if and only if $x \in B(a;t)$. This is because if $x \in B(a;t)$, then x must also be covered by the balls of radius t centred at the $k-1$ points of A that are closer to x than a, and similarly if $x \notin B(a;t)$, then there is at most $k-1$ balls of radius t centred at points of A that can cover x.

Hence the volume of $Z_k(A;a)$ that is covered by at least k balls of $B(A;t)$ is the volume of the intersection $Z_k(A;a) \cap B(a;t)$, from which we can deduce that the fractional volume of the unit cell covered by at least k balls of $B(A;t)$ is indeed given by Equation 3.14. □

As mentioned in Definition 3.20, the k-th density function of a periodic set A, ψ_k^A, is related to φ_k^A by the equation $\psi_k^A(t) = \varphi_k^A(t) - \varphi_{k+1}^A(t)$. Hence Theorem 3.21 implies the following corollary.

Corollary 3.22. *Let $A = M + \Lambda$ be a periodic set in $\mathbb{R}^n$ with lattice Λ and motif $M \subset U$, where U is the unit cell of Λ, and let $k \in \mathbb{Z}_{\geq 1}$. Then we have*

$$\psi_k^A(t) = \frac{1}{\mathrm{Vol}[U]} \sum_{a \in M} \left(\mathrm{Vol}[Z_k(A;a) \cap B(a;t)] - \mathrm{Vol}[Z_{k+1}(A;a) \cap B(a;t)]\right), \tag{3.15}$$

while for the zeroth density function we have

$$\psi_0^A(t) = 1 - \frac{1}{\mathrm{Vol}[U]} \sum_{a \in M} \mathrm{Vol}[Z_1(A;a) \cap B(a;t)]. \tag{3.16}$$

3.5.1 Volume of Sphere-Tetrahedron Intersections

By Theorem 2.6 we have that k-th Voronoi zones are a union of polytopes and thus can be triangulated. Therefore, we can deduce from Corollary 3.22 that, in dimension n, the k-th density function $\psi_k^A(t)$ can be computed as the summation of a set of volumes, where each

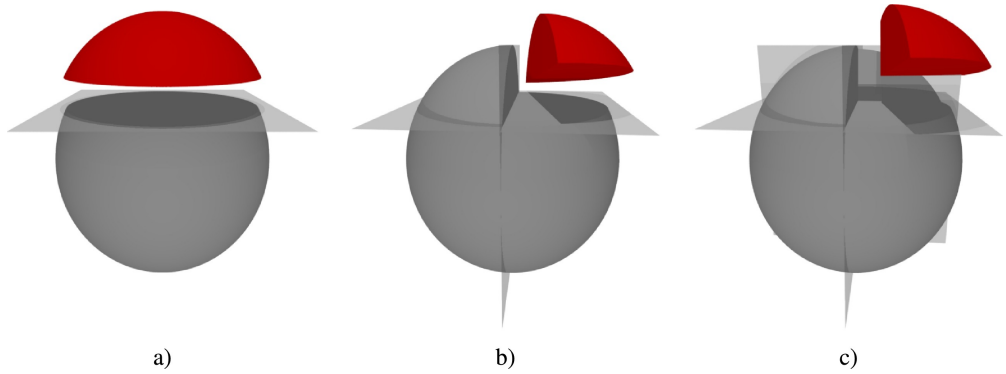

Figure 3.8: [57, Figure 1] The three significant ways up to three half-spaces can intersect a solid sphere in $\mathbb{R}^3$. Such intersections are called a spherical cap, a spherical wedge and a spherical cone respectively.

summand is the volume of intersection of a single solid sphere with a solid n-simplex. We discuss in this subsection the precise equations required to compute such an intersection in dimension three, and note that similar yet simpler equations exist for dimension two.

There are three main, increasingly complex scenarios to consider, as depicted in Figure 3.8, involving one, two and three half-spaces, which we respectively call spherical caps, spherical wedges and spherical cones.

Definition 3.23 (Half-Space H)**.** For a plane $pl \subset \mathbb{R}^3$ defined by the equation $ax + by + cz + d = 0$, the corresponding *half-space* $H \subset \mathbb{R}^3$ is the set of all points $(x, y, z) \in \mathbb{R}^3$ such that $ax + by + cz + d \geq 0$. The *opposite half-space*, $\overline{H}$, is the set of all points $(x, y, z) \in \mathbb{R}^3$ such that $ax + by + cz + d \leq 0$, and so we have $H \cap \overline{H} = pl$.

Definition 3.24 (Spherical Cap $S_{\text{cap}}(H)$)**.** Let S be a solid sphere and let H be a half-space, both in $\mathbb{R}^3$. We define the *spherical cap*, $S_{\text{cap}}(H)$, to be the intersection of S and H: $S_{\text{cap}}(H) = S \cap H$, see image (a) in Figure 3.8. The *height* h of a spherical cap is the maximum orthogonal distance of a point in $S_{\text{cap}}(H)$ to the boundary of H.

Intersecting a spherical cap with another half-space results in a spherical wedge.

Definition 3.25 (Spherical Wedge $S_{\text{wedge}}(H_1, H_2)$)**.** Let $S \subset \mathbb{R}^3$ be a solid sphere and let $H_1, H_2 \subset \mathbb{R}^3$ be two non-parallel half-spaces, whose boundaries intersect in a line that passes through S. We define the *spherical wedge*, $S_{\text{wedge}}(H_1, H_2)$, to be the intersection of S, H_1 and H_2: $S_{\text{wedge}}(H_1, H_2) = S \cap H_1 \cap H_2$, see image (b) in Figure 3.8.

It will be helpful as we consider volumes of spherical wedges to introduce a slightly simpler geometric object, the regularised spherical wedge.

Definition 3.26 (Regularised Spherical Wedge $S_{\text{rwedge}}(H_1, H_2)$)**.** [57, Section 4.3] A *regularised spherical wedge*, $S_{\text{rwedge}}(H_1, H_2)$, is a spherical wedge such that at least one of the boundaries of the half-spaces H_1, H_2 passes through the centre of the sphere S.

Intersecting a spherical wedge with another half-space results in a spherical cone.

Definition 3.27 (Spherical Cone S_{cone})**.** Let $S \subset \mathbb{R}^3$ be a solid sphere and let $H_1, H_2, H_3 \subset \mathbb{R}^3$ be three half-spaces whose boundaries intersect at a single point $p = \partial H_1 \cap \partial H_2 \cap \partial H_3$ which lies within S. We define the *spherical cone*, $S_{\text{cone}}(H_1, H_2, H_3)$, to be the intersection of S, H_1, H_2 and H_3: $S_{\text{cone}}(H_1, H_2, H_3) = S \cap H_1 \cap H_2 \cap H_3$, see image (c) in Figure 3.8. The point p is called the *vertex* of the spherical cone.

Having defined spherical caps, wedges and cones, we present formulae for their volumes. The volume of a spherical cap is well documented and there exists a simple formula in terms of the height h of the cap and the radius t of the sphere.

Lemma 3.28 (Volume of a Spherical Cap)**.** *Let $S_{cap}(H)$ be a spherical cap with height h, formed from a sphere with radius t. If $t \geq h$, then the volume $\text{Vol}[S_{cap}(H)]$ of the spherical cap is given by $\text{Vol}[S_{cap}(H)] = \frac{1}{3}\pi h^2(3t - h)$. If $h > t$, we let $h' = 2t - h$, and then $\text{Vol}[S_{cap}(H)] = \frac{4}{3}\pi t^3 - \frac{1}{3}\pi h'^2(3t - h')$.*

The volume of a spherical wedge is far more complicated. However, in [57, Section 4.2], they show that the volume of any spherical wedge can be given as the summation of the volumes of regularised spherical wedges (in which they show that there are three distinct cases to consider). In addition, they provide a formula for the volume of a regularised spherical wedge.

Lemma 3.29. *[57, Equation 5] Let $S_{rwedge}(H_1, H_2)$ be a regularised spherical wedge of a sphere S with radius t. Let α be the angle between the two half-spaces H_1 and H_2 and let d be the shortest distance from the line of intersection of the boundaries of H_1 and H_2 to the centre of S. Let $a = d\sin(\alpha)$, $b = \sqrt{|t^2 - d^2|}$, and $c = d\cos(\alpha)$. Then the volume of $S_{rwedge}(H_1, H_2)$ is given by the equation*

$$\text{Vol}[S_{rwedge}(H_1, H_2)] = \frac{1}{3}abc + a\left(\frac{1}{3}a^2 - t^2\right)\arctan\left(\frac{b}{c}\right) + \frac{2}{3}t^3\arctan\left(\frac{b\sin(\alpha)}{t\cos(\alpha)}\right). \quad (3.17)$$

The volume of a spherical cone can be broken down into the sum of the volumes of a tetrahedron, three spherical wedges and a spherical cap.

Lemma 3.30. *Let $S_{cone}(H_1, H_2, H_3)$ be a spherical cone of a sphere S with vertex p, and let l_1, l_2, l_3 be the lines of intersections of the boundaries of H_1 and H_2, H_1 and H_3, and H_2 and H_3 respectively. Let p_1, p_2, p_3 be the respective intersection points of the lines l_1, l_2, l_3 with the sphere S that lie within the remaining half-space (i.e. p_1 is the point of intersection of l_1 with S that lies within the half-space H_3). Let the plane that contains p_1, p_2, p_3 be the*

boundary of a new half-space H_4 that is oriented so that the vertex p of the spherical cone is not contained within H_4. Then the volume of the spherical cone is given by the equation

$$\begin{aligned}\mathrm{Vol}[S_{cone}(H_1, H_2, H_3)] = \mathrm{Vol}[\mathrm{Tet}(p, p_1, p_2, p_3)] + \mathrm{Vol}[S_{cap}(H_4)] \\ - \mathrm{Vol}[S_{wedge}(\overline{H_1}, H_4)] - \mathrm{Vol}[S_{wedge}(\overline{H_2}, H_4)] \\ - \mathrm{Vol}[S_{wedge}(\overline{H_3}, H_4)],\end{aligned} \tag{3.18}$$

where $\mathrm{Tet}(p, p_1, p_2, p_3)$ *is the tetrahedron spanned by the points* p, p_1, p_2, p_3.

Proof. The equation is obtained simply by using inclusion-exclusion principles. □

Finally, we have arrived at Theorem 3.31 which provides an inclusion-exclusion formula to compute the volume of intersection of a sphere with a tetrahedron.

Theorem 3.31 (Volume of Sphere-Tetrahedron Intersection)**.** *Consider a sphere S with radius t and a tetrahedron T, both in $\mathbb{R}^3$. Let T be the intersection of four half-spaces H_1, H_2, H_3, H_4, and consider the four opposite half-spaces $\overline{H_1}, \overline{H_2}, \overline{H_3}, \overline{H_4}$ (Definition 3.23). Let $H = \{\overline{H_1}, \overline{H_2}, \overline{H_3}, \overline{H_4}\}$ be the set of these four opposite half-spaces. Then the volume of the intersection between S and T,* $\mathrm{Vol}[S \cap T]$ *is given by the equation*

$$\begin{aligned}\mathrm{Vol}[S \cap T] = \frac{4}{3}\pi t^3 - \sum_{a \in H} \mathrm{Vol}[S_{cap}(a)] + \sum_{a,b \in H} \mathrm{Vol}[S_{wedge}(a, b)] \\ - \sum_{a,b,c \in H} \mathrm{Vol}[S_{cone}(a, b, c)],\end{aligned} \tag{3.19}$$

where a, b in the third term, and similarly a, b, c in the final term, are distinct half-spaces in the set H.

Proof. Again, the equation is obtained simply by using inclusion-exclusion principles. □

We conclude this discussion on the computations of density functions by noting that the time complexity to compute the k-th density function is limited by the construction of the $(k+1)$-th Voronoi zones of Chapter 2. In Theorem 2.18, we state that computing the first k Voronoi zones has time complexity $\mathcal{O}(m^n(2k)^{n^2}(\log(m) + n\log(2k)))$. But we note that, if the packing and covering radii of the periodic set are known, by using techniques described in the proof of Theorem 3.10 where we bound the radius t beyond which the k-th density function vanishes, Voronoi zones, and hence density functions, can be computed in cubic time in the order k in dimension three.

3.6 An Application to Crystal Structure Prediction

As mentioned at the start of this chapter, an effective classification of crystal structures will be a useful tool in the field of Crystal Structure Prediction (CSP). Whilst applying the

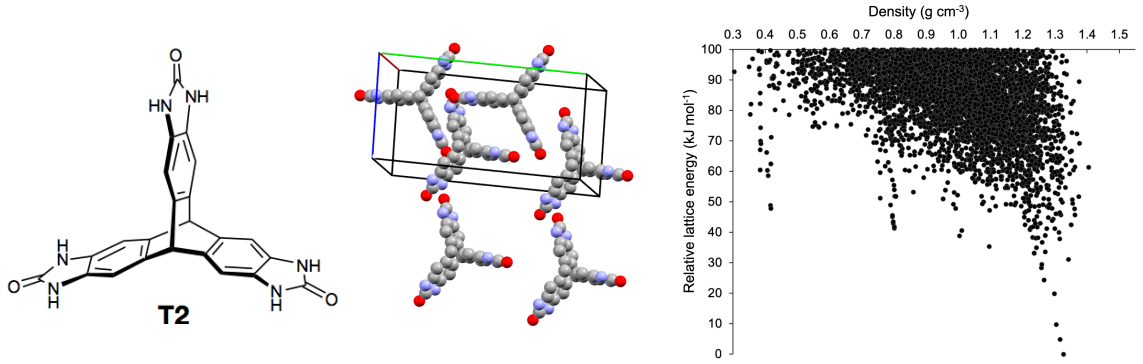

Figure 3.9: **Left:** a T2 molecule. **Middle:** the T2-δ crystal with highlighted unit cell. **Right:** the output of CSP for the T2 molecule. It is a plot of 5679 simulated T2 crystal structures [49, Figure 2c], each represented by two coordinates: the physical density (atomic mass within a unit cell divided by the unit cell volume) and energy (determining the crystal's thermodynamic stability). Structures at the bottom of the 'downward spikes' are likely to be stable.

tool to CSP is still in its infancy, we can bring to light an example of its usefulness that leads us to believe this will be a very helpful tool to materials scientists going forwards.

In [49], Crystal Structure Prediction is applied to several molecules, with the hope of finding low density molecular crystal structures, which is rare because molecules tend to pack densely. One such molecule is benzimidazolone T2, and interestingly, the output of CSP (called an energy-function-structure map) for T2 (Figure 3.9) depicted several 'downward spikes' at densities lower than the only previously reported T2 crystal structure (T2-α). It is likely that structures at the bottom of these 'downward spikes' are stable as there is a significant energy barrier preventing the structures from deforming into more dense arrangements.

The authors of [49] predicted from this output four new stable crystal polymorphs of T2 in addition to the previously reported structure T2-α, including one structure that was predicted to have half the density of T2-α. Subsequently, our collaborators in the Materials Innovation Factory at the University of Liverpool were able to synthesise these four new T2 polymorphs (T2-β, T2-γ, T2-δ, T2-ϵ), most significantly T2-γ that indeed has half the density of T2-α, and in fact has the lowest density reported for a molecular crystal.

Our collaborators scanned the synthesised crystals using X-ray powder diffraction yielding Crystallographic Information Files (CIFs), each containing the corresponding dimensions of the structure's unit cell and the fractional coordinates of atoms in the motif, among other data. These files were then compared with the simulated structures that comprise the output of CSP, either by using their physical densities alongside the COMPACK algorithm – which compares only a finite portion of the structure – or by looking at visualisations of

$\left\lVert \psi_k^A - \psi_k^Q \right\rVert_\infty$	$k=0$	1	2	3	4	5	6	7
T2-α vs entry 99	0.0042	0.0092	0.0125	0.0056	0.0099	0.0088	0.0127	0.0099
T2-β vs entry 28	0.0157	0.0156	0.0159	0.0224	0.0334	0.0396	0.0357	0.0454
T2-γ vs entry 62	0.0020	0.0080	0.0128	0.0155	0.0153	0.0250	0.0296	0.0391
T2-δ vs entry 09	0.0610	0.0884	0.1267	0.0676	0.0915	0.0801	0.0733	0.0388
T2-ϵ vs entry 01	0.0132	0.0152	0.0207	0.0571	0.0514	0.0431	0.0468	0.0550
T2-β' vs entry 09	0.2981	0.2631	0.3718	0.3747	0.2563	0.2360	0.3161	0.3232

Table 3.5: **First five rows:** the L_∞-distances between the first eight pairs of density functions of physically synthesised T2 crystals (T2-α, T2-β, etc.) and the simulated structures that had predicted them from the CSP output T2 dataset (entry XX). **Last row:** the suspiciously larger L_∞-distances revealed the mix-up of the files T2-δ and T2-β' and thus led to the depositing of the initially omitted T2-δ Crystallographic Information File into the Cambridge Structural Database.

the crystal structures. These comparisons showed that the synthesised crystals matched closely entries from the CSP output, verifying the prediction. Our collaborators then deposited these CIFs into the globally used Cambridge Structural Database which contains over one million structures.

We desired to use our newly developed density fingerprint to verify our collaborators' matchings between the synthesised crystals T2-α, T2-β, T2-γ, T2-δ and T2-ϵ and the simulated structures Entry 99, Entry 28, Entry 62, Entry 09, Entry 01 from the CSP output that they had been matched with.

We did so by computing, for each of the five matches, the L_∞-distances between the first eight density functions of the synthesised and simulated structures, which are recorded in Table 3.5. As one is the prediction of the other, we expected to see small distances. And for four of the five structures this was true: T2-γ, for example, always has an L_∞-distance of less than 0.04 over the first eight pairs of density functions. However, when we came to check the distances between density functions of T2-δ with its matched simulated structure Entry 09, we were surprised to see large distances (the final row of Table 3.5). It turned out that a mix-up of files had happened, and what was uploaded to the Cambridge Structural Database as T2-δ was in fact T2-β' (a crystal from the T2-β family). The density fingerprint revealed this error, which was verified by chemists upon a visual inspection, and it is because of this that T2-δ was subsequently correctly deposited to the Cambridge Structural Database. Plots of the density functions of correctly matched synthesised and simulated structures can be seen in Figure 3.10.

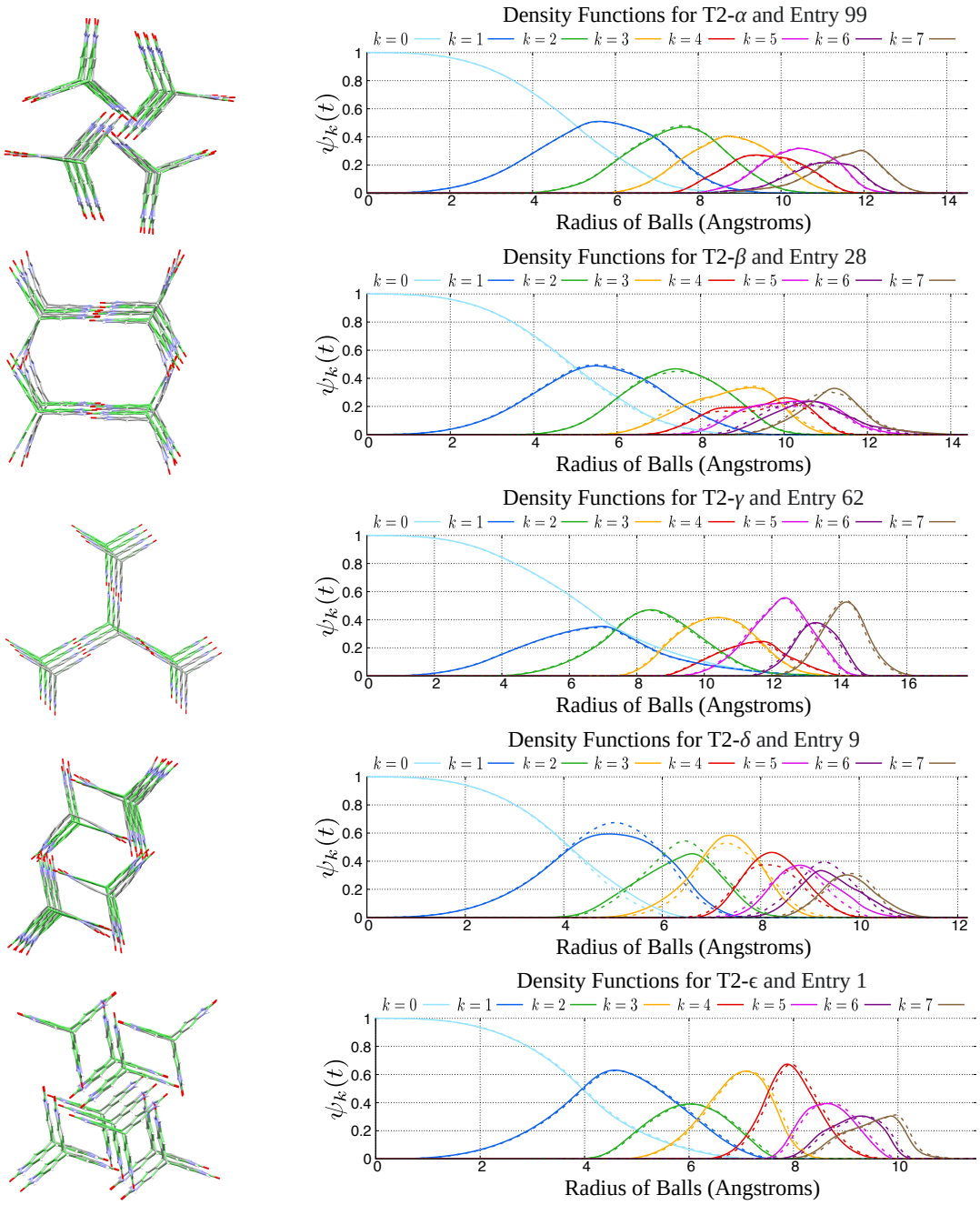

Figure 3.10: **Left:** synthesised T2 crystals (curved grey molecules) superimposed with their simulated versions (straight green molecules). **Right:** the first eight density functions of the periodic sets formed from the molecular centres of the synthesised T2 crystals (solid curves) and their corresponding simulated structures (dashed curves).

3.7 Discussion

The main contribution of this chapter is a fingerprint map from periodic sets (which model crystals) to an infinite sequence of density functions. In dimension $n = 3$, this map is invariant under isometries (Lemma 3.4), Lipschitz continuous (Theorem 3.10) and generically complete (Theorem 3.14). We conjecture that the fingerprint map is complete without the genericity assumption (Conjecture 3.19), but acknowledge that this remains an open question. In this respect, it is worth noticing that our proof of generic completeness makes only limited use of the order, k, at which the circumradius of an edge, triangle, or tetrahedron is detected (where the order is the number of points in the respective circumsphere). Is this additional information sufficient to prove general completeness?

We also link the density fingerprint to the Voronoi zones of Chapter 2 via Theorem 3.21, and give a description of how density functions can be computed as the sum of volumes of sphere-tetrahedron intersections, for which we give a detailed description culminating in Theorem 3.31.

We present an application to Crystal Structure Prediction in Section 3.6, and comment that we expect the fingerprint will also be used to simplify the large output datasets produced by CSP by comparing simulated structures with each other, thus speeding up what is currently a slow process.

Collaborators in the Materials Innovation Factory at the University of Liverpool have simultaneously been using different approaches to tackle the same problem of classifying crystal structures up to isometry [6]. Two continuous metrics on all lattices have been introduced in [43], whilst the fast Average Minimum Distances of [62] form an infinite sequence of continuous isometry invariants whose asymptotic behaviour is described. The isosets of [5] partition all points in a periodic set into equivalence classes.

We close this chapter with two extensions of the results presented. Different types of atoms are often modelled as balls with different radii. A possible geometric formalism is that of weighted points and the power distance [7]. Our geometric results generalise to this setting, although some need a careful adaptation. Our continuity result for periodic sets (Theorem 3.10) also generalises to non-periodic Delone sets that allow for a reasonable definition of density functions. Considering that quasi-periodic crystals can be modelled as such, finding out how far such an extension can be pushed may be a worthwhile direction of future research.

Chapter 4

Skeletonisation Algorithms

(This chapter is based on the paper "Skeletonisation algorithms with theoretical guarantees for unorganised point clouds with high levels of noise" authored by P.S. and V. Kurlin and published in Pattern Recognition in 2021 [56].)

The central problem in Data Science is to represent unorganised data in a simple and meaningful form. In this chapter, we focus on input data that can either be represented by a cloud of points in a metric space (Definition 1.15), or by a connected weighted graph (Definition 1.17), whose vertices are data points and the weights of the edges correspond to the distances between the endpoints.

A common approach when faced with big data of this kind is to employ clustering techniques to group entries of the dataset that are deemed to be related. However, real data rarely splits into well-defined clusters, and perhaps a more informative approach is to approximate the data by a skeleton, where data points or clusters of data points are connected to neighbours by simplices. Such skeletons allow for a better visualisation of the dataset's structure, whilst branches of the skeleton may reveal new classes of data points.

We will focus in particular on one-dimensional skeletons, which can be thought of as graphs, although some of the results we discuss can be extended to any dimension. One-dimensional skeletons have already proven useful in curve recognition for surfaces [59], extracting topological shapes of micelles [23], and for posture identification [47].

We are particularly interested in skeletonisation algorithms that have theoretical guarantees with respect to the quality of their outputs and can therefore provide solutions to the following fundamental skeletonisation problem:

Problem 2 (Data Skeletonisation Problem)**.** Given a noisy point cloud C sampled from a graph G in a metric space M, can you find conditions on G and C such that the reconstructed graph G' has the same first homology group as G ($H_1(G') \cong H_1(G)$, see Definition 1.25) and geometrically approximates G in the sense that $G' \subset G^\alpha$ and $G \subset (G')^\alpha$ for a suitable parameter α depending on G and C (see Definition 1.22)?

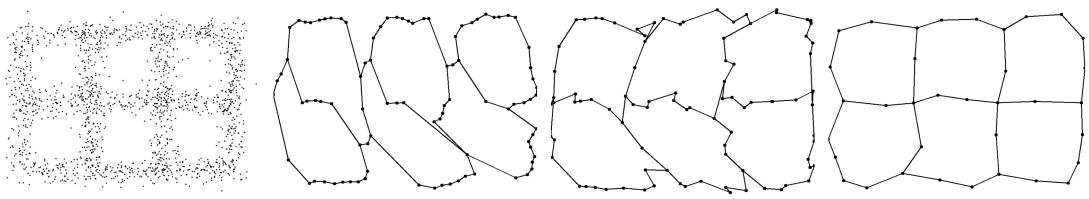

Figure 4.1: **Left:** a point cloud from the PGC Dataset (Section 4.7) sampled from the $G(3,2)$ graph with Gaussian noise ($\sigma = 0.1$). **Right:** reconstructions by Mapper, α-Reeb, and simHoPeS respectively.

Informally, the homology condition of the Data Skeletonisation Problem implies that a reconstructed graph G' can be continuously deformed into the original graph G. Meanwhile, the geometric approximation condition requires that the two graphs G and G' are close to each other with respect to a distance between graphs, i.e. one is in a small neighbourhood of the other, and vice versa.

Mapper [51], α-Reeb [11] and HoPeS [37] (see Figure 4.1 for example outputs, as well as Sections 4.2, 4.3 and 4.4 respectively) are three relevant skeletonisation algorithms that, as well as sharing similar inputs and outputs, solve the Data Skeletonisation Problem. Thus a discussion and comparison of these algorithms is the focus of this chapter.

4.0.1 Contributions and Chapter Outline

The main contributions of this chapter are as follows:

- We present a simpler proof of the optimality of HoPeS (Optimality Theorem 4.21) than for the higher dimensional version described in [32].
- The key stability theorem (Theorem 4.11) of Topological Data Analysis is extended to the graph reconstruction theorems (Theorems 4.28 and 4.32), which were first announced in [37] and are proven here. Corollary 4.33 proves a global stability of derived subskeletons of HoPeS for the first time, justifying its application to noisy data [36, 38].
- Section 4.9 extensively compares the three skeletonisation algorithms on artificial and real data, analysing topological and geometric measures alongside their runtimes. The generation of the artificial dataset is explained in Section 4.7.

Chapter outline: Section 4.1 reviews relevant skeletonisation algorithms. The Mapper algorithm, the α-Reeb algorithm and HoPeS are introduced in Sections 4.2, 4.3, and 4.4

respectively. Optimality and reconstruction guarantees of HoPeS are discussed in Sections 4.5 and 4.6. The generation of a synthetic dataset is described in Section 4.7, which is used in the experimental comparison of the skeletonisation algorithms in Section 4.9. Section 4.10 concludes the chapter with a discussion on the advantages and disadvantages of each algorithm.

4.1 Review of Related Work on Skeletonisation Algorithms

Among many skeletonisation algorithms we review the most relevant ones, some of which we do not discuss further in the paper since either they do not accept as input any point cloud or they cannot provide guarantees in relation to the Data Skeletonisation Problem.

Iterative algorithms: Singh et al. [52] iteratively approximated a point cloud $C \subset \mathbb{R}^n$ by a subgraph of a Delone triangulation, which requires $\mathcal{O}(m^{\lceil n/2 \rceil})$ time for m points of C and three thresholds: a minimum number of edges K in a cycle, plus $\delta_{\min}$ and $\delta_{\max}$ which are required for inserting and merging second order Voronoi regions. Similar parameters are needed for principal curves [33] which were later extended to iteratively computed elastic maps [28]. Since it is hard to estimate a rate of convergence for iterative algorithms, we discuss below non-iterative methods.

Skeletonisation via Reeb graphs: starting from a noisy sample C of an unknown graph G, X. Ge et al. [26] considered the Reeb graph of the Vietoris-Rips complex of C at scale α. The α-Reeb graph G was introduced by F. Chazal et al. [11] for a finite metric space C at a user-defined scale α. If C is ϵ-close to an unknown graph with edges of minimum length 8ϵ, the output G is $34(\beta(G)+1)\epsilon$-close to the input C, where $\beta(G)$ is the first Betti number of G (see [11, Theorem 4.9]). The α-Reeb graph has a metric, but it is not embedded into any space even if $C \subset \mathbb{R}^2$. The algorithm to compute α-Reeb graphs is fast with time complexity $\mathcal{O}(m \log(m))$ for m input points of C.

Mapper [51] outputs a network of interlinked clusters by using a user-defined filter function $f\colon C \to \mathbb{R}$ to associate different clusters of a point cloud C. M. Carrière et al. [10, Theorem 5.2] found a connection between the output of Mapper and the Reeb graph via MultiNerve Mapper.

Metric graph reconstruction: M. Aanjaneya et al. [1] studied a related problem approximating a metric on a large input graph Y by a metric on a small output graph $\hat{X}$. If Y is a good ϵ-approximation to an unknown graph X, then [1, Theorem 2] is the first guarantee for the existence of a homeomorphism $X \to \hat{X}$ that distorts the metrics on X and $\hat{X}$ with a multiplicative factor $1 + c\epsilon$ for $c > \frac{30}{b}$, where $b > 14.5\epsilon$ is the length of the shortest edge of X.

Graph reconstruction by discrete Morse theory: a Homological Spanning Forest [42] uses a given pixel grid of 2D images, and hence cannot be applied to an arbitrary point cloud. Similarly, the recent algorithm by T. Del et al. [58] requires, in addition to a point

cloud, a density field $\rho\colon \mathbb{R}^d \to \mathbb{R}$ (usually on a regular grid) which concentrates around a hidden graph.

4.2 The Mapper Algorithm

The Mapper algorithm is a skeletonisation framework introduced by G. Singh et al. [51], which aims to give a simple description of a dataset by a network of interlinked clusters. It takes as **input a point cloud** C alongside some additional parameters, and **outputs a simplicial complex**. The further parameters that the user can vary are outlined below:

- **A filter function** $f\colon C \to Y$ assigns to each point of C a value from the parameter space Y, which is commonly $\mathbb{R}$. Examples of filter functions include a density estimator or the distance from a base point, where the distance could either be the Euclidean distance or the distance from the base point within a neighbourhood graph.
- **A covering of the range of f:** the range of the filter function f must be covered by overlapping regions, such as line intervals if $Y = \mathbb{R}$. There are two parameters of the covering that can be varied by the user, namely the number of regions and the ratio of overlap. These parameters can be used to control the resolution of the output simplicial complex.
- **A clustering algorithm** is needed to group points in the preimage of f for a given region. The choice of algorithm may need to vary depending on the data being used.

For an input point cloud C where each point has been assigned a value by the filter function, the algorithm has two main stages:

Stage 1: the range of the filter function is covered by a set of overlapping regions $\mathcal{I}$, and for each region I_i in the covering, we cluster the preimage $f^{-1}(I_i) = \{p \in C \mid f(p) \in I_i\}$ by the chosen clustering algorithm. Each cluster is represented by a vertex (zero-dimensional simplex) in the output complex.

Stage 2: if a set of $k \geq 2$ clusters (from any region) share a common point in the point cloud C, the corresponding k vertices span a $(k-1)$-dimensional simplex in the output simplicial complex.

The Mapper algorithm is a versatile tool that can be a useful method to visualise large datasets. However, a significant drawback of the method is that the many important user-defined parameters often require an existing knowledge of the data in order to make good selections that give meaningful outputs.

In the experiments of Section 4.9, the Mapper algorithm is employed as follows. The filter function is the Euclidean distance from a base point, which is chosen to be the most distant point from a random point in C. Therefore, the parameter space $Y = \mathbb{R}$, which is covered by overlapping intervals. The overlap of contiguous intervals is fixed at 50%,

which is the highest percentage such that the algorithm outputs a one-dimensional complex, while the number of intervals is optimised in the experiments. As for clustering, in the context of the experiments of Section 4.9, the clustering algorithm should not depend on a predetermined number of clusters if Mapper is to work effectively, which rules out the popular k-means clustering algorithm. Although we tried single-edge clustering, we found that for the noisy point clouds of the PGC Dataset (Section 4.7), the algorithm DBSCAN is more appropriate, and its choice of parameters is discussed in Subsection 4.2.1.

4.2.1 DBSCAN

DBSCAN [24] is a density-based spatial clustering for applications with noise. It requires two parameters: the radius ϵ around a point within which we search for neighbours; and minPts which is the minimum number of points required within a neighbourhood before a cluster is formed. Given an input point cloud C, there are three main stages to DBSCAN:

Stage 1: a single point $p_1 \in C$ is randomly selected, and the set $\text{Nbhd}(p_1)$ of all points within a distance ϵ of p_1 is computed. If $|\text{Nbhd}(p_1)| < \text{minPts}$, then p_1 is labelled as noise. Otherwise, we label all points in $\text{Nbhd}(p_1)$ that have not already been assigned a cluster as belonging to the cluster of p_1 (even if a point has previously been labelled as noise).

Stage 2: we loop over all points of $\text{Nbhd}(p_1) \setminus \{p_1\}$, where for each point p_i we compute $\text{Nbhd}(p_i)$ and label all points of $\text{Nbhd}(p_i)$ that have yet to be assigned a cluster as belonging to the cluster of p_1. If $|\text{Nbhd}(p_i)| \geq \text{minPts}$, then all points of $\text{Nbhd}(p_i)$ are added to the set $\text{Nbhd}(p_1) \setminus \{p_1\}$. Hence, the loop over the points of the growing set $\text{Nbhd}(p_1) \setminus \{p_1\}$ finishes only when all points in the cluster of p_1 have been identified.

Stage 3: Stages 1 and 2 are repeated for a new unlabelled point p_2, and we continue so forth until all points are either assigned to a cluster or are labelled as noise.

In the experiments of Section 4.9, the parameter ϵ is optimised over a range of values, while we found it acceptable to set $\text{minPts} = 5$.

4.3 The α-Reeb Algorithm

The α-Reeb graph is a parametric version of a Reeb graph, which itself is a simplified representation of a simplicial complex formed by taking the quotient by an equivalence relation defined using level sets.

Definition 4.1 (Level Set)**.** Let $f\colon X \to \mathbb{R}$ be a real-valued function on a space X. The *level set* of f corresponding to a value $t \in \mathbb{R}$ is the set $L_t(f) = \{x \in X \mid f(x) = t\}$.

Definition 4.2 (Reeb Graph $\text{Reeb}(Q, f)$)**.** For a simplicial complex Q, let $f\colon Q \to \mathbb{R}$ be a real-valued function on Q. We define, for points $x, y \in Q$, the equivalence relation $\sim$ to be such that $x \sim y$ if and only if $f(x) = f(y)$ and x and y are in the same connected

component of the level set $L_{f(x)}(f)$. The *Reeb graph*, $\mathrm{Reeb}(Q, f)$, is the quotient space of Q formed by mapping all equivalent points under $\sim$ to a single point.

Definition 4.2 makes sense even when Q is generalised to be a topological space. If, instead of having an entire simplicial complex Q, we only have a finite set of points sampled from Q, even approximating the Reeb graph is not straightforward. Therefore, to bridge this barrier, Chazal et al. introduced the α-Reeb graph [11].

Definition 4.3 (α-Reeb Graph)**.** Let Q be a simplicial complex and $f\colon Q \to \mathbb{R}$ be a continuous real-valued function on Q. Let $\alpha > 0$ and let $\mathcal{I} = \{I_i\}$ be a covering of the range of f, where each I_i is a closed interval of length α. Consider the transitive closure of the following equivalence relation $\sim_\alpha$: for $x, y \in Q$, we define $x \sim_\alpha y$ if and only if $f(x) = f(y)$ and x and y are in the same connected component of the preimage $f^{-1}(I_i) = \{x \in Q \mid f(x) \in I_i\}$ for some interval $I_i \in \mathcal{I}$. Then the *α-Reeb graph* associated with the covering $\mathcal{I}$ of a simplicial complex Q is the quotient space formed from Q by mapping all equivalent points under $\sim_\alpha$ to a single point.

Focusing on dimension two, the α-Reeb algorithm takes as **input a connected neighbourhood graph and scale parameter** $\alpha \in \mathbb{R}_{>0}$, and **outputs an α-Reeb graph**. A neighbourhood graph $N(C; \epsilon)$ (Definition 1.17) is obtained from a point cloud C by adding edges between points that are closer to each other than the specified threshold ϵ. If such an operation yields a disconnected graph, then the algorithm can be applied individually to each connected component.

Given a connected neighbourhood graph $N(C; \epsilon)$ of a point cloud C and scale parameter α, there are three main stages to the α-Reeb algorithm:

Stage 1: a root vertex of $N(C; \epsilon)$ is chosen (for example the most distant vertex from a randomly selected one) and the function $f\colon C \to \mathbb{R}$ which assigns to each vertex its distance from the root within $N(C; \epsilon)$ is calculated. The range of the function f, $[0, \max(f)]$, is covered by the set of intervals $\mathcal{I} = \{I_i\}_{0 \le i \le m}$, where $I_i = [\frac{i\alpha}{2}, \frac{i\alpha}{2} + \alpha]$, and m is the smallest integer such that $m \ge \frac{2(\max(f) - \alpha)}{\alpha}$. These intervals are ordered, so we say that one interval is lower than another if its midpoint is smaller.

Stage 2: for each interval in the covering $\mathcal{I}$, we consider its preimage $f^{-1}(I_i) \subseteq C$. After adding an edge between two vertices in the preimage if there exists an edge between the two vertices in $N(C; \epsilon)$, we obtain a possibly disconnected subgraph of $N(C; \epsilon)$. We then build an intermediate graph G by first adding a vertex to G for each connected component of each subgraph, and then connect pairs of vertices of G by an edge if their corresponding connected components of $N(C; \epsilon)$ share a vertex.

Stage 3: for each vertex $v \in G$ related to the interval I_i, we take a copy of the interval I_i and place its midpoint at v, splitting the interval into a top and bottom half. The α-Reeb graph is the quotient of the disjoint union of these partially ordered copies of intervals, where the top half of one interval is identified to the bottom half of a higher interval if there is an edge between the corresponding vertices in G.

Similarly to the Mapper algorithm, the α-Reeb algorithm aims to connect close clusters. However the clustering of preimages of intervals in Mapper is replaced with finding connected subgraphs. In the limit $\alpha \to 0$, the α-Reeb graph tends to the Reeb graph of Definition 4.2.

In the experiments of Section 4.9, the threshold ϵ used to form the neighbourhood graph $N(C;\epsilon)$ is fixed at double the maximum birth of dots above the first widest diagonal gap of $\mathrm{PD}\{C^\alpha\}$ (see Definitions 4.7, 4.8 and 4.22). The scale parameter α that effectively determines the resolution of the output graph is optimised in the experiments.

4.4 The Homologically Persistent Skeleton $\mathrm{HoPeS}(C)$

The homologically persistent skeleton, $\mathrm{HoPeS}(C)$, first introduced in [37], was motivated by the prospect of extending reconstructed hole boundaries of unorganised clouds of edge pixels to a one-dimensional skeleton [34, 35], and has since been extended to higher dimensions [32, Definition 4.5]. In the one-dimensional setting, it seeks to extend a minimum spanning tree of a point cloud by adding critical edges to form cycles.

4.4.1 Minimum Spanning Trees and Forests of a Filtration

Definition 4.4 (Minimum Spanning Tree $\mathrm{MST}(C)$)**.** Let C be a point cloud and let $\{Q(C;\alpha)\}$ be a filtration (Definition 1.24) of complexes on C. Assigning to an edge e in the filtration a length equal to twice the minimum α such that $e \subseteq Q(C;\alpha)$, we define a *minimum spanning tree* $\mathrm{MST}(C)$ of the filtration $\{Q(C;\alpha)\}$ to be a connected graph with vertex set C whose total length of edges is the minimum possible. For any $\alpha \geq 0$, we can obtain a forest $\mathrm{MST}(C;\alpha)$ from $\mathrm{MST}(C)$ by removing all edges that are longer than 2α.

We have that $\mathrm{MST}(C;0) = C$, and for large enough α we will have that $\mathrm{MST}(C;\alpha) = \mathrm{MST}(C)$. $\mathrm{MST}(C)$ may not be unique if multiple edges enter the filtration at the same scale α, and neither is it stable under perturbations. The mergegrams in [22] can be used however to extract stable information from minimum spanning trees. Despite this lack of stability, it is always the case that $\mathrm{MST}(C;\alpha)$ enjoys the optimality of Lemma 4.6 among all spanning graphs of $Q(C;\alpha)$.

Definition 4.5 (Spanning Graph)**.** A graph G *spans* a possibly disconnected simplicial complex Q on a cloud C if G has vertex set C, is a subset of Q, and the inclusion of G into Q induces a 1-1 correspondence between connected components.

$\mathrm{MST}(C;\alpha)$ will always be a spanning graph of $Q(C;\alpha)$. This can be seen since, by definition, it has vertex set C, and is a subset of $Q(C;\alpha)$. It therefore cannot have fewer connected components than $Q(C;\alpha)$, and neither can it have more. Otherwise, this would lead to a contradiction in the minimality of $\mathrm{MST}(C)$, since there would be two components

connected by an edge of length at most 2α in $Q(C;\alpha)$ that are connected by a strictly longer edge in $\mathrm{MST}(C)$.

Lemma 4.6. *Let $\{Q(C;\alpha)\}$ be a filtration of simplicial complexes on a point cloud C. For any fixed scale $\alpha \geq 0$, the forest $\mathrm{MST}(C;\alpha)$ has the minimum total length of edges among all graphs that span $Q(C;\alpha)$.*

Proof. Let $e_1, \ldots, e_m$ be all of the edges of $\mathrm{MST}(C)$ that are longer than 2α. Hence $\mathrm{MST}(C) = \mathrm{MST}(C;\alpha) \cup e_1 \cup \ldots \cup e_m$. Now, if we assume that there exists a graph G that spans $Q(C;\alpha)$ and whose total length of edges is shorter than $\mathrm{MST}(C;\alpha)$, then we will also have that $G \cup e_1 \cup \ldots \cup e_m$ is connected with a shorter total length of edges than $\mathrm{MST}(C)$, contradicting Definition 4.4. □

Lemma 4.6 implies that, for any scale α, $\mathrm{MST}(C;\alpha)$ will be, in terms of the minimal total length of edges, optimal amongst graphs that span the complex $Q(C;\alpha)$ and thus share the same zeroth homology group.

4.4.2 Persistent Homology and its Stability Under Perturbations

The homologically persistent skeleton extends the optimality of $\mathrm{MST}(C;\alpha)$ from Lemma 4.6 to the first homology group, drawing from the field of persistent homology which summarises the evolution of homology (Definition 1.25) throughout a filtration by recording when homological features appear and disappear.

Definition 4.7 (Births and Deaths)**.** For any filtration $\{Q(C;\alpha)\}$ of complexes on a point cloud C, a homology class $\gamma \in H_k(Q(C;\alpha_i))$ is said to be *born* at $\mathrm{birth}(\gamma) = \alpha_i$ if γ is not in the full image under any of the induced homomorphisms $Q(C;\alpha) \to Q(C;\alpha_i)$ for any $\alpha < \alpha_i$. The homology class γ is said to *die* at $\mathrm{death}(\gamma) = \alpha_j$ when the image of γ under the induced homomorphism $H_k(Q(C;\alpha_i)) \to H_k(Q(C;\alpha_j))$ merges with the image of another homology class under the induced homomorphism $H_k(Q(C;\alpha)) \to H_k(Q(C;\alpha_j))$ for some $\alpha < \alpha_i$.

The birth and death pairs defined in Definition 4.7 are the key output of persistent homology, and a popular way of recording this information is in persistence diagrams.

Definition 4.8 (Persistence Diagram $\mathrm{PD}\{Q(C;\alpha)\}$)**.** Let $\{Q(C;\alpha)\}$ be a filtration of complexes on a point cloud C. The *persistence diagram* $\mathrm{PD}\{Q(C;\alpha)\} \subset \mathbb{R}^2$ is the multiset of all birth-death pairs $(\mathrm{birth}(\gamma), \mathrm{death}(\gamma))$ of all independent homology classes γ that persist in the filtration $\{Q(C;\alpha)\}$, see Figure 4.2. Since multiple independent homology classes can have the same birth and death values, there can be multiple occurrences of the same point $(\alpha_i, \alpha_j) \in \mathbb{R}^2$, and we say such points have *multiplicity* $u_{i,j}$. It is convention to include in the multiset all diagonal points (x, x) with infinite multiplicity. When the multiset is plotted in $\mathbb{R}^2$, we refer to the birth-death pairs as dots of the persistence diagram.

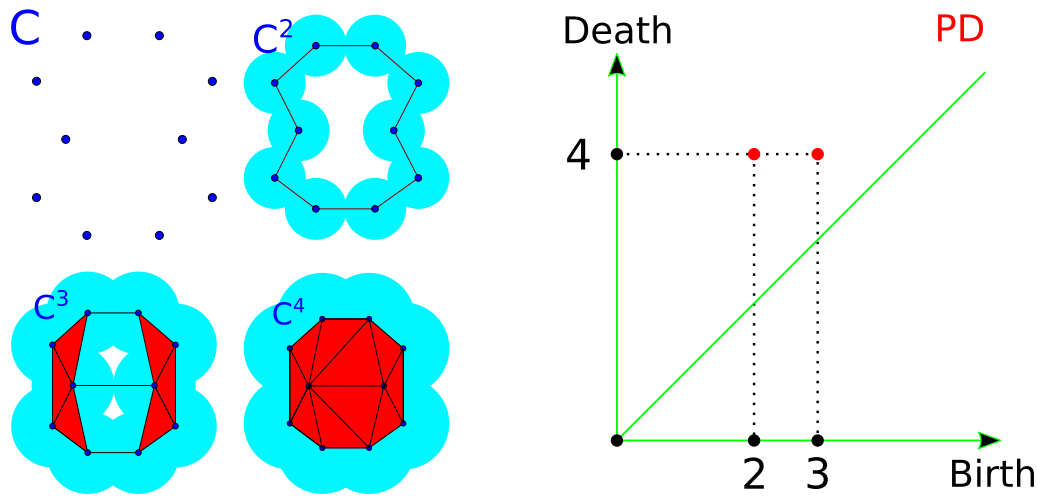

Figure 4.2: **Left:** a filtration of α-complexes of a point cloud C at $\alpha = 0, 2, 3, 4$. One cycle is born at $\alpha = 2$, and a second cycle is born at $\alpha = 3$. Both cycles persist until $\alpha = 4$, at which the cycles are filled in by 2-simplices. **Right:** the persistence diagram of the filtration, with dots at $(2, 4)$ and $(3, 4)$.

In this chapter, we consider persistence diagrams consisting of dots corresponding only to one-dimensional homology classes $\gamma \in H_1(Q(C;\alpha))$. For filtrations $\{C^\alpha\}$ of α-offsets (Definition 1.22) of a point cloud C, persistence diagrams $\mathrm{PD}\{C^\alpha\}$ are invariant under isometry transformations on C. Moreover, the key advantage of $\mathrm{PD}\{C^\alpha\}$ over other geometric invariants is its stability as outlined in Theorem 4.11, for which we need definitions of an ϵ-sample and the bottleneck distance between persistent diagrams.

Definition 4.9 (ϵ-sample)**.** A point cloud C is said to be an *ϵ-sample* of a graph G if $C \subset G^\epsilon$ and $G \subset C^\epsilon$. Hence, any point $p \in C$ is within a distance ϵ from G, and any point of the graph G is within a distance ϵ of a point $p \in C$.

Definition 4.10 (Bottleneck Distance $d_B(\mathrm{PD}, \mathrm{PD}')$)**.** The *bottleneck distance* $d_B(\mathrm{PD}, \mathrm{PD}')$ between persistence diagrams $\mathrm{PD}, \mathrm{PD}'$ is defined to be the infimum over all bijections $\psi\colon \mathrm{PD} \to \mathrm{PD}'$ of the supremum over all points $p \in \mathrm{PD}$ of the L_∞-distance $\|p - \psi(p)\|_\infty$. Namely,

$$d_B(\mathrm{PD}, \mathrm{PD}') = \inf_{\psi} \sup_{p \in \mathrm{PD}} \|p - \psi(p)\|_\infty .$$

Theorem 4.11 (Stability of Persistence)**.** *[12, simplified Theorem 5.6] Let C be any ϵ-sample of a graph G in a totally bounded metric space M. Then the persistence diagrams of Čech filtrations on G and C are ϵ-close, that is $d_B(\mathrm{PD}\{\check{\mathrm{C}}\mathrm{h}(G; M; \alpha)\}, \mathrm{PD}\{\check{\mathrm{C}}\mathrm{h}(C; M; \alpha)\}) \leq \epsilon$. This inequality also holds for the filtrations of α-offsets by Nerve Lemma 1.23. Namely, $d_B(\mathrm{PD}\{G^\alpha\}, \mathrm{PD}\{C^\alpha\}) \leq \epsilon$.*

4.4.3 HoPeS(C) is the Persistence-based Extension of MST(C)

HoPeS(C) is a skeleton of the point cloud C obtained from MST(C) by adding critical edges found using persistent homology.

Definition 4.12 (Critical Edges)**.** [32, modified from Definition 4.5] Let $\{Q(C;\alpha)\}$ be a filtration of complexes on a point cloud C. An edge e is *critical* if, upon its entry into the filtration, it gives rise to a new homology class that does not immediately die. The birth value of a critical edge e is the scale α at which the edge appears in the filtration $\{Q(C;\alpha)\}$. Defining the death value of a critical edge is less straighforward. To do this, we first define $E(\alpha) = \{e_1, \ldots, e_s\}$ to be the set of all critcal edges that have $\text{birth}(e_i) \leq \alpha$ but have not yet been assigned a death value. Then $[e_1], \ldots, [e_s]$ form a basis of $H_1(\text{MST}(C;\alpha) \cup E(\alpha)/\,\text{MST}(C;\alpha))$. Define

$$f\colon H_1(\text{MST}(C;\alpha) \cup E(\alpha)/\,\text{MST}(C;\alpha)) \to H_1(Q(C;\alpha)/\,\text{MST}(C;\alpha))$$

to be the homomorphism induced by the inclusion

$$\text{MST}(C;\alpha) \cup E(\alpha)/\,\text{MST}(C;\alpha) \to Q(C;\alpha)/\,\text{MST}(C;\alpha).$$

Let $\{b_1, \ldots, b_r\}$ be a basis of $\ker(f)$, where $r = \dim(\ker(f))$ is equal to the number of critical edges that die at exactly α (so $r \leq s$). We can expand each basis element as $b_i = \sum_{j=1}^{s} c_{i,j} e_j$, with $c_{i,j} \in \mathbb{Z}_2$, and consider (in $\mathbb{Z}_2$) the system of equations $\sum_{j=1}^{s} c_{i,j} x_j$ for $1 \leq i \leq r$. Since basis elements are linearly independent, this system of equations can be solved with r leading variables expressed in terms of the remaining $s - r$ free variables. Letting $I \subseteq \{1, \ldots, s\}$ be the set of all the indices of the leading variables, we set the death value of the critical edge e_i to be α if and only if $i \in I$.

The *elder rule* selects those leading variables whose corresponding set of critical edges has the greatest combined birth value (and so are younger), allowing older classes to persist for longer. If there are distinct sets with the greatest combined birth value, we have a genuine choice, although this choice does not affect the theoretical guarantees of HoPeS in the subsequent sections. We are now ready to define the homologically persistent skeleton.

Definition 4.13 (HoPeS(C) and Reduced Skeletons HoPeS($C;\alpha$))**.** Let $\{Q(C;\alpha)\}$ be a filtration of complexes on a point cloud C. The *homologically persistent skeleton* HoPeS(C) is the union of a minimum spanning tree MST(C) with all critical edges e with labels $(\text{birth}(e), \text{death}(e))$, see Figure 4.3. For any scale $\alpha \geq 0$, the *reduced skeleton* HoPeS($C;\alpha$) is obtained from HoPeS(C) by removing all edges longer than 2α and all critical edges e with $\text{death}(e) \leq \alpha$.

Although HoPeS(C) is dependent of the filtration $\{Q(C;\alpha)\}$ of complexes on the point cloud C, for simplicity we omit this dependence on the filtration in our notation.

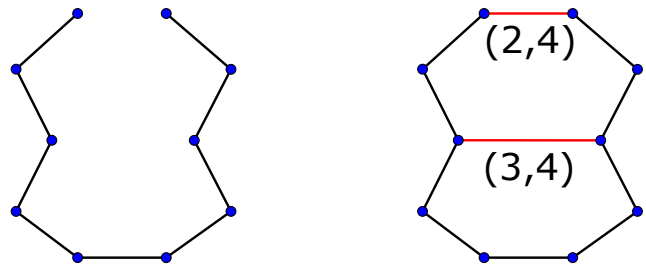

Figure 4.3: **Left:** a minimum spanning tree $\mathrm{MST}(C)$ of the point cloud C from Figure 4.2. **Right:** $\mathrm{HoPeS}(C)$, where the two critical edges are in red and are labelled with birth and death values $(\mathrm{birth}(e), \mathrm{death}(e))$.

4.5 Optimality Guarantees of Reduced Skeletons $\mathrm{HoPeS}(C;\alpha)$

Theorem 4.21 proves that, for any scale $\alpha \geq 0$, the reduced skeleton $\mathrm{HoPeS}(C;\alpha)$ is optimal in terms of minimising the total length of edges amongst graphs that span C and have the same zeroth and first homology groups as the complex $Q(C;\alpha)$. To prove Theorem 4.21, there are several statements that we should first introduce.

Lemma 4.14. *Let a homology class $\gamma \in H_1(Q(C;\alpha))$ be born due to a critical edge e added to the filtration $\{Q(C;\alpha)\}$ at scale α. Then e has length $|e| = 2 \cdot \mathrm{birth}(\gamma)$ relative to the filtration $\{Q(C;\alpha)\}$.*

Proof. By Definition 4.12, a critical edge e is the last edge added to a cycle $L \subseteq Q(C;\alpha)$, thus giving birth to the homology class γ at scale $\alpha = \mathrm{birth}(\gamma)$. Since the length $|e|$ equals the doubled scale 2α, we have that $|e| = 2 \cdot \mathrm{birth}(\gamma)$. □

Lemma 4.15. *The reduced skeleton $\mathrm{HoPeS}(C;\alpha)$ is a subgraph of the simplicial complex $Q(C;\alpha)$ for any α.*

Proof. By Definition 4.13, $\mathrm{HoPeS}(C;\alpha)$ consists of the forest $\mathrm{MST}(C;\alpha)$ and all critical edges e satisfying the two constraints $|e| \leq 2\alpha$ and $\mathrm{death}(e) > \alpha$. Lemma 4.6 implies that $\mathrm{MST}(C;\alpha) \subseteq Q(C;\alpha)$, and any critical edge e belongs to $Q(C;\alpha)$ for $|e| \leq 2\alpha$. Hence $\mathrm{HoPeS}(C;\alpha) \subseteq Q(C;\alpha)$. □

Lemma 4.15 implies that the inclusion $\mathrm{HoPeS}(C;\alpha) \to Q(C;\alpha)$ induces a homomorphism f_* on homology groups. Lemmas 4.16 and 4.17 analyse what happens with f_* when a critical edge e is added to or deleted from $\mathrm{HoPeS}(C;\alpha)$.

Lemma 4.16 (Addition of a critical edge)**.** *Let an inclusion $f\colon G \to Q$ of a graph G into a complex Q induce an isomorphism $f_*\colon H_1(G) \to H_1(Q)$. Now, between some vertices $u, v \in G$ let us add an edge e to G and Q that creates a homology class $\gamma \in H_1(Q \cup e)$. Then f_* extends to an isomorphism $H_1(G \cup e) \to H_1(Q \cup e)$.*

Proof. Let $L \subseteq G \cup e$ be a cycle containing the added edge e, and note that $H_1(G \cup e) \cong H_1(G) \oplus \langle [L] \rangle$. Now f extends to the inclusion $G \cup e \to Q \cup e$, and $f(L)$ is a cycle in $Q \cup e$. Hence $H_1(Q \cup e) \cong H_1(Q) \oplus \langle [f(L)] \rangle$. By mapping $[L]$ to $[f(L)]$, we can extend f_* to the required isomorphism $H_1(G) \oplus \langle [L] \rangle \to H_1(Q) \oplus \langle [f(L)] \rangle$. □

Lemma 4.17 (Deletion of a critical edge)**.** *Let an inclusion $f\colon G \to Q$ of a graph G into a complex Q induce an isomorphism $f_*\colon H_1(G) \to H_1(Q)$. Let a homology class $\gamma \in H_1(Q)$ die after adding the 2-simplex T to Q, and let e be an open edge of the 2-simplex T. Then f_* descends to an isomorphism $f_*\colon H_1(G - e) \to H_1(Q \cup T)$.*

Proof. Adding T to Q kills the one-dimensional homology class $\partial T \subseteq Q$. Then $H_1(Q \cup T) \cong H_1(Q)/\langle [\partial T] \rangle$. Deleting an open edge e from the boundary ∂T kills the one-dimensional homology class $\partial T \subseteq G$, so $H_1(G - e) \cong H_1(G)/\langle [\partial T] \rangle$. Therefore, f_* can descend to the required isomorphism $f_*\colon H_1(G - e) \to H_1(Q \cup T)$. □

Lemmas 4.16 and 4.17 can be used to show that the first homology groups of $\mathrm{HoPeS}(C;\alpha)$ and $Q(C;\alpha)$ are isomorphic.

Proposition 4.18. *The inclusion $\mathrm{HoPeS}(C;\alpha) \to Q(C;\alpha)$ from Lemma 4.15 induces an isomorphism of one-dimensional homology groups $H_1(\mathrm{HoPeS}(C;\alpha)) \to H_1(Q(C;\alpha))$.*

Proof. Both $\mathrm{HoPeS}(C;0)$ and $Q(C;0)$ coincide exactly with the point cloud C, and so their one-dimensional homologies are trivial. Each time a homology class is born or dies in $H_1(Q(C;\alpha))$, the isomorphism $H_1(\mathrm{HoPeS}(C;\alpha)) \to H_1(Q(C;\alpha))$ is preserved by Lemmas 4.16 and 4.17. □

We have shown in Lemma 4.6 and Proposition 4.18 that the zeroth and first homology groups of $\mathrm{HoPeS}(C;\alpha)$ agree with those of the simplicial complex $Q(C;\alpha)$ in the filtration $\{Q(C;\alpha)\}$ from which $\mathrm{HoPeS}(C;\alpha)$ is derived. Moreover, it can be shown that $\mathrm{HoPeS}(C;\alpha)$ is optimal amongst all such graphs in terms of minimising the total length of edges.

Lemma 4.19. *Let $\{Q(C;\alpha)\}$ be a filtration and let $L \subseteq Q(C;\alpha)$ be a cycle that represents a homology class $\gamma \in H_1(Q(C;\alpha))$. Then any longest edge $e \subset L$ has length $|e| \geq 2 \cdot \mathrm{birth}(\gamma)$.*

Proof. Assume to the contrary that a longest edge e of the cycle L has half-length $0.5 \cdot |e| < \mathrm{birth}(\gamma)$. Then L enters the filtration $\{Q(C;\alpha)\}$ earlier than $\alpha = \mathrm{birth}(\gamma)$ and so cannot represent the homology class γ that starts living only from $\mathrm{birth}(\gamma)$. □

Recall that a forest $\mathrm{MST}(C;\alpha)$ on a point cloud C at scale α is obtained from a minimum spanning tree $\mathrm{MST}(C)$ by removing all open edges that are longer than the doubled scale 2α.

Proposition 4.20. *Let a graph $G \subseteq Q(C;\alpha)$ span $Q(C;\alpha)$ and let $H_1(G) \to H_1(Q(C;\alpha))$ be an isomorphism induced by the inclusion. Let (b_i, d_i), $1 \leq i \leq m$, be all of the dots in*

the persistence diagram $\mathrm{PD}\{Q(C;\alpha)\}$ *such that* $b_i \leq \alpha < d_i$ *(these dots correspond to all homology classes that are alive at scale* α*). Then the total length of* G *is bounded below by the sum of the total length of edges in the forest* $\mathrm{MST}(C;\alpha)$ *plus the summation* $2\sum_{i=1}^{m} b_i$*.*

Proof. Let the subgraph $G_1 \subseteq G$ consist of all non-bridge edges (Definition 1.18) of G and let $e_1 \subset G_1$ be a longest open edge of G_1. Removing e_1 from G makes $H_1(G)$ smaller. Hence there is a cycle $L \subseteq G_1$ containing e_1 that represents a homology class $\gamma_1 \in H_1(Q(C;\alpha))$ which corresponds to some off-diagonal dot of $\mathrm{PD}\{Q(C;\alpha)\}$, say (b_1, d_1). Then we have that γ_1 lives over the interval $b_1 = \mathrm{birth}(\gamma_1) \leq \alpha < \mathrm{death}(\gamma_1) = d_1$, and Lemma 4.19 implies that $|e_1| \geq 2b_1$. Similarly, let the graph $G_2 \subseteq G - e_1$ consist of all non-bridge edges of $G - e_1$ and let e_2 be a longest open edge of G_2. Then, as before, we find its corresponding dot, say (b_2, d_2), and deduce that $|e_2| \geq 2b_2$. Repeating this process until we have a forest, we can conclude that $\sum_{i=1}^{m} |e_i| \geq 2\sum_{i=1}^{m} b_i$. The graph $G - (e_1 \cup \ldots \cup e_m)$ still spans the possibly disconnected complex $Q(C;\alpha)$ since we removed each time a non-bridge edge, so the total length of $G - (e_1 \cup \ldots \cup e_m)$ is at least the total length of $\mathrm{MST}(C;\alpha)$ by Lemma 4.6, and the result follows. □

Proposition 4.20 gives a lower bound for the total length of edges of a graph G that spans $Q(C;\alpha)$ and agrees with $Q(C;\alpha)$ on the first homology group. Theorem 4.21 states that $\mathrm{HoPeS}(C;\alpha)$ achieves this lower bound.

Theorem 4.21 (Optimality of $\mathrm{HoPeS}(C;\alpha)$)**.** *Let* $\{Q(C;\alpha)\}$ *be a filtration of complexes on a point cloud* C*. For any* $\alpha \geq 0$*, the reduced skeleton* $\mathrm{HoPeS}(C;\alpha)$ *has the minimum total length over all graphs* $G \subseteq Q(C;\alpha)$ *that span* $Q(C;\alpha)$ *and induce an isomorphism* $H_1(G) \to H_1(Q(C;\alpha))$ *under inclusion.*

Proof. For any $\alpha \geq 0$, the inclusion $\mathrm{HoPeS}(C;\alpha) \to Q(C;\alpha)$ induces an isomorphism on one-dimensional homology groups by Proposition 4.18. Let $\gamma_1, \ldots, \gamma_m$ represent all m dots of the persistence diagram $\mathrm{PD}\{Q(C;\alpha)\}$ counted with multiplicities that have $\mathrm{birth}(\gamma_i) \leq \alpha < \mathrm{death}(\gamma_i)$. Then $\gamma_1, \ldots, \gamma_m$ form a basis of $H_1(Q(C;\alpha)) \cong H_1(\mathrm{HoPeS}(C;\alpha))$ by Definition 4.8. The total length of $\mathrm{HoPeS}(C;\alpha)$ equals the total length of $\mathrm{MST}(C;\alpha)$ plus $2 \cdot \sum_{i=1}^{m} \mathrm{birth}(\gamma_i)$ by Lemma 4.14, which is exactly the lower bound for any graph $G \subseteq Q(C;\alpha)$ that spans $Q(C;\alpha)$ and has the same one-dimensional homology group as $Q(C;\alpha)$ as stated in Proposition 4.20. □

4.6 Guarantees for Reconstructions using Derived Skeletons

A persistence diagram captures all homological features regardless of how long they persist in the filtration. It is often the case that we want to distinguish between noisy features that persist for a short time and genuine, dominant features that persist for much longer. As such, in this section we introduce important subdiagrams of the persistence diagram that seek to separate noisy features from those that are genuine, and it is by using these

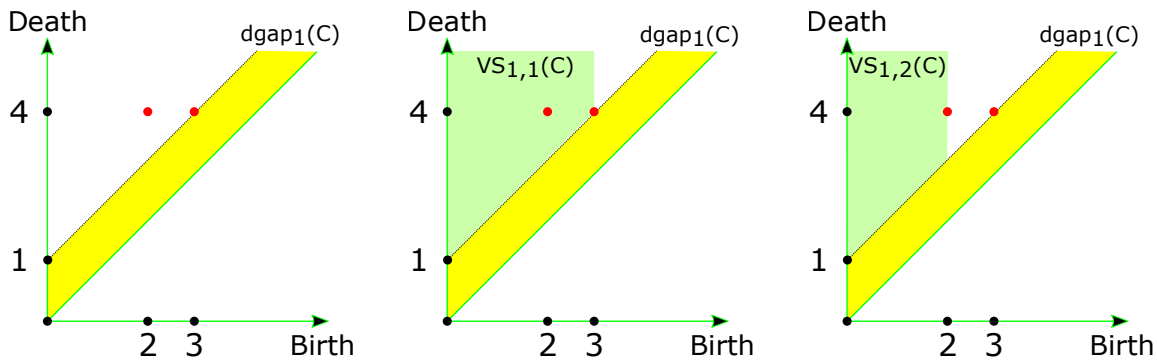

Figure 4.4: Subdiagrams of the persistence diagram of the point cloud C in Figure 4.2. **Left:** Both dots are above the (yellow) first widest diagonal gap $\mathrm{dgap}_1(C) = \{0 < y - x < 1\}$. In this case, $\mathrm{ds}_1(C) = 1$. **Middle:** Both dots are in the green region and so belong to $\mathrm{VS}_{1,1}(C)$. We have that $\mathrm{vs}_{1,1}(C) = 3$. **Right:** $\mathrm{vs}_{1,2}(C) = 2$, and so only the leftmost dot is in $\mathrm{VS}_{1,2}(C)$.

subdiagrams that we obtain a family of subskeletons of $\mathrm{HoPeS}(C)$ for which there exists reconstruction guarantees.

Definition 4.22 (Diagonal Gap $\mathrm{dgap}_k(C)$, Subdiagram $\mathrm{DS}_k(C)$, Scale $\mathrm{ds}_k(C)$)**.** For any point cloud C, a *diagonal gap* of $\mathrm{PD}\{C^\alpha\}$ is a strip $\{0 \leq a < y - x < b\}$ that has dots of $\mathrm{PD}\{C^\alpha\}$ in both boundary lines $\{y - x = a\}$ and $\{y - x = b\}$ but not in the interior of the strip. For any $k \geq 1$, the *k-th widest diagonal gap* $\mathrm{dgap}_k(C)$ has the k-th largest vertical width $|\mathrm{dgap}_k(C)| = b - a$. The *diagonal subdiagram* $\mathrm{DS}_k(C) \subset \mathrm{PD}\{C^\alpha\}$ consists of only the dots above the lowest of the first k widest diagonal gaps $\mathrm{dgap}_i(C)$, $1 \leq i \leq k$. Each $\mathrm{DS}_k(C)$ is bounded below by the higher diagonal line $y - x = b$ of the lowest of the first k widest diagonal gaps, and so we say that it has *diagonal scale* $\mathrm{ds}_k(C) = b$.

Definition 4.23 (Vertical Gap $\mathrm{vgap}_{k,l}(C)$, Subdiagram $\mathrm{VS}_{k,l}(C)$, Scale $\mathrm{vs}_{k,l}(C)$)**.** In the diagonal subdiagram $\mathrm{DS}_k(C)$ from Definition 4.22, we define a *vertical gap* to be the widest vertical strip $\{0 \leq a < x < b\}$ such that the boundary line $\{x = a\}$ contains a dot of $\mathrm{DS}_k(C)$, and there is no dot in the interior of the strip. For $l \geq 1$, the l-th widest vertical gap $\mathrm{vgap}_{k,l}(C)$ has the l-th widest horizontal width $\left|\mathrm{vgap}_{k,l}(C)\right| = b - a$. The *vertical subdiagram* $\mathrm{VS}_{k,l}(C) \subseteq \mathrm{DS}_k(C)$ consists of only the dots of $\mathrm{DS}_k(C)$ that lie to the left of the leftmost of the first l widest vertical gaps $\mathrm{vgap}_{k,i}(C)$, $1 \leq i \leq l$. Each $\mathrm{VS}_{k,l}(C)$ is bounded on the right by the left boundary line $x = a$ of the leftmost of the first l widest vertical gaps, and so we say that it has *vertical scale* $\mathrm{vs}_{k,l}(C) = a$.

In Definitions 4.22 and 4.23, if there are different diagonal gaps or vertical gaps with the same width, we split the tie by saying that the lowest or the leftmost gap has the larger width respectively. Example subdiagrams can be seen in Figure 4.4.

Definition 4.24 (Derived Skeletons $\mathrm{HoPeS}_{k,l}(C)$)**.** Let $\mathrm{HoPeS}(C)$ be obtained from a filtration $\{Q(C;\alpha)\}$ on a point cloud C. For intergers $k, l \geq 1$, the *derived skeleton* $\mathrm{HoPeS}_{k,l}(C)$ is obtained from $\mathrm{HoPeS}(C)$ by removing all edges that are longer than $2\,\mathrm{vs}_{k,l}(C)$ and by keeping only the critical edges that correspond to dots of $\mathrm{VS}_{k,l}(C)$ that do not die until after $\alpha = \mathrm{vs}_{k,l}(C)$.

Lemma 4.25. *The derived skeleton* $\mathrm{HoPeS}_{k,l}(C)$ *is within* $\mathrm{HoPeS}(C;\mathrm{vs}_{k,l}(C))$.

Proof. By Definition 4.13, $\mathrm{HoPeS}(C;\mathrm{vs}_{k,l}(C))$ is obtained from $\mathrm{HoPeS}(C)$ by removing all edges that are longer that $2\,\mathrm{vs}_{k,l}(C)$, in addition to removing critical edges whose death value is not greater than $\mathrm{vs}_{k,l}(C)$. Definition 4.24 imposes these same constraints on the derived skeleton $\mathrm{HoPeS}_{k,l}(C)$, with the additional constraint that all critical edges must correspond to dots of $\mathrm{VS}_{k,l}(C)$. Hence $\mathrm{HoPeS}_{k,l}(C) \subseteq \mathrm{HoPeS}(C;\mathrm{vs}_{k,l}(C))$. □

Perhaps the most significant derived skeleton is $\mathrm{HoPeS}_{1,1}(C)$ which is often the most effective at separating noisy and genuine homological features. It is for this particular skeleton that, under certain conditions, Theorem 4.28 provides guarantees on the reconstruction of a graph G from an ϵ-sample of G. One of these conditions involves the thickness $\theta(G)$ of a graph G.

Definition 4.26 (Radius ρ of a Cycle, Thickness $\theta(G)$ of a Graph)**.** For a graph G in a metric space, the *radius* ρ of a non-self-intersecting cycle $L \subseteq G^\alpha$ is the persistence $\mathrm{death}(\gamma) - \mathrm{birth}(\gamma)$ of its corresponding homology class γ in the filtration. Not all homology classes are necessarily born at $\alpha = 0$, and so we define the *thickness* $\theta(G)$ of G to be the maximum persistence of any homology class born after $\alpha = 0$.

Lemma 4.27. *For a graph G in a metric space with* $\mathrm{rank}(H_1(G)) = m$*, the one-dimensional persistence diagram* $\mathrm{PD}\{G^\alpha\}$ *has exactly m dots (with multiplicities) on the vertical axis (not including the conventional infinitely many dots at the origin).*

Proof. By Definition 4.7, a dot of the persistence diagram $\mathrm{PD}\{G^\alpha\}$ is of the form $(0, d_i)$, $d_i > 0$, if and only if it corresponds to a homology class that is born an $\alpha = 0$ and hence is present in $H_1(G^0) = H_1(G)$. □

Theorem 4.28. *Let C be any ϵ-sample of a connected graph G in a metric space such that* $\mathrm{rank}(G) = m$*. Let the m dots on the vertical axis (Lemma 4.27) of* $\mathrm{PD}\{G^\alpha\}$ *have ordered deaths $y_1 \leq \ldots \leq y_m$. If $y_1 > 7\epsilon + 2\theta(G) + \max_{1 \leq i \leq m-1}\{y_{i+1} - y_i\}$, then* $\mathrm{vs}_{1,1}(C)$ *is a lower bound for the noise ϵ. Moreover, the derived skeleton* $\mathrm{HoPeS}_{1,1}(C)$ *is contained within the 2ϵ-offset of G and has the same first homology group, namely* $\mathrm{HoPeS}_{1,1}(C) \subset G^{2\epsilon}$ *and* $H_1(\mathrm{HoPeS}_{1,1}(C)) \cong H_1(G)$*. (For an example of a graph satisfying the condition, see Figure 4.5.)*

Proof. Besides the m dots on the vertical axis of $\mathrm{PD}\{G^\alpha\}$, all other dots correspond to homology classes born after $\alpha = 0$ and thus by Definition 4.26 cannot lie above the line $y - x = \theta(G)$. The stated inequality $y_1 > 7\epsilon + 2\theta(G) + \max_{1\le i\le m-1}\{y_{i+1} - y_i\}$ guarantees that the diagonal gap $\{\theta(G) < y - x < y_1\}$ is wider than any other diagonal gaps of $\mathrm{PD}\{G^\alpha\}$.

By Stability Theorem 4.11, the perturbed diagram $\mathrm{PD}\{C^\alpha\}$ is in the ϵ-offset of $\mathrm{PD}\{G^\alpha\}$ with respect to the L_∞-metric on $\mathbb{R}^2$. Therefore, all noisy dots near the diagonal cannot be above the diagonal line $y - x = \theta(G) + 2\epsilon$, whilst the remaining dots cannot be lower than the line $y - x = y_1 - 2\epsilon$. So the diagonal strip $\{\theta(G) + 2\epsilon < y - x < y_1 - 2\epsilon\}$ of vertical width $y_1 - \theta(G) - 4\epsilon$ is empty in $\mathrm{PD}\{C^\alpha\}$. We will show that this is the widest diagonal gap in $\mathrm{PD}\{C^\alpha\}$.

Again, by Stability Theorem 4.11, any dot $(0, y_i) \in \mathrm{PD}\{G^\alpha\}$ cannot move outside of the diagonal strip $\{y_i - 2\epsilon \le y - x \le y_i + \epsilon\}$ (where the asymmetry here is due to the fact that the dot cannot move in the negative x-direction). Therefore, in $\mathrm{PD}\{C^\alpha\}$, the widest diagonal gap between these perturbed dots cannot have a vertical width of more than $\max_{1\le i\le m-1}\{y_{i+1} - y_i\} + 3\epsilon$, whilst as mentioned the greatest diagonal gap between dots near the diagonal cannot have a vertical width of more than $\theta(G) + 2\epsilon$. Hence, the given inequality $y_1 - \theta(G) - 4\epsilon > \theta(G) + \max_{1\le i\le m-1}\{y_{i+1} - y_i\} + 3\epsilon$ implies that all other diagonal gaps have a vertical width smaller than $y_1 - \theta(G) - 4\epsilon$.

Hence the widest diagonal gap of $\mathrm{PD}\{C^\alpha\}$ covers the diagonal strip $\{\theta(G) + 2\epsilon < y - x < y_1 - 2\epsilon\}$ which is within the widest diagonal gap of $\mathrm{PD}\{G^\alpha\}$. Then the diagonal subdiagram $\mathrm{DS}_1(C)$ contains only dots above the diagonal line $y - x = y_1 - 2\epsilon$, which are exactly the m perturbations of the original dots $(0, y_i)$, all of which lie in the vertical strip $\{0 \le x \le \epsilon\}$. Hence, we have that the vertical scale $\mathrm{vs}_{1,1}(C) \le \epsilon$, and the derived skeleton $\mathrm{HoPeS}_{1,1}(C)$ contains exactly m critical edges corresponding to the m dots of $\mathrm{DS}_1(C)$.

It remains to prove that $\mathrm{HoPeS}_{1,1}(C)$ is 2ϵ-close to G. We have that $\mathrm{MST}(C) = \mathrm{MST}(C;\epsilon)$, or in other words all edges of $\mathrm{MST}(C)$ are no longer than 2ϵ. Otherwise, C^ϵ would have multiple connected components which would imply that C could not be an ϵ-sample of G as G is connected. Since all dots of $\mathrm{DS}_1(C)$ lie within the vertical strip $\{0 \le x \le \epsilon\}$, all critical edges e of $\mathrm{HoPeS}_{1,1}(C)$ have length $|e| \le 2\epsilon$ too. Therefore, all edges of $\mathrm{HoPeS}_{1,1}(C)$ have lengths at most 2ϵ, and as C is an ϵ-sample of G, we can deduce that $\mathrm{HoPeS}_{1,1}(C) \subset C^\epsilon \subset G^{2\epsilon}$. □

In Lemmas 4.29 and 4.30 we extend Stability Theorem 4.11 to diagonal and vertical subdiagrams. By doing so, we are able to state and prove Theorem 4.32 that generalises Theorem 4.28.

Lemma 4.29. *Let C be any ϵ-sample of a connected graph G in a metric space. If $|\mathrm{dgap}_k(G)| - |\mathrm{dgap}_{k+1}(G)| > 8\epsilon$, then there is a bijection $\psi\colon \mathrm{DS}_k(G) \to \mathrm{DS}_k(C)$ such that $\|q - \psi(q)\|_\infty \le \epsilon$ for all $q \in \mathrm{DS}_k(G)$.*

Proof. By Stability Theorem 4.11, there is a bijection $\psi\colon \mathrm{PD}\{G^\alpha\} \to \mathrm{PD}\{C^\alpha\}$ such that

$\|q - \psi(q)\|_\infty \leq \epsilon$ for all $q \in \mathrm{PD}\{G^\alpha\}$. We will show that with the given inequality, ψ descends to a bijection between the k-th diagonal subdiagrams.

The ϵ-neighbourhood of a dot $q = (u, v)$ using the L_∞-distance is the square $[u-\epsilon, u+\epsilon] \times [v-\epsilon, v+\epsilon]$. This square is bounded by the diagonal strip $\{v-u-2\epsilon < y-x < v-u+2\epsilon\}$. Hence any diagonal gap $\{a < y - x < b\}$ in $\mathrm{PD}\{G^\alpha\}$ can become thinner or wider by at most 4ϵ in $\mathrm{PD}\{C^\alpha\}$.

By the given inequality, the first k widest diagonal gaps $\mathrm{dgap}_i(G)$, $1 \leq i \leq k$, in $\mathrm{PD}\{G^\alpha\}$ are wider by more than 8ϵ than all other $\mathrm{dgap}_i(G)$, $i > k$, which is a sufficiently advantageous width such that none of the $\mathrm{dgap}_i(G)$, $i > k$, can become wider than the first k widest gaps in the perturbed persistence diagram $\mathrm{PD}\{C^\alpha\}$.

Although the order of the first k widest diagonal gaps may not be preserved under the bijection ψ, the lowest of these diagonal gaps, $\{a < y - x < b\}$, is preserved by ψ. To show this, consider the thinner strip $S = \{a+2\epsilon < y-x < b-2\epsilon\}$ which has no dots from $\mathrm{PD}\{C^\alpha\}$ within it and has vertical width $|S| \geq |\mathrm{dgap}_k(G)| - 4\epsilon \geq |\mathrm{dgap}_{k+1}(G)| + 4\epsilon \geq |\mathrm{dgap}_{k+1}(C)|$. So the strip S is wider than $\mathrm{dgap}_{k+1}(C)$ and so must be contained by one of the first k widest diagonal gaps of $\mathrm{PD}\{C^\alpha\}$. As none of the lower diagonal gaps of $\mathrm{PD}\{C^\alpha\}$ are among the k widest, S is contained in the lowest of the first k widest diagonal gaps. Hence all dots above S remain above S under the bijection ψ, and by Definition 4.22 these are exactly the dots of $\mathrm{PD}\{G^\alpha\}$ and $\mathrm{PD}\{C^\alpha\}$ that form the diagonal subdiagrams $\mathrm{DS}_k(G)$ and $\mathrm{DS}_k(C)$ respectively. Hence ψ descends to a bijection between the k-th diagonal subdiagrams. □

Lemma 4.30. *Let $\psi\colon \mathrm{DS}_k(G) \to \mathrm{DS}_k(C)$ be a bijection such that $\|q - \psi(q)\|_\infty \leq \epsilon$ holds for all $q \in \mathrm{DS}_k(G)$, as in Lemma 4.29. If $|\mathrm{vgap}_{k,l}(G)| - |\mathrm{vgap}_{k,l+1}(G)| > 4\epsilon$, then ψ descends to a bijection $\psi\colon \mathrm{VS}_{k,l}(G) \to \mathrm{VS}_{k,l}(C)$.*

Proof. We follow a similar method as used in the proof of Lemma 4.29. The x-coordinate of any dot $q \in \mathrm{DS}_k(G)$ changes under the given bijection by at most ϵ, and so each vertical gap can become thinner or wider by at most 2ϵ. By the given inequality, the first l widest vertical gaps $\mathrm{vgap}_{k,i}(G)$, $1 \leq i \leq l$, are wider by more than 4ϵ than all other $\mathrm{vgap}_{k,i}(G)$, $i > l$, which is a sufficiently advantageous width such that none of the $\mathrm{vgap}_{k,i}(G)$, $i > l$, can become wider than the first l widest vertical gaps in the perturbed k-th diagonal subdiagram $\mathrm{DS}_k(C)$.

Again, although the order of the first l widest vertical gaps may change under the bijection ψ, the leftmost of these vertical gaps, $\{a < x < b\}$, is preserved by ψ. To show this, consider the thinner strip $S = \{a + \epsilon < x < b - \epsilon\}$ which has no dots from $\mathrm{DS}_k(C)$ within it and has vertical width $|S| \geq |\mathrm{vgap}_{k,l}(G)| - 2\epsilon \geq |\mathrm{vgap}_{k,l+1}(G)| + 2\epsilon \geq |\mathrm{vgap}_{k,l+1}(C)|$. So the strip S is wider than $\mathrm{vgap}_{k,l+1}(C)$ and so must be contained by one of the first k widest vertical gaps of $\mathrm{DS}_k(C)$. As none of the further left vertical gaps of $\mathrm{PD}\{C^\alpha\}$ are among the k widest, S is contained in the leftmost of the first k widest vertical gaps. Hence all dots to the left of S remain to the left of S under the bijection ψ, and by Definition 4.23 these are exactly the dots of $\mathrm{DS}_k(G)$ and $\mathrm{DS}_k(C)$ that form the vertical

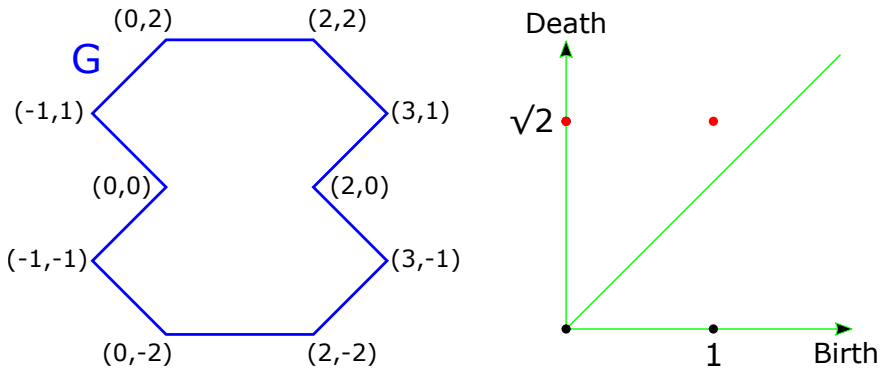

Figure 4.5: An example of a graph G satisfying the conditions of Theorem 4.32 for $k = 1$, $l = 2$. The persistent diagram of the graph has two dots with coordinates $(0, \sqrt{2})$ and $(1, \sqrt{2})$. Consequently, we have that $\mathrm{ds}_1(G) = \sqrt{2}$, $|\mathrm{dgap}_1(G)| = 1$, $|\mathrm{dgap}_2(G)| = \sqrt{2} - 1$, $\mathrm{vs}_{1,2}(G) = 0$, $|\mathrm{vgap}_{1,2}(G)| = 1$ and $|\mathrm{vgap}_{1,3}(G)| = 0$. Therefore, for $\epsilon < \frac{2-\sqrt{2}}{8}$, G satisfies the conditions of Theorem 4.32 for $k = 1$, $l = 2$. We also note that the thickness of G, $\theta(G) = \sqrt{2} - 1$, and so for $\epsilon < \frac{2-\sqrt{2}}{7}$, G also satisfies the condition of Theorem 4.28.

subdiagrams $\mathrm{VS}_{k,l}(G)$ and $\mathrm{VS}_{k,l}(C)$ respectively. Hence ψ descends to a bijection between these vertical subdiagrams. □

The geometric approximation guarantee of derived skeletons in Theorem 4.32 requires the following lemma about reduced skeletons.

Lemma 4.31 (Approximation by Reduced $\mathrm{HoPeS}(C;\alpha)$)**.** *Let C be a finite ϵ-sample of a subspace G of a metric space. Then the reduced skeleton $\mathrm{HoPeS}(C;\alpha)$ is contained within the $(\epsilon+\alpha)$-offset $G^{\epsilon+\alpha}$ for any scale $\alpha \geq 0$.*

Proof. Any edge $e \subset \mathrm{HoPeS}(C;\alpha)$ has length at most 2α by Definition 4.13, and hence is covered by the balls with radius α and centres at the endpoints of e. Therefore, we have that $\mathrm{HoPeS}(C;\alpha) \subset C^{\alpha} \subset G^{\epsilon+\alpha}$ since C is an ϵ-sample of G. □

Theorem 4.32 (Reconstruction by Derived Skeletons)**.** *Let C be an ϵ-sample of an unknown graph G in a metric space M, where G satisfies the following conditions for some $k, l \geq 1$:*

I Any cycle $L \subseteq G$ corresponds to a homology class γ such that $\mathrm{death}(\gamma) \geq \mathrm{ds}_k(G)$.

II $|\mathrm{dgap}_k(G)| - |\mathrm{dgap}_{k+1}(G)| > 8\epsilon$.

III $\mathrm{vs}_{k,l}(G) = 0$.

IV $|\mathrm{vgap}_{k,l}(G)| - |\mathrm{vgap}_{k,l+1}(G)| > 4\epsilon$.

Then the noise ϵ is bounded below by $\mathrm{vs}_{k,l}(C)$, and the derived skeleton $\mathrm{HoPeS}_{k,l}(C)$ is contained within the 2ϵ-offset of the underlying graph G and has the same first homology group as G. (For an example of a graph satisfying the conditions, see Figure 4.5.)

Proof. Due to Condition II, Lemma 4.29 implies that there is a bijection $\psi\colon \mathrm{DS}_k(G) \to \mathrm{DS}_k(C)$ such that $\|q - \psi(q)\|_\infty \leq \epsilon$ for all $q \in \mathrm{DS}_k(G)$. Similarly, due to Condition IV, Lemma 4.30 implies that ψ descends to a bijection $\psi\colon \mathrm{VS}_{k,l}(G) \to \mathrm{VS}_{k,l}(C)$.

Let $L_1, \ldots, L_m \subset G$ be all m independent cycles of G, corresponding to the m homology classes $\gamma_1, \ldots, \gamma_m$ that generate the first homology group $H_1(G)$. In the filtration $\{G^\alpha\}$, these homology classes persist from $\alpha = 0$ until they die at $\alpha = \mathrm{death}(\gamma_i)$, and are exactly the dots of $\mathrm{PD}\{G^\alpha\}$ that lie on the vertical death axis. Condition I implies that all these dots $(0, \mathrm{death}(\gamma_i))$ belong to $\mathrm{DS}_k(G)$.

We can deduce from Condition III that the leftmost of the first l widest vertical gaps is attached to the vertical death axis in $\mathrm{PD}\{G^\alpha\}$, implying that $\mathrm{VS}_{k,l}(G)$ consists of exactly the m dots $(0, \mathrm{death}(\gamma_i))$ with birth value 0.

Due to the bijection ψ, $\mathrm{VS}_{k,l}(C)$ has exactly m dots, which are the noisy images of the dots $(0, \mathrm{death}(\gamma_i)) \in \mathrm{VS}_k(G)$. Due to the property of the bijection, these dots can be at most a distance ϵ from the vertical death axis, implying that $\mathrm{vs}_{k,l}(C) \leq \epsilon$.

The minimum death of the dots of $\mathrm{VS}_{k,l}(C)$ is bounded below by $\mathrm{ds}_k(G) - \epsilon \geq |\mathrm{dgap}_k(G)| - \epsilon > 7\epsilon > \mathrm{vs}_{k,l}(C)$. So all dots of $\mathrm{VS}_{k,l}(C)$ satisfy the condition death $> \mathrm{vs}_{k,l}(C)$. Hence the derived skeleton $\mathrm{HoPeS}_{k,l}(C)$ contains exactly m critical edges corresponding to the m dots of $\mathrm{VS}_{k,l}(C)$, and so $H_1(\mathrm{HoPeS}_{k,l}(C))$ has the required rank m.

The geometric approximation $\mathrm{HoPeS}_{k,l}(C) \subset G^{2\epsilon}$ follows from Lemmas 4.25 and 4.31, with $\alpha = \mathrm{vs}_{k,l}(C) \leq \epsilon$. □

Corollary 4.33. *In the same setting as Theorem 4.32, let $\tilde{C}$ be a $(\delta + \epsilon)$-sample of the graph G. Then $\mathrm{HoPeS}_{k,l}(\tilde{C})$ is $(2\delta + 4\epsilon)$-close to $\mathrm{HoPeS}_{k,l}(C)$.*

Proof. Reconstruction Theorem 4.32 states that $\mathrm{HoPeS}_{k,l}(\tilde{C}) \subset G^{2\delta+2\epsilon}$, and $\mathrm{HoPeS}_{k,l}(C) \subset G^{2\epsilon}$. Therefore, $\mathrm{HoPeS}_{k,l}(\tilde{C}) \subset \mathrm{HoPeS}_{k,l}(C)^{2\delta+4\epsilon}$. □

4.7 The Dataset of 79K Noisy Point Clouds

We have carried out an extensive comparison of the Mapper, α-Reeb and HoPeS algorithms on both real and synthetic datasets. We produced the synthetic dataset, which we call the Planar Graph Cloud (PGC) Dataset and consists of noisy point clouds randomly sampled from planar graphs, so that we could accurately determine how effective the algorithms are at outputting reconstructed graphs that meet certain topological and geometric criteria. Since the Mapper, α-Reeb and HoPeS algorithms have guarantees relating to the homotopy type of reconstructed graphs, but not relating to the stricter homeomorphism type that captures branches outside of closed cycles, it is enough to consider for the experimental

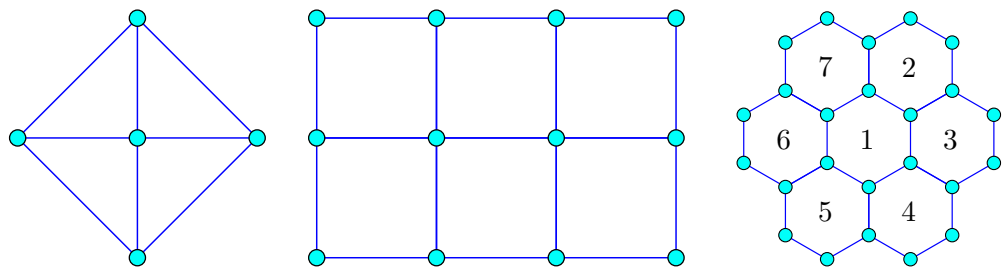

Figure 4.6: 4-wheel W(4), (3, 2)-grid G(3, 2), and 7-hexagons H(7) graphs.

comparisons graphs with no vertices of degree one. We therefore generated the PGC Dataset using three patterns of planar graphs:

- The *k-wheel graph* $\mathrm{W}(k) \subset \mathbb{R}^2$ has $k \geq 3$ circumference vertices equally distributed along the unit circle centred at the origin, at which there is another vertex. $\mathrm{W}(k)$ has edges between the central vertex and all circumference vertices, and between successive circumference vertices, see W(4) in Figure 4.6.

- For $k, l \geq 1$, the *(k, l)-grid graph* $\mathrm{G}(k, l) \subset [0, k] \times [0, l]$ has vertices at the integer coordinate points (i, j), $0 \leq i \leq k$, $0 \leq j \leq l$. Each vertex (i, j) is connected to up to four other neighbouring vertices $(i \pm 1, j)$, $(i, j \pm 1)$, see G(3, 2) in Figure 4.6.

- The *k-hexagons graph* $\mathrm{H}(k) \subset \mathbb{R}^2$ consists of the boundaries of k regular hexagons with edges of unit length. The $(k+1)$-hexagons graph is obtained from the k-hexagons graph by adding the boundary of a new hexagon. The last image in Figure 4.6 shows the order in which hexagons are added for $k \leq 7$.

To produce the dataset of noisy point clouds from these families of graphs, for each graph G we sampled 100 points per unit length (e.g. 400 points for G(1, 1)), with each point being sampled in the following way. By fixing the order of edges, G can be parameterised by a single variable t that takes values in the interval $\left[0, \sum_{i=1}^{k} l_i\right]$, where $l_1, \ldots, l_k$ are the lengths of the edges $e_1, \ldots, e_k \subset G$. A value of t is uniformly selected, and then if t belongs to the j-th interval $\left[\sum_{i=1}^{j-1} l_i, \sum_{j-1}^{j} l_i\right]$, a point is chosen from the j-th edge, corresponding to the weighted combination $w\vec{u} + (1 - w)\vec{v}$ of the edge's endpoints $u, v \in \mathbb{R}^2$, where $w = \left(t - \sum_{i=1}^{j-1} l_i\right) / l_j$.

For each sampled point p lying on an edge e of the graph G, we generate two independent random shifts d_e, $d_\perp$ from the same distribution, and then shift p by d_e in the direction parallel to e, and by $d_\perp$ in the direction perpendicular to e. The distributions used are:

- Uniform noise with bound μ: $d_e, d_\perp$ are uniformly selected from the interval $[-\mu, \mu]$.

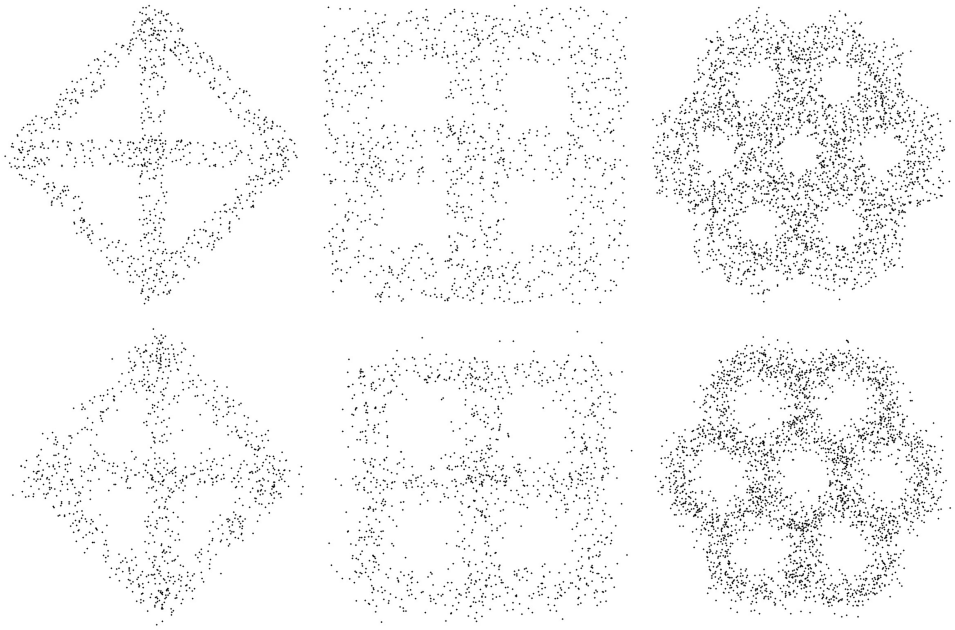

Figure 4.7: Point clouds of the PGC Dataset. **Top:** samples with uniform noise and bounds $\mu = 0.1$, $\mu = 0.25$, $\mu = 0.5$. **Bottom:** samples with Gaussian noise and deviations $\sigma = 0.08$, $\sigma = 0.12$, $\sigma = 0.2$.

- Gaussian noise with mean 0 and standard deviation σ: $d_e, d_\perp$ have the Gaussian density $f(t, \sigma) = \frac{1}{2\pi\sigma^2} e^{\frac{-t^2}{2\sigma^2}}$ for $t \in \mathbb{R}$.

The Planar Graph Cloud Dataset contains 79000 clouds consisting of 200 clouds of each of the following 395 types:

- 70 wheel types: point clouds are sampled from $\mathrm{W}(k)$, $3 \leq k \leq 9$, with uniform noise with five values of the upper bound μ ranging from 0.05 to 0.25 in 0.05 intervals, or with Gaussian noise with five values of the standard deviation σ ranging from 0.02 to 0.1 in 0.02 intervals.
- 108 grid types: point clouds are sampled from $\mathrm{G}(k, l)$, $1 \leq k, l \leq 3$, $k \geq l$, with uniform noise with 8 values of the upper bound μ ranging from 0.05 to 0.4 in 0.05 intervals, or with Gaussian noise with 10 values of the standard deviation σ ranging from 0.02 to 0.2 in 0.02 intervals.
- 217 hexagonal types: point clouds are sampled from $\mathrm{H}(k)$, $1 \leq k \leq 7$, with uniform noise with fifteen values of the upper bound μ ranging from 0.05 to 0.75 in 0.05

intervals, or with Gaussian noise with 16 values of the standard deviation σ ranging from 0.02 to 0.32 in 0.02 intervals.

The above intervals for the noise parameters were chosen to report the maximum noise at which the skeletonisation algorithms have a high success rate of outputting reconstructions with the correct first Betti number (Definition 1.26). Examples of point clouds in the PGC Dataset can be seen in Figure 4.7.

4.8 Drawing and Simplifying Skeletons of Point Clouds

Lemma 4.15 states that $\mathrm{HoPeS}(C;\alpha)$ is a subgraph of the complex $Q(C;\alpha)$. Therefore, if a point cloud $C \subset \mathbb{R}^n$ and the filtration of α-offsets $\{C^\alpha\}$ is used to derive $\mathrm{HoPeS}(C)$, then $\mathrm{HoPeS}(C) \subseteq \mathrm{Del}(C)$. Hence, $\mathrm{HoPeS}(C)$ is embedded in $\mathbb{R}^n$, thus visualising the shape of C directly in the space where C lives. In particular, if $C \subset \mathbb{R}^2$, then $\mathrm{HoPeS}(C)$, and its whole family of reduced and derived skeletons, are embedded in the plane. However, both Mapper and α-Reeb outputs are abstract, and so do not live in the same space as C. Therefore, in order to draw these outputs in the plane when $C \subset \mathbb{R}^2$, it is necessary for us to project them to the plane as naturally as possible.

For Mapper, each vertex v in the output graph corresponds to a cluster C_v of points in the cloud C, and so we naturally embed this vertex in $\mathbb{R}^2$ by placing it at the geometric centre $\frac{1}{|C_v|}\sum_{p\in C_v} p$.

It is less natural to embed the α-Reeb graph into $\mathbb{R}^2$, since it is defined to be the quotient of a set of intervals under an equivalence relation. Noting this difficulty, we argue that the most natural way of embedding this output is, in a similar way to Mapper, to map the centre of each interval to the geometric centre of its corresponding connected subgraph.

In all cases, edges between vertices are drawn as straight line segments. However, unlike any output of HoPeS which has no intersecting edges, embedding the Mapper and α-Reeb outputs in this way can lead to the intersection of edges in locations other than at vertices.

Since the algorithms are compared on noisy samples of planar graphs that have no vertices of degree one, the outputs of all three algorithms undergo a simplification by having all vertices of degree one (and their corresponding open edges) iteratively removed, see Figures 4.8 and 4.9.

We apply a second simplification to outputs of the HoPeS algorithm. This is because $\mathrm{HoPeS}(C)$, and all of its derived and reduced skeletons, have vertex set C, yet an ideal output should be simpler than the input, with much fewer vertices. Hence, we provide an algorithm below that, after removing vertices of degree one, further reduces the number of vertices in $\mathrm{HoPeS}(C)$.

Algorithm 4.34 (Simplifying $\mathrm{HoPeS}(C)$)**.** Firstly, all vertices of degree one (and their corresponding open edges) are iteratively removed. Then, for a given threshold ϵ, we iteratively collapse all edges of $\mathrm{HoPeS}(C)$ with lengths less than ϵ as follows:

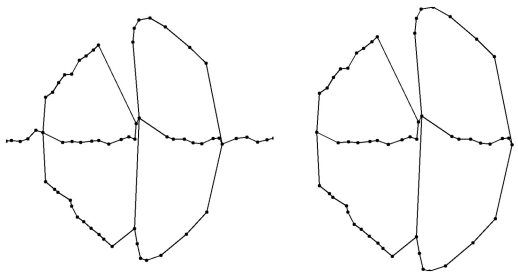

Figure 4.8: **Left:** an original Mapper output for a noisy sample of $W(4)$ in the top-left of Figure 4.7. **Right:** the simplified Mapper output by pruning vertices of degree one.

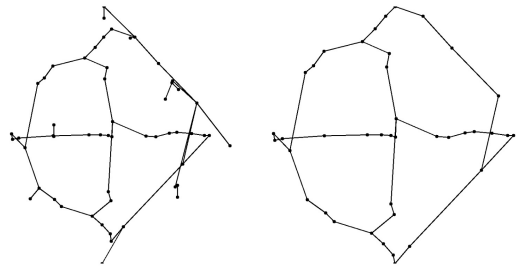

Figure 4.9: **Left:** an original α-Reeb output for a noisy sample of $W(4)$ in the top-left of Figure 4.7. **Right:** the simplified α-Reeb output by pruning vertices of degree one.

Stage 1: We skip any edge shorter than ϵ that is contained in a triangular cycle (a cycle with three edges) in order to preserve the first homology group $H_1(\mathrm{HoPeS}(C))$.

Stage 2: To collapse an edge e with endpoints v_1, v_2, we first remove v_1, v_2 and all edges incident on them, storing in memory all other vertices that either v_1 or v_2 were directly connected to.

Stage 3: If $\deg(v_1) = \deg(v_2) = 2$, or $\deg(v_1) \neq 2$ and $\deg(v_2) \neq 2$, a new vertex v is placed at the midpoint of the straight line segment between v_1 and v_2. If (say) $\deg(v_1) \neq 2$ but $\deg(v_2) = 2$, then the new vertex v is placed at the original location of v_1, to better preserve the geometric approximation of the skeleton.

Stage 4: We add an edge between the new vertex v and each of the vertices stored in memory, which were the original neighbours of either v_1 or v_2.

Stage 5: If any of the new edges intersect with existing edges, the collapse is reversed and we skip over this edge.

We denote the output of Algorithm 4.34 by $\mathrm{simHoPeS}(C;\epsilon)$. Algorithm 4.34 can also be applied to reduced skeletons $\mathrm{HoPeS}(C;\alpha)$ and derived skeletons $\mathrm{HoPeS}_{k,l}(C)$, denoting the outputs as $\mathrm{simHoPeS}(C;\alpha,\epsilon)$ and $\mathrm{simHoPeS}_{k,l}(C;\epsilon)$ respectively. An example of the simplification process can be seen in Figure 4.10.

In the experiments of Section 4.9, we always set ϵ to be equal to the maximum death over all dots of $\mathrm{DS}_1(C)$ (dropping ϵ from the notation for simplicity). It is possible that such a simplification breaches the geometric approximation guarantees of Theorems 4.28 and 4.32, but the homology guarantees remain valid.

Corollary 4.35. *Under the conditions of Theorems 4.28 and 4.32,* $\mathrm{simHoPeS}_{k,l}(C;\epsilon)$ *has the same first homology group as the underlying graph G.*

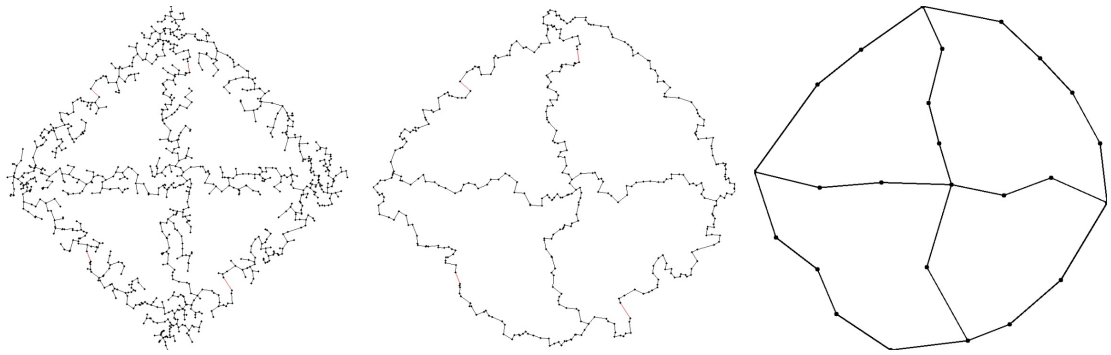

Figure 4.10: **Left:** derived $\mathrm{HoPeS}_{1,1}(C)$ for a point cloud sample of $W(4)$. **Middle:** all degree one vertices have been removed. **Right:** $\mathrm{simHoPeS}(C;\alpha)$ obtained by Algorithm 4.34.

Proof. Removing vertices of degree one does not effect the first homology group of a graph. Collapsing short edges can only lead to a change in the first homology group if either a triangular cycle is collapsed or, considering the output as an embedded graph in $\mathbb{R}^2$, the collapse leads to edges intersecting in locations other than at vertices. Algorithm 4.34 prevents both cases from arising, and so the first homology group of $\mathrm{simHoPeS}_{k,l}(C;\epsilon)$ is isomorphic to $H_1(G)$ in the cases specified by Theorems 4.28 and 4.32. □

Figures 4.11 - 4.13 show typical outputs of all three algorithms after the described simplifications for point clouds from the PGC Dataset

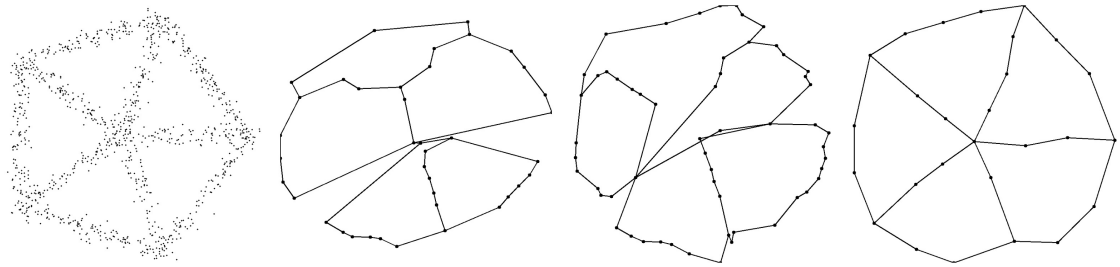

Figure 4.11: From left to right: a cloud sampled from the $W(5)$ graph with Gaussian noise with $\sigma = 0.04$; the Mapper output; the α-Reeb output; $\mathrm{simHoPeS}_{1,1}(C)$

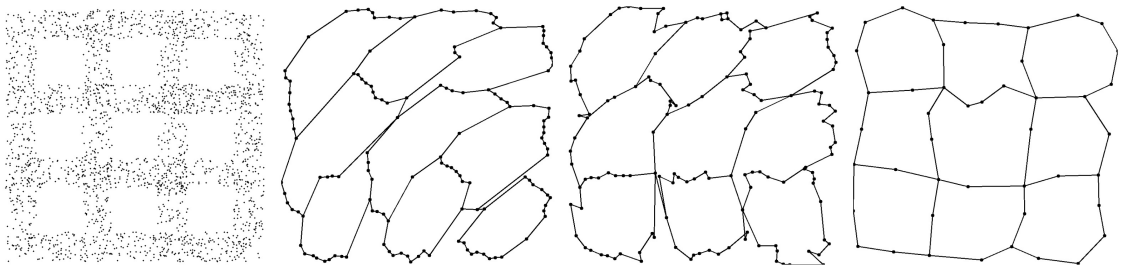

Figure 4.12: From left to right: a cloud sampled from the $G(3,3)$ graph with uniform noise with $\mu = 0.2$; the Mapper output; the α-Reeb output; $\text{simHoPeS}_{1,1}(C)$

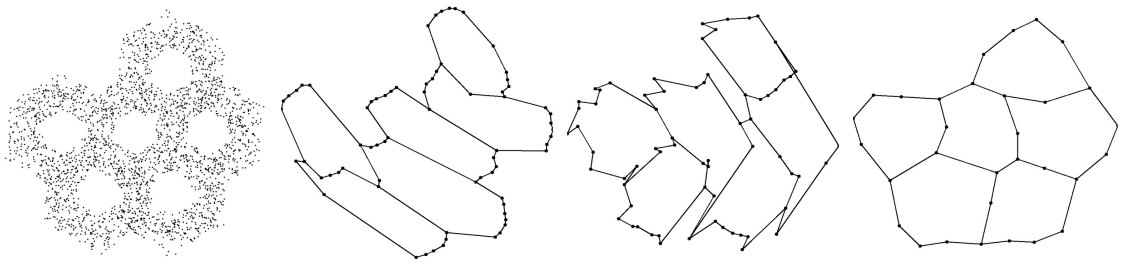

Figure 4.13: From left to right: a cloud sampled from the $H(6)$ graph with uniform noise with $\mu = 0.4$; the Mapper output; the α-Reeb output; $\text{simHoPeS}_{1,1}(C)$

4.9 Experiments on Synthetic and Real Data

We present in this section an experimental comparison of the three skeletonisation algorithms already introduced (Mapper, α-Reeb, and HoPeS) on the synthetic PGC Dataset from Section 4.7, as well as on real data.

Recall that the PGC Dataset consists of 395 types of cloud, with each type distinguished by its underlying graph G, the type of noise and the magnitude of noise. The outputs of the skeletonisation algorithms on the 200 clouds of each type are used to compute four measures for comparison:

- **The Betti success rate:** the percentage of the 200 outputs that have first homology groups of the same rank as $H_1(G)$ (i.e. the first Betti numbers of the output and the underlying graph G agree).
- **The homeomorphism success rate:** the empirical conditional probability (as a percentage) that the output skeleton is homeomorphic to the underlying graph G, given that the skeleton already agrees with G on the first Betti number.

- **The root mean square (RMS) error:** this measures how close geometrically an output skeleton S is to a cloud C. For each point $p \in C$, the distance $d(p, S)$ is the Euclidean distance of p from the closest point of S (which could be a point in the interior of an edge). Then the RMS error is $\sqrt{\sum_{p \in C} d(p, S)^2}$. We average over only the outputs that agree with G on the first Betti number.

- **The runtime:** the time taken to produce a skeleton, averaged over all 200 outputs.

Each algorithm depends on input parameters that affect the output. For HoPeS, the choice of parameters is straightforward: we simply take the derived skeleton $\mathrm{HoPeS}_{1,1}(C)$ consisting of critical edges that correspond to dots of $\mathrm{DS}_1(C)$, and simplify the output by Algorithm 4.34 using as the threshold ϵ the maximum death among these critical edges.

Yet for the Mapper and α-Reeb algorithms, the quality of the output skeletons is much more sensitive to the input parameters, and choosing the same configuration of parameters for each type of cloud would be inadequate. If parameters such as α in the α-Reeb algorithm, or ϵ used in DBSCAN as part of the Mapper algorithm, are too small, then noisy cycles will also be captured in addition to the dominant cycles. Yet if these parameters are too large, even the dominant cycles will be missed. Hence, for each cloud type, we optimise the parameters separately for each measure (apart from the running time) by searching over a wide range of parameters. The results presented below are for the parameters that yielded the best outputs for Mapper and α-Reeb (while the same parameters used to achieve the best Betti success rate are again used to compute the mean runtime). The range of values of the parameters used in the Mapper and α-Reeb algorithms are as follows:

- Mapper: the amount of overlap of the intervals that cover the range of the filter function was fixed at 50%. For a cloud of n points, we took $tn/100$ as the number of intervals, rounded to the nearest integer, where t took 10 values from 1.5 to 3.3 in 0.02 intervals. The clustering parameter ϵ also took 10 values from 0.05 to 0.5 in 0.05 intervals. We therefore used 100 configurations of the Mapper parameters.

- α-Reeb: for the scale α, we took 10 values from 0.15 to 0.6 in 0.05 intervals.

Figures 4.14-4.19 show a selection of the results for clouds sampled from wheel, grid and hexagons graphs with uniform or Gaussian noise. In each plot, the x-coordinate is the first Betti number of the underlying graph, whilst the y-coordinate is the mean value over all qualifying clouds (with optimal parameters for the Mapper and α-Reeb algorithms). Figures 4.20 and 4.21 illustrate the maximum noise each algorithm can tolerate whilst maintaining quality outputs (a Betti success rate of at least 90% and 95% respectively).

The real data used to compare the skeletons are clouds of Canny edge pixels extracted from 500 real images of the Berkeley Segmentation Database (BSD500) [41]. Table 4.1 shows how well the outputs geometrically approximate the clouds, showing the mean RMS

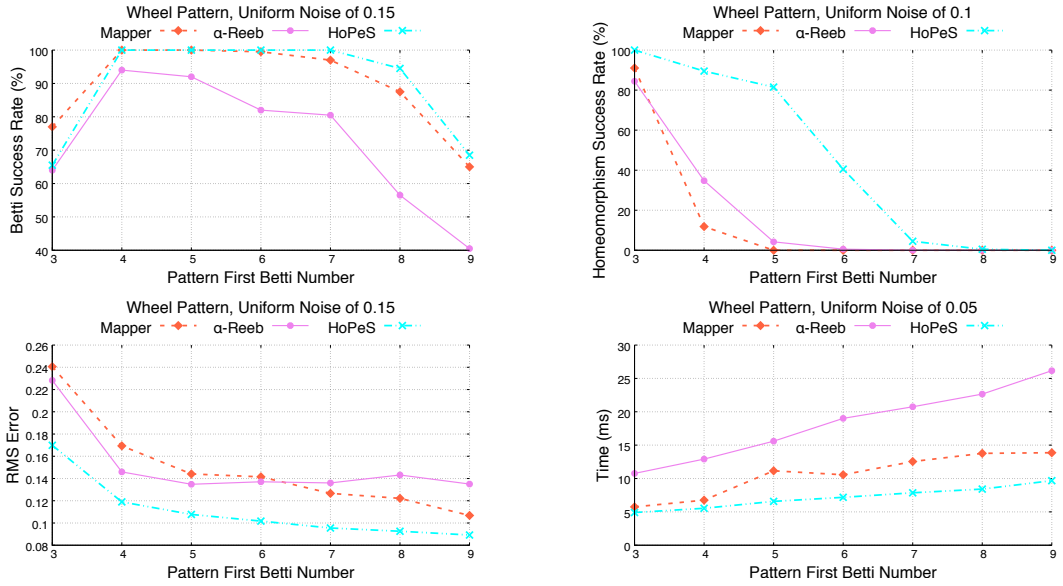

Figure 4.14: Clouds are sampled with uniform noise around wheel graphs $\mathrm{W}(k)$, $3 \leq k \leq 9$. **Top-left:** Betti success rate; **Top-right:** homeomorphism success rate; **Bottom-left:** RMS error; **Bottom-right:** runtime.

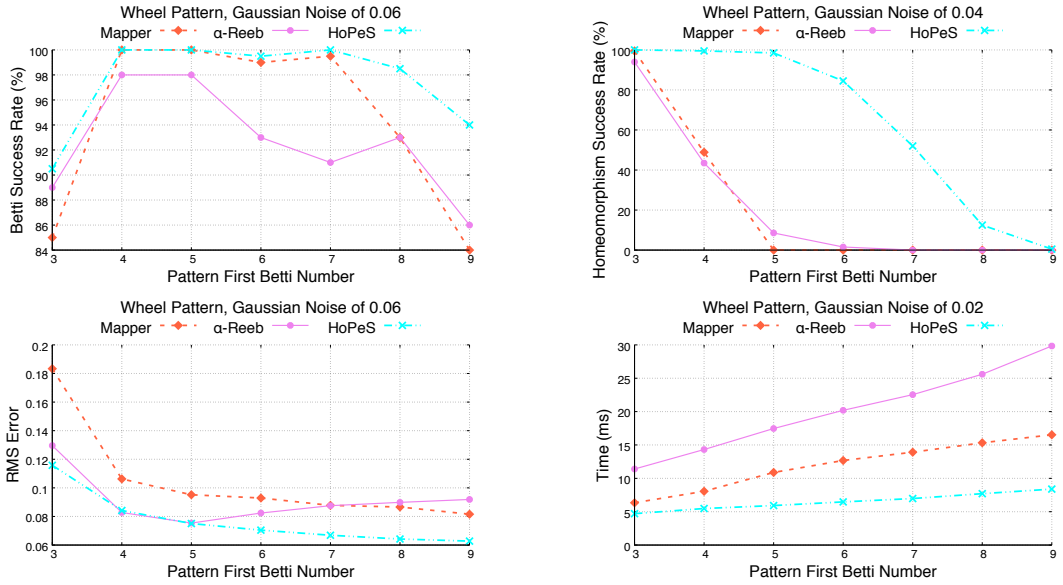

Figure 4.15: Clouds are sampled with Gaussian noise around wheel graphs $\mathrm{W}(k)$, $3 \leq k \leq 9$. **Top-left:** Betti success rate; **Top-right:** homeomorphism success rate; **Bottom-left:** RMS error; **Bottom-right:** runtime.

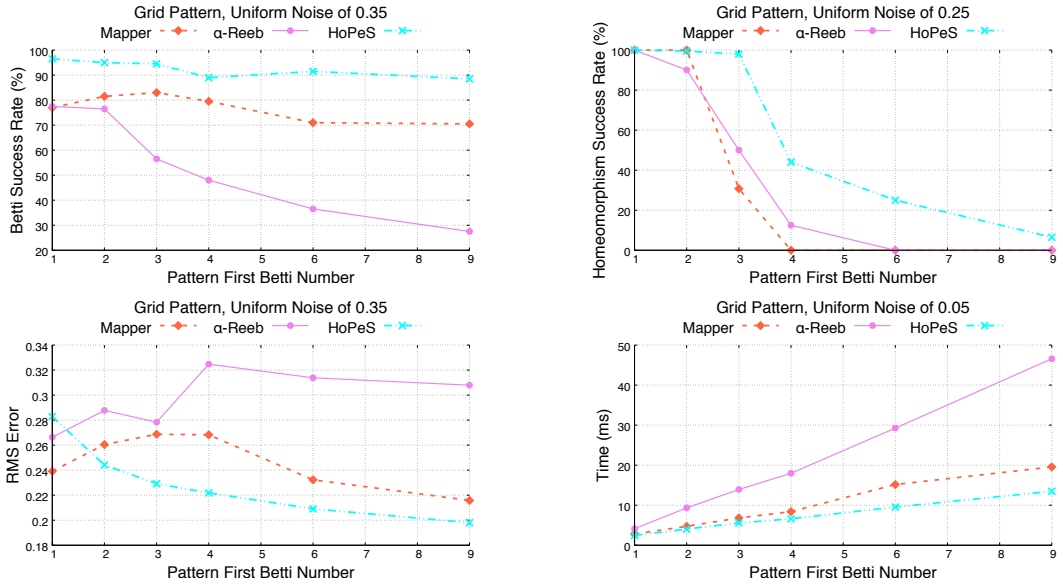

Figure 4.16: Clouds are sampled with uniform noise around grid graphs $\mathrm{G}(k,l)$, $1 \leq k,l \leq 3$. **Top-left:** Betti success rate; **Top-right:** homeomorphism success rate; **Bottom-left:** RMS error; **Bottom-right:** runtime.

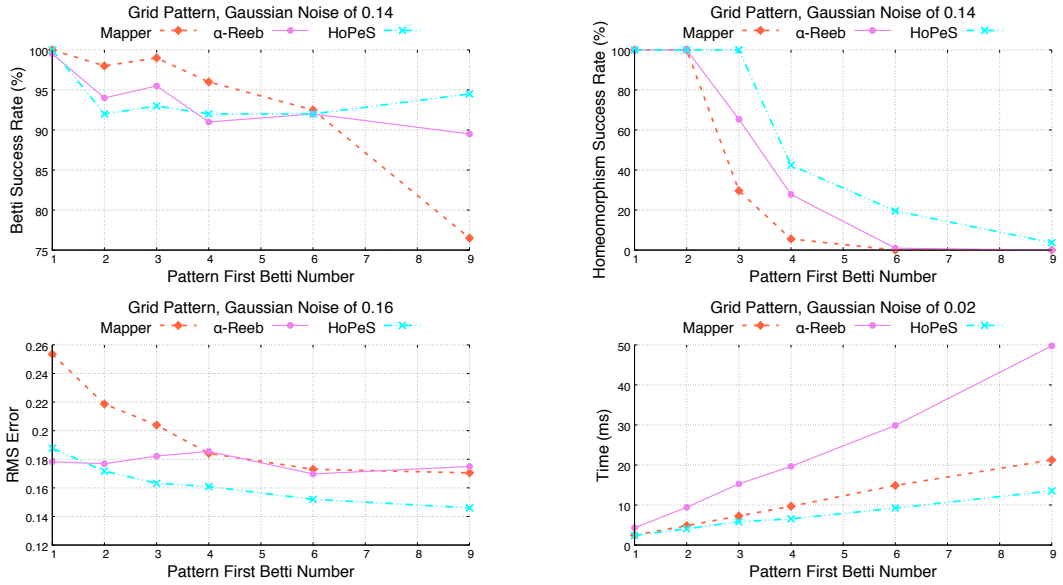

Figure 4.17: Clouds are sampled with Gaussian noise around grid graphs $\mathrm{G}(k,l)$, $1 \leq k,l \leq 3$. **Top-left:** Betti success rate; **Top-right:** homeomorphism success rate; **Bottom-left:** RMS error; **Bottom-right:** runtime.

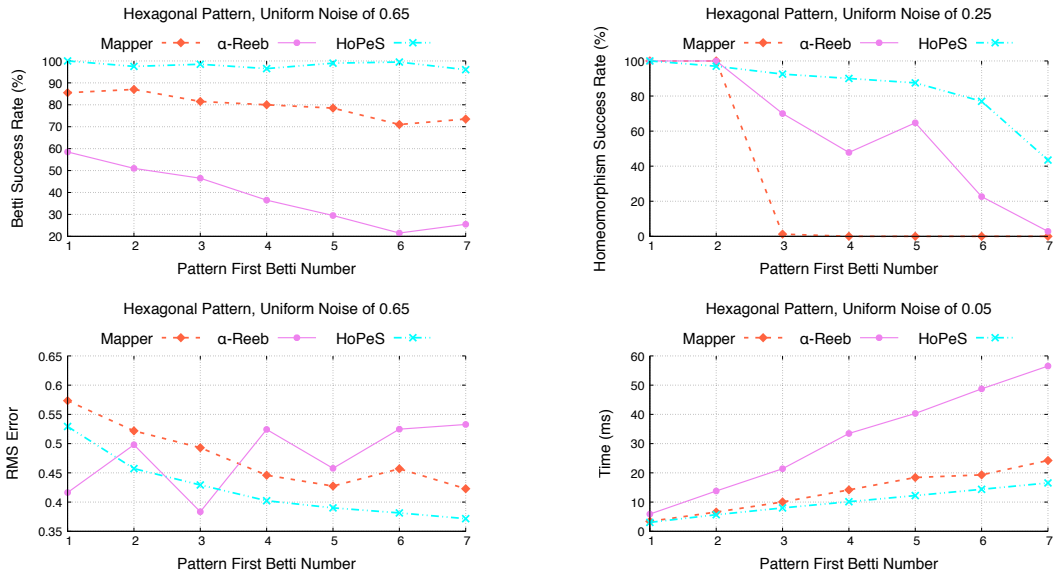

Figure 4.18: Clouds are sampled with uniform noise around hexagon graphs $\mathrm{H}(k)$, $1 \leq k \leq 7$. **Top-left:** Betti success rate; **Top-right:** homeomorphism success rate; **Bottom-left:** RMS error; **Bottom-right:** runtime.

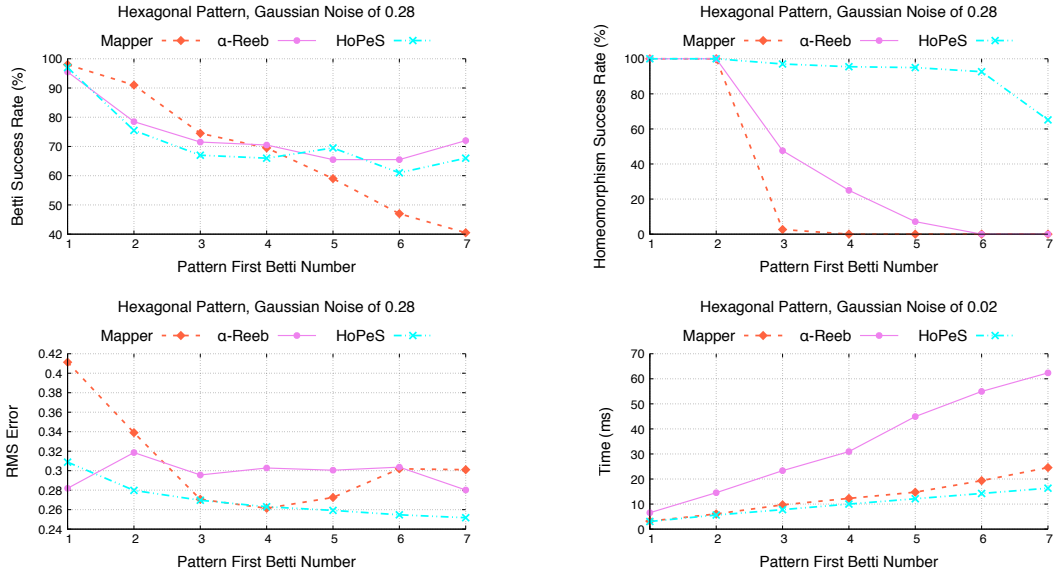

Figure 4.19: Clouds are sampled with Gaussian noise around hexagon graphs $\mathrm{H}(k)$, $1 \leq k \leq 7$. **Top-left:** Betti success rate; **Top-right:** homeomorphism success rate; **Bottom-left:** RMS error; **Bottom-right:** runtime.

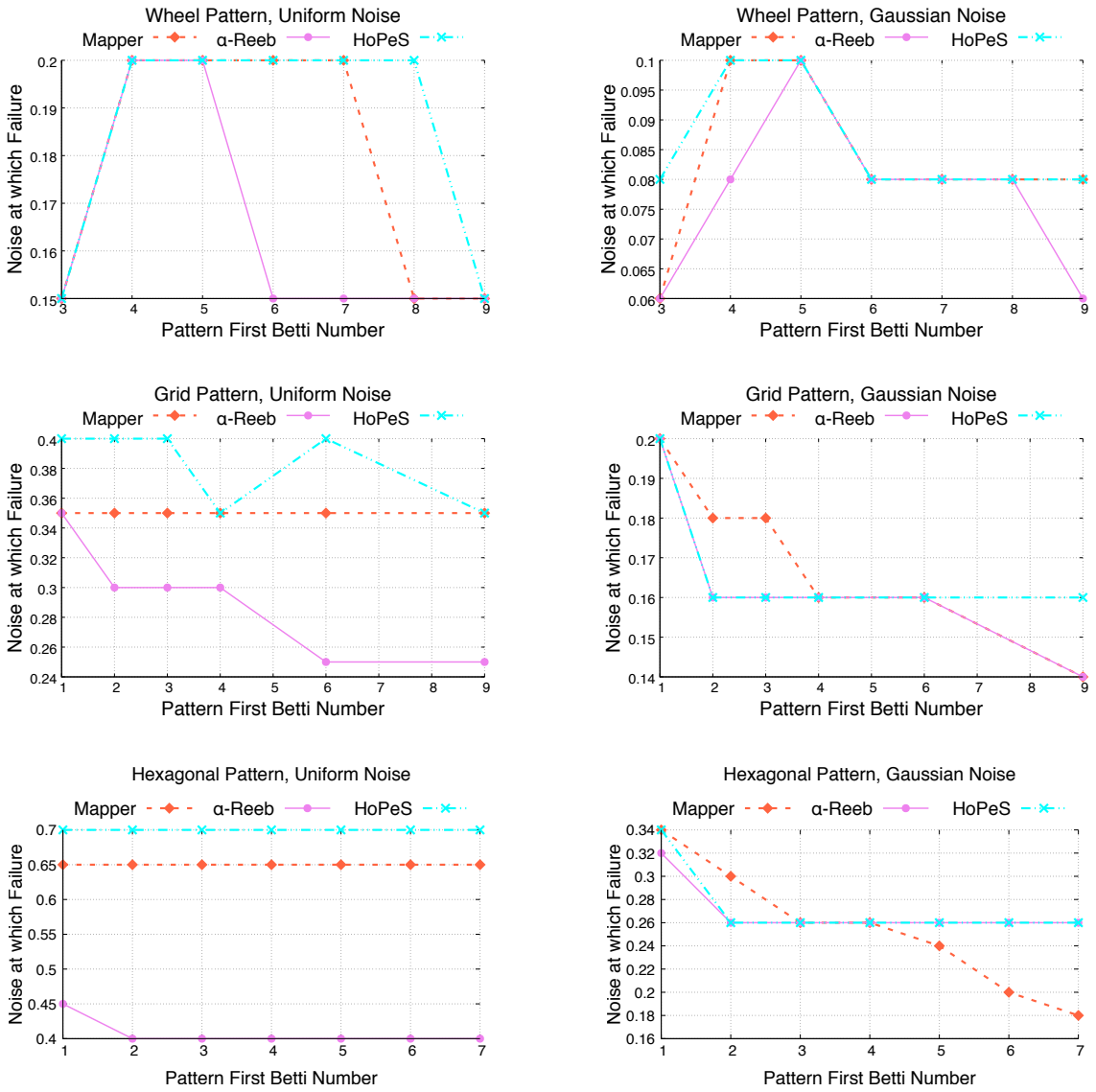

Figure 4.20: The magnitude of noise at which the Betti success rate drops below 90%. **Top-left:** wheel pattern, uniform noise; **Top-right:** wheel pattern, Gaussian noise; **Middle-left:** grid pattern, uniform noise; **Middle-right:** grid pattern, Gaussian noise; **Bottom-left:** hexagons pattern, uniform noise; **Bottom-right:** hexagons pattern, Gaussian noise.

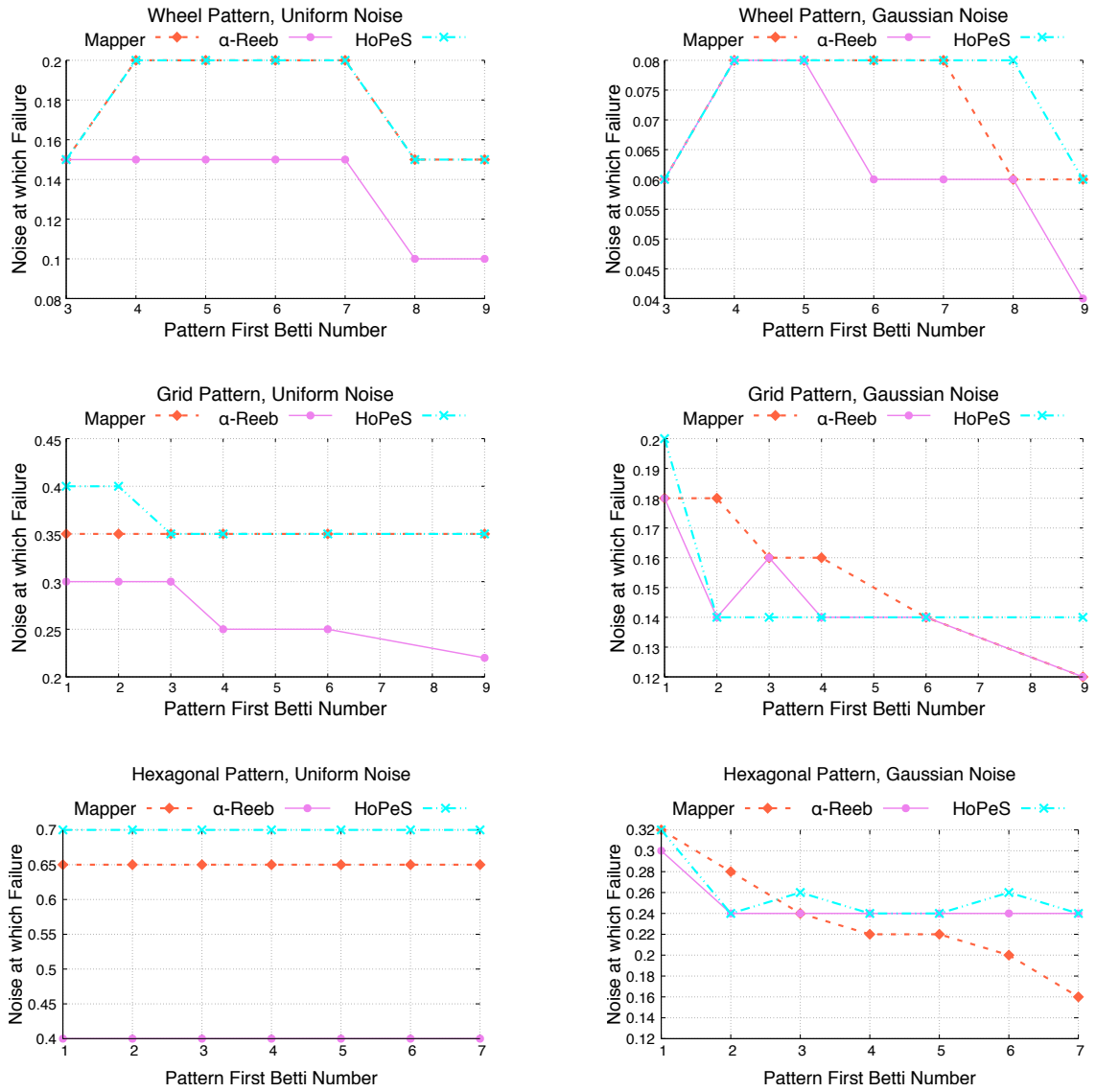

Figure 4.21: The magnitude of noise at which the Betti success rate drops below 95%. **Top-left:** wheel pattern, uniform noise; **Top-right:** wheel pattern, Gaussian noise; **Middle-left:** grid pattern, uniform noise; **Middle-right:** grid pattern, Gaussian noise; **Bottom-left:** hexagons pattern, uniform noise; **Bottom-right:** hexagons pattern, Gaussian noise.

Measures / Algorithms	Mapper	α-Reeb	simHoPeS	HoPeS
RMS error (pixels)	10.726	6.48247	5.61771	0
Max distance (pixels)	55.892	45.0883	29.1306	0
Runtime (ms)	310	4110	1256	88

Table 4.1: Comparison of the algorithms on Canny edge pixels of 500 real images in the BSD500 dataset.

error between the cloud and the skeleton, the maximum distance between a point of the cloud and the skeleton, and the runtime. $\text{HoPeS}_{1,1}(C)$ is an extension of a minimum spanning tree, and so of course has an RMS error of zero. But we include the simpler version $\text{simHoPeS}_{1,1}(C)$ to provide a fairer comparison. The Mapper and α-Reeb outputs are again optimised over a range of parameters. Vertices of degree one were not removed for any of the outputs in these experiments on real data. Figures 4.22-4.25 are included to provide the reader with a visual comparison of the quality of the outputs.

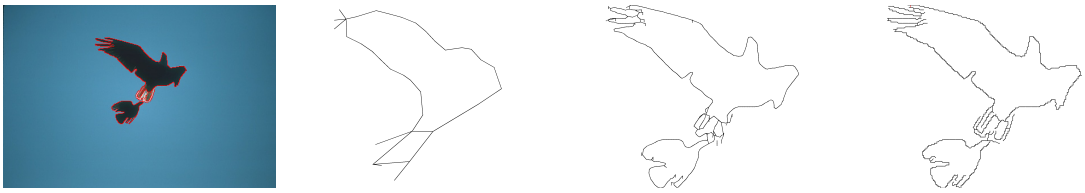

Figure 4.22: From left to right: image 135069 from the BSD500 dataset with the cloud C of Canny edge pixels in red; the outputs of the Mapper, α-Reeb and $\text{HoPeS}_{1,1}(C)$ algorithms, where the single critical edge of $\text{HoPeS}_{1,1}(C)$ is in red.

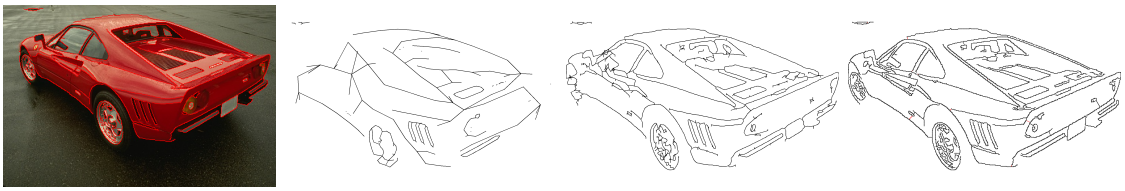

Figure 4.23: From left to right: image 29030 from the BSD500 dataset with the cloud C of Canny edge pixels in red; the outputs of the Mapper, α-Reeb and $\text{HoPeS}_{1,1}(C)$ algorithms, where the critical edges of $\text{HoPeS}_{1,1}(C)$ are in red.

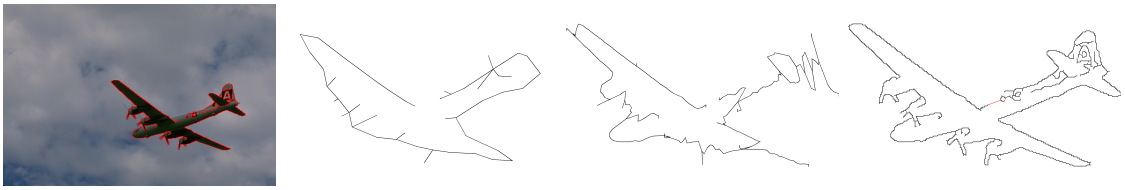

Figure 4.24: From left to right: image 3096 from the BSD500 dataset with the cloud C of Canny edge pixels in red; the outputs of the Mapper, α-Reeb and $\mathrm{HoPeS}_{1,1}(C)$ algorithms, where the single critical edge of $\mathrm{HoPeS}_{1,1}(C)$ is in red.

Figure 4.25: From left to right: image 302003 from the BSD500 dataset with the cloud C of Canny edge pixels in red; the outputs of the Mapper, α-Reeb and $\mathrm{HoPeS}_{1,1}(C)$ algorithms, where the critical edges of $\mathrm{HoPeS}_{1,1}(C)$ are in red.

4.10 Conclusions: Pluses and Minuses of the Algorithms

The key advantage of the Mapper algorithm is its versatility due to the various parameters that the user can change. Yet this abundance of choice can lead to difficulties as a prior knowledge of the dataset is often needed to select parameters that yield meaningful outputs. The α-Reeb algorithm has effectively just the scale α as its only parameter, and an α-Reeb graph can be computed in the very fast time $\mathcal{O}(n \log(n))$. Theoretically, HoPeS has the advantage of being a parameter-free algorithm that is also embedded in the same space as the point cloud C. Hence, the time-consuming process of optimising parameters is not required and the intersection of edges is avoided. Yet HoPeS maintains versatility since the output contains families of reduced and derived subskeletons that may better describe the geometry of the point cloud.

Moreover, the experimental comparisons of Section 4.9 show that, despite optimising the important parameters of the Mapper and α-Reeb algorithms over a wide range of values, the derived skeleton $\mathrm{simHoPeS}_{1,1}(C)$ performs comparably or even better without the need to optimise parameters. The HoPeS algorithm could be improved to deal better

with outliers, since there is a drop in its performance on clouds with Gaussian noise, whilst all algorithms can be improved by a further minimisation of the RMS error of the skeleton to the cloud.

To summarise, this chapter gives detailed proofs of Optimality Theorem 4.21 and Reconstuction Theorems 4.28 and 4.32 for the first time, whilst Corollaries 4.33 and 4.35 are new results. The extensive comparison of the three algorithms on synthetic and real data in Section 4.9 reveals for the first time the maximum levels of noise at which the algorithms reliably produce quality outputs. The C++ code implementing all three algorithms is available at [55], whilst the PGC Dataset from Section 4.7 of 79000 point clouds (2GB) for comparison with other skeletonisation algorithms is available by request.

Chapter 5

Conclusion

The emerging field of Crystal Structure Prediction (CSP) is enabling the discovery of new materials at an ever-increasing rate. Yet, the large datasets of simulated crystals that are outputted by CSP are largely unstructured due to a lack of a rigorous comparison of crystals that is invariant under isometries, continuous and complete. We have presented two main directions of research that can be applied to these large CSP datasets to better and more quickly analyse them.

Firstly, in response to Problem 1, we have introduced the density fingerprint (Definition 3.5) which is a continuous isometry invariant of periodic sets (which model crystals). In particular, the density fingerprint contains a wealth of information that is stored in an infinite sequence of continuous density functions. We have been able to show that it is invariant under isometries (Lemma 3.4), Lipschitz continuous for small perturbations (Theorem 3.10) and complete for an open and dense space of periodic sets in $\mathbb{R}^3$. Our failure to find a counterexample to completeness in Subsection 3.4.1 in any dimension greater than one has led us to conjecture that the density fingerprint map is complete for all periodic sets in $\mathbb{R}^n$, $n \geq 2$ (Conjecture 3.19).

A prerequisite for computing the density fingerprint of a periodic set are the Voronoi zones (Definition 2.4) introduced in Chapter 2. Here, we have generalised previous related work on lattices to periodic sets, which are better at modelling crystals since crystals can have multiple atoms or molecules within its motif. We have structurally described these zones, observing their spherical nature in Theorem 2.6 and stating in Theorem 2.12 how the total volume of each zone remains constant as the order k increases. In Section 2.2, we have introduced an algorithm computing the first k Voronoi zones of a periodic set for a given order k.

It is Theorem 3.21 that states how density functions can be computed via Voronoi zones. Specifically, we can deduce from this theorem that, in dimension three, density functions can be computed as a sum of the volumes of sphere-tetrahedron intersections, where exactly computing such volumes is described in Subsection 3.5.1. Density functions

can be computed in cubic time in the order k if the packing and covering radii of the periodic set are known. Implementations of the algorithms to compute Voronoi zones of periodic sets and density functions can be found at [54, 53] respectively.

Density fingerprints can easily be compared using the d_∞-distance. Therefore, we expect the fingerprint can be used to simplify the large output datasets produced by CSP by comparing simulated structures with each other. This will enable near-duplicates to be removed from the dataset, and clustering or even skeletonisation algorithms to visualise the dataset's structure. A small example of how the density fingerprint can be applied is described in Section 3.6.

The second direction of research relates to skeletonisation algorithms. Particularly, we have carried out an extensive comparison of three relevant skeletonisation algorithms (Mapper [51], α-Reeb [11] and HoPeS [37]) on synthetic and real datasets. This comparison has revealed that on point clouds with uniform noise, HoPeS performs comparably or even better on objective measures, without the need to optimise parameters. On Gaussian noise with outliers, while HoPeS continues to perform comparably, we note that the algortihm could be improved to better accommodate outliers. The generation of the synthetic Planar Graph Cloud Dataset has been described in Section 4.7, and can be used in future research to compare other skeletonisation algorithms.

We have proven optimality and reconstruction guarantees for HoPeS in Theorems 4.21, 4.28 and 4.32, in addition to describing an algorithm to simplify the output of HoPeS by reducing the number of vertices (Algorithm 4.34). Skeletonisation algorithms have wider applications than just to CSP outputs, as they can be used to visualise the structure of any dataset that can be represented as a point cloud in a metric space.

We conclude this thesis by reminding the reader of some open problems. The proof of generic completeness of the density fingerprint map (Theorem 3.14) makes only limited use of the order k at which the circumradius of an edge, triangle, or tetrahedron is detected (where the order is the number of points in the respective circumsphere). This raises the question of whether this additional information is sufficient to prove completeness for all periodic sets in dimensions greater than one?

Separately, different types of atoms are often modelled as balls with different radii. A possible geometric formalism is that of weighted points and the power distance [7]. The geometric results for the density fingerprint generalise to this setting, although some need a careful adaptation. The continuity result for periodic sets (Theorem 3.10) also generalises to non-periodic Delone sets that allow for a reasonable definition of density functions. Considering that quasi-periodic crystals can be modelled as such, finding out how far such an extension can be pushed may be a worthwhile direction of future research.

Bibliography

[1] Mridul Aanjaneya et al. "Metric graph reconstruction from noisy data". In: *International Journal of Computational Geometry & Applications* 22.4 (2012), pp. 305–325. DOI: 10.1142/S0218195912600072.

[2] Radhakrishna Achanta et al. "SLIC superpixels compared to state-of-the-art superpixel methods". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34.11 (2012), pp. 2274–2282. DOI: 10.1109/TPAMI.2012.120.

[3] Richard C. Andrew, Trisha Salagaram, and Nithaya Chetty. "Visualising higher order Brillouin zones with applications". In: *European Journal of Physics* 38.3 (2017), p. 035501. DOI: 10.1088/1361-6404/aa5e0d.

[4] Lawrence C. Andrews, Herbert J. Bernstein, and G.A. Pelletier. "A perturbation stable cell comparison technique". In: *Acta Crystallographica* A36.2 (1980), pp. 248–252. DOI: 10.1107/S0567739480000496.

[5] Olga Anosova and Vitaliy Kurlin. "An isometry classification of periodic point sets". In: *Proceedings of Discrete Geometry and Mathematical Morphology*. 2021. URL: https://livrepository.liverpool.ac.uk/3119390/.

[6] Olga Anosova and Vitaliy Kurlin. *Introduction to periodic geometry and topology*. 2021. arXiv: 2103.02749.

[7] Franz Aurenhammer. "Power diagrams: properties, algorithms and applications". In: *SIAM Journal on Computing* 16.1 (1987), pp. 78–96. DOI: 10.1137/0216006.

[8] Michael Van den Bergh et al. "SEEDS: superpixels extracted via energy-driven sampling". In: *International Journal of Computer Vision* 111.3 (2015), pp. 298–314. DOI: 10.1007/s11263-014-0744-2.

[9] Ilya Bronshtein, Gregory Leitus, and Boris Rybtchinski. "In situ growth of high quality crystals for organic electonics". In: *ACS Applied Electonic Materials* 2.3 (2020), pp. 790–795. DOI: 10.1021/acsaelm.9b00845.

[10] Mathieu Carrière and Steve Oudot. "Structure and stability of the one-dimensional Mapper". In: *Foundations of Computational Mathematics* 18.6 (2018), pp. 1333–1396. DOI: 10.1007/s10208-017-9370-z.

[11] Frédéric Chazal, Ruqi Huang, and Jian Sun. "Gromov-Hausdorff approximation of filamentary structures using Reeb-type graphs". In: *Discrete & Computational Geometry* 53.3 (2015), pp. 621–649. DOI: 10.1007/s00454-015-9674-1.

[12] Frédéric Chazal, Vin de Silva, and Steve Oudot. "Persistent stability for geometric complexes". In: *Geometriae Dedicata* 173.1 (2014), pp. 193–214. DOI: 10.1007/s10711-013-9937-z.

[13] James A. Chisholm and Sam Motherwell. "COMPACK: a program for identifying crystal structure similarity using distances". In: *Journal of Applied Crystallography* 38.1 (2005), pp. 228–231. DOI: 10.1107/S0021889804027074.

[14] International Union of Crystallography. *IUCr definition of a Brillouin zone*. URL: https://dictionary.iucr.org/Brillouin_zone.

[15] Nikolai P. Dolbilin and Daniel H. Huson. "Periodic Delone tilings". In: *Periodica Mathematica Hungarica* 34.1 (1997), pp. 57–64. DOI: 10.1023/A:1004272423695.

[16] Nikolai P. Dolbilin, Jeffrey C. Lagarias, and Marjorie Senechel. "Multiregular point systems". In: *Discrete & Computational Geometry* 20.4 (1998), pp. 477–498. DOI: 10.1007/PL00009397.

[17] Herbert Edelsbrunner. "The union of balls and its dual shape". In: *Discrete & Computational Geometry* 13.3 (1995), pp. 415–440. DOI: 10.1007/BF02574053.

[18] Herbert Edelsbrunner and Mabel Iglesias-Ham. "Multiple covers with balls I: Inclusion-exclusion". In: *Computational Geometry* 68 (2018), pp. 119–133. DOI: 10.1016/j.comgeo.2017.06.014.

[19] Herbert Edelsbrunner and Mabel Iglesias-Ham. "On the optimality of the FCC lattice for soft sphere packing". In: *SIAM Journal on Discrete Mathematics* 32.1 (2018), pp. 750–782. DOI: 10.1137/16M1097201.

[20] Herbert Edelsbrunner and Raimund Seidel. "Voronoi diagrams and arrangements". In: *Discrete & Computational Geometry* 1.1 (1986), pp. 25–44. DOI: 10.1007/BF02187681.

[21] Herbert Edelsbrunner et al. "The density fingerprint of a periodic point set". In: *The 37th International Symposium on Computational Geometry*. 2021.

[22] Yury Elkin and Vitaliy Kurlin. "The mergegram of a dendrogram and its stability". In: *Proceedings of MFCS (Mathematical Foundations of Computer Science)*. 2020, 32:1–32:13. ISBN: 978-3-959771-59-7. DOI: 10.4230/LIPIcs.MFCS.2020.32.

[23] Yury Elkin, Di Liu, and Vitaliy Kurlin. "A fast approximate skeleton with guarantees for any cloud of points". In: *Proceedings of TopoInVis 2019 (Topological Methods in Data Analysis and Visualization)*. 2019. arXiv: 2007.08900.

[24] Martin Ester et al. "A density-based algorithm for discovering clusters in large spatial databases with noise". In: *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*. 1996, pp. 226–231. ISBN: 978-1-577350-04-0. DOI: 10.5555/3001460.3001507.

[25] Gemma de la Flor et al. "Comparison of structures applying the tools available at the Bilbao Crystallographic Server". In: *Journal of Applied Crystallography* 49.2 (2016), pp. 653–664. DOI: 10.1107/S1600576716002569.

[26] Xiaoyin Ge et al. "Data skeletonization via Reeb graphs". In: *Advances in Neural Information Processing Systems*. Vol. 24. 2011, pp. 837–845. ISBN: 978-1-618395-99-3. URL: https://proceedings.neurips.cc/paper/2011/file/3a0772443a0739141292a5429b952fe6-Paper.pdf.

[27] Rafael Grompone von Gioi et al. "LSD: a line segment detector". In: *Image Processing On Line* 2 (2012), pp. 35–55. DOI: 10.5201/ipol.2012.gjmr-lsd.

[28] Alexander N. Gorban and Andrei Y. Zinovyev. "Principal graphs and manifolds". In: *Handbook of Research on Machine Learning Applications and Trends*. 2010, pp. 28–59. ISBN: 978-1-605667-66-9. DOI: 10.4018/978-1-60566-766-9.ch002.

[29] F. Alberto Grünbaum and Calvin C. Moore. "The use of higher-order invariants in the determination of generalized Patterson cyclotomic sets". In: *Acta Crystallographica* A51.3 (1995), pp. 310–323. DOI: 10.1107/S0108767394009827.

[30] Gus L.W. Hart et al. "A robust algorithm for k-point grid generation and symmetry reduction". In: *Journal of Physics Communications* 3.6 (2019). DOI: 10.1088/2399-6528/ab2937.

[31] Allen Hatcher. *Algebraic Topology*. Cambridge University Press, 2002. ISBN: 978-0-521795-40-1. URL: https://pi.math.cornell.edu/~hatcher/AT/AT.pdf.

[32] Sara Kališnik, Vitaliy Kurlin, and Davorin Lešnik. "A higher-dimensional homologically persistent skeleton". In: *Advances in Applied Mathematics* 102 (2019), pp. 113–142. DOI: 10.1016/j.aam.2018.07.004.

[33] Balázs Kégl and Adam Krzyżak. "Piecewise linear skeletonization using principal curves". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24.1 (2002), pp. 59–74. DOI: 10.1109/34.982884.

[34] Vitaliy Kurlin. "A fast and robust algorithm to count topologically persistent holes in noisy clouds". In: *IEEE Conference on Computer Vision and Pattern Recognition*. 2014, pp. 1458–1463. ISBN: 978-1-479951-18-5. DOI: 10.1109/CVPR.2014.189.

[35] Vitaliy Kurlin. "A fast persistence-based segmentation of noisy 2D clouds with provable guarantees". In: *Pattern Recognition Letters* 83.1 (2016), pp. 3–12. DOI: 10.1016/j.patrec.2015.11.025.

[36] Vitaliy Kurlin. "A homologically persistent skeleton is a fast and robust descriptor of interest points in 2D images". In: *Computer Analysis of Images and Patterns*. 2015, pp. 606–617. ISBN: 978-3-319231-92-1. DOI: 10.1007/978-3-319-23192-1_51.

[37] Vitaliy Kurlin. "A one-dimensional homologically persistent skeleton of an unstructured point cloud in any metric space". In: *Computer Graphics Forum* 34.5 (2015), pp. 253–262. DOI: 10.1111/cgf.12713.

[38] Vitaliy Kurlin. "Auto-completion of contours in sketches, maps, and sparse 2D images based on topological persistence". In: *16th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing*. 2014, pp. 594–601. ISBN: 978-1-479984-48-0. DOI: 10.1109/SYNASC.2014.85.

[39] Vitaliy Kurlin and Philip Smith. "Resolution-independent meshes of superpixels". In: *Advances in Visual Computing*. 2019, pp. 194–205. ISBN: 978-3-030337-20-9. DOI: 10.1007/978-3-030-33720-9_15.

[40] Clare F. Macrae et al. "Mercury: visualization and analysis of crystal structures". In: *Journal of Applied Crystallography* 39.3 (2006), pp. 453–457. DOI: 10.1107/S002188980600731X.

[41] David Martin et al. "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics". In: *Proceedings of the Eighth IEEE International Conference on Computer Vision. ICCV 2001*. Vol. 2. 2001, pp. 416–423. ISBN: 0-7695-1143-0. DOI: 10.1109/ICCV.2001.937655.

[42] Helena Molina-Abril and Pedro Real. "Homological spanning forest framework for 2D image analysis". In: *Annals of Mathematics and Artificial Intelligence* 64.4 (2012), pp. 385–409. DOI: 10.1007/s10472-012-9297-7.

[43] Marco Mosca and Vitaliy Kurlin. "Voronoi-based similarity distances between arbitrary crystal lattices". In: *Crystal Research and Technology* 55.5 (2020), p. 1900197. DOI: 10.1002/crat.201900197.

[44] Phong Q. Nguyen and Damien Stehlé. "Low-dimensional lattice basis reduction revisited". In: *ACM Transactions on Algorithms* 5.4 (2009). DOI: 10.1145/1597036.1597050.

[45] Paul Niggli. "Kristallographische und strukturtheoretische Grundbegriffe". In: *Handbuch der Experimentalphysik*. Vol. 7. 1. 1928.

[46] Georg Osang, Mael Rouxel-Labbé, and Monique Teillaud. "Generalizing CGAL periodic Delaunay triangulations". In: *28th European Symposium on Algorithms*. 2020, 75:1–75:17. DOI: 10.4230/LIPIcs.ESA.2020.75.

[47] Cosimo Patruno et al. "People re-identification using skeleton standard posture and color descriptors from RGB-D data". In: *Pattern Recognition* 89 (2019), pp. 77–90. DOI: 10.1016/j.patcog.2019.01.003.

[48] Linus Pauling and Maple D. Shappell. "8. The crystal structure of bixbyite and the C-modification of the sesquioxides". In: *Zeitschrift fü Kristallographie - Crystalline Materials* 75.1 (1930), pp. 128–142. DOI: 10.1515/zkri-1930-0109.

[49] Angeles Pulido et al. "Functional materials discovery using energy-structure-function maps". In: *Nature* 543.7647 (2017), pp. 657–664. DOI: 10.1038/nature21419.

[50] Susan M. Reutzel-Edens and Rajini M. Bhardwaj. "Crystal forms in pharmaceutical applications: olanzapine, a gift to crystal chemistry that keeps on giving". In: *International Union of Crystallography Journal* 7.6 (2020), pp. 955–964. DOI: 10.1107/S2052252520012683.

[51] Gurjeet Singh, Facundo Mémoli, and Gunnar Carlsson. "Topological methods for the analysis of high dimensional data sets and 3D object recognition". In: *Eurographics Symposium on Point-Based Graphics*. 2007, pp. 91–100. ISBN: 978-3-905673-51-7. DOI: 10.2312/SPBG/SPBG07/091-100.

[52] Rahul Singh, Vladimir Cherkassky, and Nikolaos Papanikolopoulos. "Self-organizing maps for the skeletonization of sparse shapes". In: *IEEE Transactions on Neural Networks* 11.1 (2000), pp. 241–248. DOI: 10.1109/72.822527.

[53] Philip Smith. *Density functions of a periodic set in C++*. GitHub Repository. 2021. URL: https://github.com/Phil-Smith1/Density_Functions.

[54] Philip Smith. *Higher periodic Voronoi zones in C++*. GitHub Repository. 2021. URL: https://github.com/Phil-Smith1/Voronoi_Zones.

[55] Philip Smith. *Skeletonisation algorithms for unorganised point clouds in C++*. GitHub Repository. 2021. URL: https://github.com/Phil-Smith1/Cloud_Skeletonization_3.

[56] Philip Smith and Vitaliy Kurlin. "Skeletonisation algorithms with theoretical guarantees for unorganised point clouds with high levels of noise". In: *Pattern Recognition* 115 (2021). DOI: 10.1016/j.patcog.2021.107902.

[57] Severin Strobl, Arno Formella, and Thorsten Pöschel. "Exact calculation of the overlap volume of spheres and mesh elements". In: *Journal of Computational Physics* 311 (2016), pp. 158–172. DOI: 10.1016/j.jcp.2016.02.003.

[58] Dey Tamal K, Jiayuan Wang, and Yusu Wang. "Graph reconstruction by discrete Morse theory". In: *The 24th International Symposium on Computational Geometry*. 2018, 31:1–31:15. ISBN: 978-3-959770-66-8. DOI: 10.4230/LIPIcs.SoCG.2018.31.

[59] Maria-Laura Torrente, Silvia Biasotti, and Bianca Falcidieno. "Recognition of feature curves on 3D shapes using an algebraic approach to Hough transforms". In: *Pattern Recognition* 73 (2018), pp. 111–130. DOI: 10.1016/j.patcog.2017.08.008.

[60] Fabio Viola, Andrew Fitzgibbon, and Roberto Cipolla. "A unifying resolution-independent formulation for early vision". In: *2012 IEEE Conference on Computer Vision and Pattern Recognition*. 2012, pp. 494–501. ISBN: 978-1-467312-27-1. DOI: 10.1109/CVPR.2012.6247713.

[61] Georges Voronoi. "Nouvelles applications des paramètres continus à la théorie des formes quadratiques. Deuxième mémoire. Recherches sur les parallèloèdres primitifs." In: *Jounal für die reine und angewandte Mathematik (Crelles Journal)* 134 (1908), pp. 198–287.

[62] Daniel Widdowson et al. *Average minimum distances of periodic point sets*. 2020. arXiv: 2009.02488.

[63] Qi Yu et al. "Photomechanical organic crystals as smart materials for advanced applications". In: *Chemistry - A European Journal* 25.22 (2019), pp. 5611–5622. DOI: 10.1002/chem.201805382.

Appendix A

Resolution-independent Meshes of Superpixels

(This short appendix describes separate work completed during the PhD, which culminated in the publication of the paper "Resolution-independent meshes of superpixels" authored by V. Kurlin and P.S. in proceedings of the 14th International Symposium on Visual Computing, 2019 [39].)

An important problem in low-level computer vision is to quickly detect key structures such as corners and edges where colour intensities substantially change. Consequently, the over-segmentation of a digital image into superpixels is an important pre-processing step, compressing the input size of the image and speeding up higher level tasks.

A superpixel was traditionally considered as a small cluster of square-based pixels that have similar colour intensities and are closely located to each other. In this discrete model, the boundaries of superpixels often have irregular zigzags consisting of horizontal or vertical edges from a given pixel grid.

However, digital images represent a continuous world, where colour intensities change gradually over two to three pixels without jumps, see [60, Figure 1]. Hence, instead of combining pixels to form a superpixel, splitting an image into polygons that have straight edges with any possible slope and vertices at sub-pixel resolution can be more suitable. We call such polygons resolution-independent superpixels. In particular, we consider the following resolution-independent formulation of the over-segmentation problem introduced by Viola et al. [60]. We split an image into a fixed number of possibly non-convex polygons such that:

- All polygons have straight edges and vertices with any real coordinates (not restricted to a given pixel grid, and so are independent of the initial image resolution);
- The resulting coloured mesh (with the best constant colour for each polygon) approximates the original image, for example by minimising an energy.

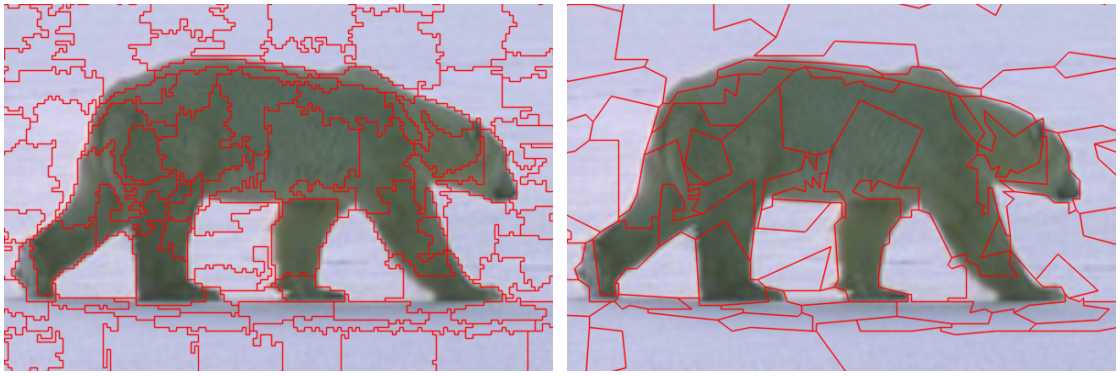

Figure A.1: SLIC superpixels (left) with zigzag boundaries of pixel-based superpixels are converted into a resolution-independent mesh (right) of polygons with straight edges that can be rendered at any higher resolution for better and smoother animations.

In the paper, we introduce RIMe (a Resolution-Independent Mesh of polygons): a fast conversion algorithm of any traditional set of pixel-based superpixels into resolution-independent superpixels, guaranteeing that their edges do not intersect, see Figure A.1. The resolution-independent meshes based on the superpixels SEEDS (Superpixels Extracted via Energy-Driven Sampling) [8] and SLIC (Simple Linear Iterative Clustering) [2] are compared with past meshes based on the Line Segment Detector [27]. The experiments on the Berkeley Segmentation Database (BSD) [41] confirm that the new superpixels have more compact shapes than pixel-based superpixels.

The main contributions of the paper to computer vision are as follows:

- We introduce the algorithm RIMe that can convert any set of pixel-based superpixels into a resolution-independent mesh with quality guarantees.
- Experimental analysis confirms that the resolution-independent meshes based on SEEDS and SLIC achieve better results on objective measures and perform similarly to SEEDS and SLIC on the BSD benchmarks.
- RIMe outperforms all other resolution-independent superpixels on the objective reconstruction error and benchmarks of BSD.

Appendix B

Notations

We record here a list of all the notations used in the thesis.

- General notation:
  - Dimension: n.
  - Point: p (Notation 1.1).
  - Vector: $\vec{v}$ (Notation 1.1).
  - Summations: i.
  - Coefficient: c.
  - Index set: I.
  - Transformation: T.
  - Bijection: γ or ψ.
  - Symmetric difference: $\ominus$.
- Introduction:
  - Lattice: Λ (Definition 1.2).
  - Lattice point: v.
  - Unit cell: U (Definition 1.2).
  - Motif: M (Definition 1.4).
  - Cardinality of motif M: m (Definition 1.4).
  - Point in a motif: a.
  - Periodic point set: A (Definition 1.5).
  - A second periodic point set: Q.

- k-extended unit cell: kU (Definition 1.6).
- Packing radius: r (Definition 1.11).
- Covering radius: R (Definition 1.12).
- Metric space: (M, d) or simply M (Definition 1.14).
- Point cloud: C (Definition 1.15).
- Subspace of a metric space: X.
- Graph: G (Definition 1.17).
- Neighbourhood graph: $N(C; \epsilon)$ (Definition 1.17).
- Edge: e.
- n-dimensional simplex: Δ^n (Definition 1.19).
- Simplicial complex: Q (Definition 1.19).
- Čech complex: $\check{\mathrm{C}}\mathrm{h}(C, M; \alpha)$ (Definition 1.20).
- Vietoris-Rips complex: $\mathrm{VR}(C, M; \alpha)$ (Definition 1.20).
- Delone triangulation: $\mathrm{Del}(C)$ (Definition 1.21).
- α-complex: $C(\alpha)$ (Definition 1.21).
- α-offset of a subspace $X \subset M$: X^α (Definition 1.22).
- Filtration: $\{Q(C; \alpha)\}$ (Definition 1.24).
- Filtration of α-offsets: $\{C^\alpha\}$.
- Homology:
  * Vector space of cycles: Z_1 (Definition 1.25).
  * Vector space of boundaries: B_1 (Definition 1.25).
  * k-th homology group: H_k (Definition 1.25).

- Voronoi Zones:
  - Voronoi domain of $p \in C$: $V(C; p)$ (Definition 2.1).
  - k-th degree Voronoi domain of $p \in C$: $V_k(C; p)$ (Definition 2.2).
  - k-th Voronoi zone of $p \in C$: $Z_k(C; p)$ (Definition 2.4).
  - Zone index: $\mathrm{ind}(x; C; p)$ (Definition 2.5).
  - Set of bisectors: $b(C; p)$ (Definition 2.5).
  - k-th Voronoi subdomain of a periodic set A: $V^{(k)}(A; 0)$ (Definition 2.7).
  - k-th Voronoi subzone of a periodic set A: $Z^{(k)}(A; a)$ (Definition 2.8).
  - Half-open Voronoi domain: $V^h(\Lambda; 0)$ (Definition 2.9).

- Piecewise shift: $f_k(x)$ (Definition 2.10).
- Integrable function: $\mu(x)$.
- Lattice points on the boundary of $2iU$: Λ_i (Lemma 2.16).
- Bisecting line between $0 \in \mathbb{R}^2$ and $p \in \mathbb{R}^2$: $L(p)$.
- Binary tree: T.

- Density Functions:
  - Set of balls of radius t centred at points of a set C: $B(C;t)$ (Notation 3.1).
  - k-th density function: $\psi_k^A(t)$ (Definition 3.2).
  - Density fingerprint: $\Psi(A)$ (Definition 3.5).
  - Lipschitz coefficient: C (Definition 3.6).
  - Bottleneck distance between periodic sets A and Q: $d_B(A,Q)$ (Definition 3.7).
  - Shorthand for bottleneck distance $d_B(A,Q)$: δ.
  - d_∞ distance between fingerprints of A and Q: $d_\infty(\Psi(A),\Psi(Q))$ (Definition 3.8).
  - Union of points covered by at least k balls of $B(C;t)$: $\cup^k B(C;t)$.
  - Union of points covered by exactly k balls of $B(C;t)$: C_t^k.
  - Intensity $|M|/\operatorname{Vol}[U]$: ρ.
  - List of simplices in A up to threshold θ: $L(A;\theta)$.
  - Largest finite circumradius of up to four points: $\operatorname{Rad}(A)$ (Definition 3.13).
  - Diameter of a unit cell: D.
  - Isometry class of A: $[A]$.
  - Fractional volume of a unit cell covered by at least k balls: $\varphi_k^A(t)$ (Definition 3.20).
  - Half-space: H (Definition 3.23).
  - Opposite half-space: $\overline{H}$ (Definition 3.23).
  - Plane: pl.
  - Spherical cap: $S_{\text{cap}}(H)$ (Definition 3.24).
  - Height of a spherical cap: h (Definition 3.24).
  - Spherical wedge: $S_{\text{wedge}}(H_1,H_2)$ (Definition 3.25).
  - Regularised spherical wedge: $S_{\text{rwedge}}(H_1,H_2)$ (Definition 3.26).
  - Spherical cone: $S_{\text{cone}}(H_1,H_2,H_3)$ (Definition 3.27).
  - Tetrahedron spanned by the points p_1,p_2,p_3,p_4: $\operatorname{Tet}(p_1,p_2,p_3,p_4)$.

- Skeletonisation Algorithms:
  - Level set: $L_t(f)$ (Definition 4.1).
  - Parameter space: Y.
  - Region of a covering: I.
  - Covering of intervals: $\mathcal{I}$.
  - DBSCAN's radius parameter: ϵ.
  - DBSCAN's minimum number of points: minPts.
  - Reeb graph: $\mathrm{Reeb}(Q, f)$ (Definition 4.2).
  - Minimum spanning tree of a filtration on C: $\mathrm{MST}(C)$ (Definition 4.4).
  - Forests contained in $MST(C)$: $\mathrm{MST}(C; \alpha)$ (Definition 4.4).
  - Persistent Homology:
    * Homology class: γ.
    * Birth value: $\mathrm{birth}(\gamma)$ (Definition 4.7).
    * Death value: $\mathrm{death}(\gamma)$ (Definition 4.7).
    * Persistence diagram: $\mathrm{PD}\{Q(C; \alpha)\}$ (Definition 4.8).
    * Multiplicity of a persistence dot: $u_{i,j}$ (Definition 4.8).
    * Cycle: L.
  - Homologically persistent skeleton: $\mathrm{HoPeS}(C)$ (Definition 4.13).
  - Reduced HoPeS: $\mathrm{HoPeS}(C; \alpha)$ (Definition 4.13).
  - Diagonal gap: $\mathrm{dgap}_k(C)$ (Definition 4.22).
  - Diagonal subdiagram: $\mathrm{DS}_k(C)$ (Definition 4.22).
  - Diagonal scale: $\mathrm{ds}_k(C)$ (Definition 4.22).
  - Vertical gap: $\mathrm{vgap}_{k,l}(C)$ (Definition 4.23).
  - Vertical subdiagram: $\mathrm{VS}_{k,l}(C)$ (Definition 4.23).
  - Vertical scale: $\mathrm{vs}_{k,l}(C)$ (Definition 4.23).
  - Derived HoPeS: $\mathrm{HoPeS}_{k,l}(C)$ (Definition 4.24).
  - Radius of a cycle: ρ (Definition 4.26).
  - Thickness of a graph: θ (Definition 4.26).
  - Simplified $\mathrm{HoPeS}(C)$: $\mathrm{simHoPeS}(C; \epsilon)$.
  - Skeleton: S.
  - k-wheel graph: $\mathrm{W}(k)$.
  - (k, l)-grid graph: $\mathrm{G}(k, l)$.
  - k-hexagons graph: $\mathrm{H}(k)$.