

# Merge Trees of Periodic Filtrations

Herbert Edelsbrunner  

ISTA (Institute of Science and Technology Austria), Klosterneuburg, Austria

Teresa Heiss  

ISTA (Institute of Science and Technology Austria), Klosterneuburg, Austria

---

## Abstract

Motivated by applications to crystalline materials, we generalize the merge tree and the related barcode of a filtered complex to the periodic setting in Euclidean space. They are invariant under isometries, changing bases, and indeed changing lattices. In addition, we prove stability under perturbations and provide an algorithm that under mild geometric conditions typically satisfied by crystalline materials takes  $\mathcal{O}((n + m) \log n)$  time, in which  $n$  and  $m$  are the numbers of vertices and edges in the quotient complex, respectively.

**2012 ACM Subject Classification** Theory of computation → Computational geometry

**Keywords and phrases** lattices, periodic sets, merge tree, persistent homology, union-find data structure, Euclid's gcd algorithm, interleaving distance, quotient pseudo-metric, 1-Wasserstein distance, optimal transport.

**Digital Object Identifier** 10.4230/LIPIcs...

**Funding** Both authors are partially supported by the European Research Council (ERC) Horizon 2020 project 'Alpha Shape Theory Extended', grant no. 788183. The first author is also partially supported by the DFG Collaborative Research Center TRR 109, 'Discretization in Geometry and Dynamics', Austrian Science Fund (FWF), grant no. I 02979-N35.

## 1 Introduction

The main objective of this paper is the extension of persistent homology to periodic data in Euclidean space. Such data frequently arises in material science, in particular the study of crystalline materials, and we refer to Fedorov [13] for an early attempt to classify such structures. Modeling the atoms of a crystalline material as points leads to the mathematical concept of a *periodic set*, which we define as the Minkowski sum of a finite set and a lattice; see e.g. [4, 32]. We study the case in which the periodic data is given within a finite parallelepiped; that is: the *unit cell* of the lattice.

Persistent homology describes the connected components and holes of shapes that are filtered, meaning their parts appear at different times. If this shape is a complex—perhaps the Delaunay triangulation of a point set—there are various ways to derive a filter, including the radius function on the Delaunay triangulation; see e.g. [11, Chapter III.4]. Taking the persistence diagram of this function, we get a summary of the topological connectivity on all scales. In the last two decades, this pipeline of going from the data to a filtered complex and further to the persistence diagram has become a popular tool in the topological approach to data analysis; see e.g. [3, 31]. While periodic data is abundant and would benefit from such an analysis, persistent homology has not yet been properly extended to periodic sets. This paper takes a step in this direction, by generalizing the 0-dimensional aspects of persistent homology to the periodic setting. We list the specific contributions of this paper:

- the introduction of the *shadow monomial* as the crucial ingredient to generalize the merge tree and the related 0-th barcode or persistence diagram to the periodic setting;
- an algorithm that under usually satisfied conditions constructs the periodic merge tree and 0-th barcode in  $\mathcal{O}((n + m) \log n)$  time, with  $n$  and  $m$  the number of vertices and edges in a unit cell;
- the proof of invariance under changing the underlying lattice, both for periodic 0-th barcodes and for the classes of periodic merge trees defined by the splintering equivalence relation;



© Herbert Edelsbrunner, Teresa Heiss;

licensed under Creative Commons License CC-BY 4.0

Leibniz International Proceedings in Informatics

LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

## XX:2 Merge Trees of Periodic Filtrations

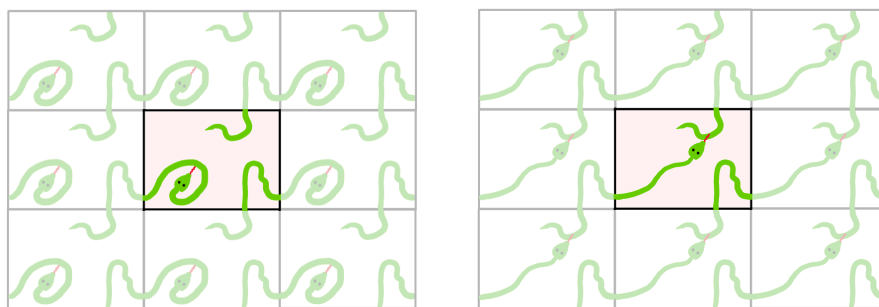
- the extension of the Wasserstein distance between barcodes and the interleaving distance between merge trees to the periodic setting, and proofs of stability for both extensions.

Our method can be applied to data in any dimension that can be viewed as a periodic graph with a periodic filter on it. Similar to classical persistent homology, this includes but is not limited to

- *locally finite periodic sets*, which generate filtered complexes (e.g. possibly weighted Delaunay) by growing balls centered at the points and monitoring when and where they overlap;
- *periodic digital images*, in which the grayscale or color values in the pixels provide a filter, and the graph could either be the edge-skeleton of the cubical complex, or its dual;
- *periodic functions* (e.g. piecewise linear or smooth), for which the graph structure and filter are induced by the critical points and their values at which the sublevel set alters its topology.

Even aperiodic materials or the aperiodic universe are often simulated with enforced periodic boundary conditions to prevent boundary effects from contaminating the results gathered within the bulk.

Prior approaches to persistent homology for periodic data relied primarily on one of two heuristics: work with a sufficiently large finite subset of the data, e.g. [16, 20], or compactify using the torus topology, which is a classic mathematical concept and supported by modern software, including the implementations of Delaunay triangulations in CGAL [33]. The first heuristic suffers from unwanted boundary effects and unnecessarily high computation costs, while the latter loses information about the patterns in the periodic setting. Our approach is related to both and reaps the benefits of the compactness of the torus while carefully recording the relation to the infinite periodic setting; see Figure 1. Indeed, each connected component on the torus has preimages (shadows) in the Euclidean space—perhaps infinitely many—and similar to Onus and Robins [24], we count these preimages but in a fine-grained manner that keeps track of growth-rates as well as densities (exponents and coefficients of the shadow monomials). The growth-rate distinguishes between connected clumps, strings, sheets, and blocks, while the density quantifies how densely these are distributed. Both pieces of information are needed for detailed statements about how materials are composed. The periodicity lattice—which is used in the definition of shadow monomials—has been introduced before under different names, and used to compute 0-homology [6, 9, 24]. However, using it to quantify the growth-rate and density of the components is to the best knowledge of the authors new.



■ Figure 1: In a unit cell with periodic boundary conditions (the torus), we see a single snake that bites itself, both in the *left* and the *right* panel. There is however a significant difference in the periodically tiled plane, since the snakes on the *right* connect in infinite diagonal lines, while the snakes on the *left* remain isolated, a distinction we will quantify with the novel concept of a *shadow monomial*. The material properties of the two examples would indeed be rather different, with higher resistance to tearing on the *right*.

Shifting the focus from counting components to recording when components merge, we mention the work of Ingrid Hotz and her group [30]. In collaborations with domain scientists, they study the electron density of 3-dimensional layered materials, and found that the timing of the merge events is useful in the quick comparison of materials. They compromise the (infinite) periodic setting by

considering merge events within an array of  $2 \times 2 \times 2$  unit cells with periodic boundary conditions. Our method removes the restriction to a finite portion of the data and provides extra information, such as how many and which components join at a merge event.

**Outline.** Section 2 extends the concept of a merge tree to the periodic setting. Section 3 explains the algorithm for constructing periodic merge trees. Section 4 proves the invariance and stability of equivalence classes of periodic merge trees. Section 5 introduces the periodic 0-th barcode and proves invariance and stability. Section 6 illustrates the concepts using a 3-dimensional periodic graph as an example. Section 7 concludes this paper.

## 2 The Periodic Merge Tree

This paper is written for arbitrary but fixed dimension,  $d$ , with  $d = 3$  being the most important case for applications, including to crystalline materials. We limit ourselves to the discrete setting of complexes or graphs, and stress that the ideas also apply to smooth and piecewise smooth functions.

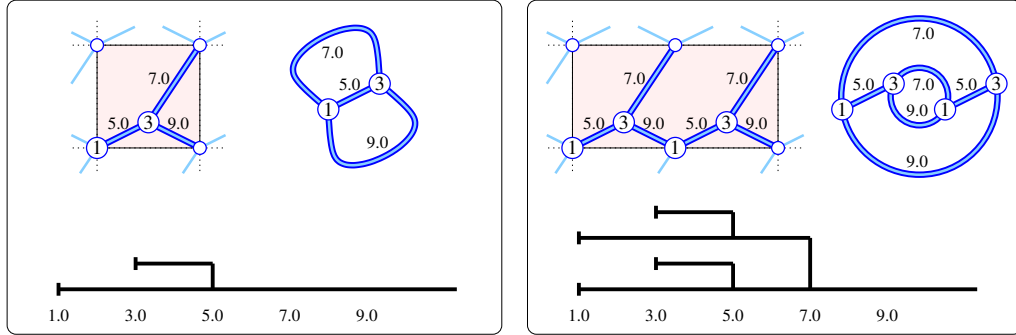
### 2.1 Merge Tree

We begin with standard definitions. A *cell complex*,  $K$ , in  $\mathbb{R}^d$  consists of points and cells that have these points as vertices. We require that each cell,  $\tau$ , be homeomorphic to the  $p$ -dimensional closed unit ball, for some  $0 \leq p \leq d$ , and the boundary of  $\tau$  is a union of lower-dimensional cells in the complex, called *faces* of  $\tau$ . In addition, we require that intersections of cells in the complex are either empty or a union of cells in their shared boundaries. Depending on the context, we treat  $K$  as a combinatorial object (a collection of cells with face relations), or as a geometric object (the set of points in the union of these cells). We allow for infinitely many points and cells but require that  $K$  be locally finite. The points are referred to as the *vertices*, and the 1-dimensional cells as the *edges* of  $K$ . The complex is *connected* if the graph that consists of its vertices and edges is connected. Since this paper is primarily concerned with 0-dimensional homology—whose classes correspond to the connected components—we will mostly ignore cells of dimension 2 and higher. After removing such cells, we are left with a graph consisting of vertices and edges, allowing for edges that start and end at the same vertex, and multiple edges connecting the same two vertices. However, in anticipation of an extension of our methods to higher homology dimensions, and because the overhead is modest, we nonetheless write this paper using the more general terminology of complexes.

A *filter* of  $K$  is a function  $F: K \rightarrow \mathbb{R}$  that satisfies  $F(\sigma) \leq F(\tau)$  whenever  $\sigma$  is a face of  $\tau$ . Given  $t \in \mathbb{R}$ , the corresponding *sublevel set* consists of all cells with value at most  $t$ , denoted  $K_t = F^{-1}(-\infty, t]$ . By definition of filter,  $K_t \subseteq K$  is itself a complex and thus a *subcomplex* of  $K$ . More generally,  $K_s$  is a subcomplex of  $K_t$ , whenever  $s \leq t$ , which implies that every connected component of  $K_s$  includes into a connected component of  $K_t$ . The ordered sequence of different sublevel sets is the *filtration* defined by  $F$ . Think of the sublevel set as an object that evolves as the threshold grows continuously. With this picture in mind, the merge tree of  $F$  keeps track of the connected components and how they merge. Intuitively, this tree has a point for each connected component of each sublevel set. We draw the tree from left to right, nevertheless calling the threshold for the sublevel set the *height* of the point. As long as a connected component stays the same or just grows, we draw the point along a left-to-right trajectory, but when components merge, then the corresponding trajectories meet, and afterwards continue as a single trajectory again in left-to-right direction. Since trajectories meet but never bifurcate, and—assuming  $K$  is connected—eventually combine to a single trajectory, we indeed have a tree, as suggested by the name. The number of points in a vertical section of this tree is the number of connected components of the corresponding sublevel set, and the connectivity in the left-to-right direction reflects the inclusion between connected

## XX:4 Merge Trees of Periodic Filtrations

components in different sublevel sets; see Figure 2, which shows two merge trees of the same periodic set. Each tree is drawn as a collection of horizontal intervals, called *beams*, which we connect with short vertical line segments to represent the merging of two components.



■ Figure 2: *Left panel*: a periodic graph with two vertices and three edges inside a unit cell in the shape of the unit square in the *upper left portion*, and the corresponding graph in the 2-dimensional torus in the *upper right portion of the panel*. The filter maps the vertices to their (real) labels and the edges to the values shown, which defines the merge tree at the *bottom in the panel*. *Right panel*: the same periodic graph as in the *left panel*, but now represented by a sublattice with a rectangular unit cell of twice the area. Correspondingly, the graph in the 2-dimensional torus has twice the number of vertices and edges, and the merge tree is richer than in the *left panel*.

Note the asymmetry, in which we always draw the vertical segment from the shorter to the longer beam (comparing their lengths to the left of the bifurcation point). As a result, the shorter beam ends, and the longer beam continues without interruption to the right. While not being part of the definition of merge tree, this asymmetry will come handy when we transition from merge trees to barcodes in Section 5. We call this the *elder rule* according to which the younger component gets absorbed into the older component; see [11, page 150]. Ties are broken arbitrarily, and we will see in Section 5 that different ways to break a tie does not affect the barcode.

To give a more formal definition of the merge tree, we introduce the *sublevel space* of  $F$ , denoted  $[F]$ , which consists of all points  $(x, s) \in K \times \mathbb{R}$  with  $x \in K_s$ , in which  $x$  may be a vertex or a point on an edge or higher-dimensional cell in  $K$ . Call two of its points *equivalent*, written  $(x, s) \sim (y, t)$ , if  $s = t$  and  $x, y$  belong to the same connected component of  $K_s = K_t$ . The *quotient*, denoted  $[F]_{\sim}$ , effectively combines all equivalent points into one and inherits the *quotient topology* from the Euclidean topology of  $\mathbb{R}^{d+1}$  restricted to  $[F] \subseteq \mathbb{R}^{d+1}$ .

► **Definition 2.1 (Merge Tree).** *The merge tree of a filter  $F: K \rightarrow \mathbb{R}$  is the quotient of the sublevel space of  $F$ , denoted  $\mathbb{M} = \mathbb{M}(F) = [F]_{\sim}$ , together with the corresponding quotient topology. It is equipped with the height function,  $h: \mathbb{M} \rightarrow \mathbb{R}$ , which maps each point to the value of the sublevel set in which the point is a connected component.*

Whenever we talk about a merge tree, we will tacitly assume that it comes with a height function. We will furthermore simplify language by saying that points  $\alpha$  and  $\beta$  of the merge tree not only represent but in fact *are* connected components of their respective sublevel sets, namely of  $K_{h(\alpha)}$  and  $K_{h(\beta)}$ , respectively. We say  $\beta$  *covers*  $\alpha$  if  $h(\alpha) \leq h(\beta)$  and  $\alpha \subseteq \beta$ .

## 2.2 Periodic Setting

We are specifically interested in filters that are periodically repeated on a complex that consists of periodically arranged copies of a finite complex. The natural language for such a setting is that of lattices in Euclidean space, discussed e.g. in [4, 32]. Let  $u_1, u_2, \dots, u_p$  be linearly independent

vectors in  $\mathbb{R}^d$ . The set of all integer combinations is the *lattice* spanned by these vectors:

$$\Lambda = \Lambda(u_1, u_2, \dots, u_p) = \{\lambda_1 u_1 + \lambda_2 u_2 + \dots + \lambda_p u_p \mid \lambda_i \in \mathbb{Z} \text{ for } 1 \leq i \leq p\}, \quad (1)$$

and we call  $p = \dim \Lambda$  the *dimension* of the lattice. The *span* of the vectors, denoted  $\text{span}(\Lambda)$ , is the set of all real combinations of the vectors, which is a  $p$ -dimensional linear subspace of  $\mathbb{R}^d$ . Similarly, the *unit cell* of the vectors is the set of real combinations with coefficients in  $[0, 1)$ . The same lattice can be spanned by different sets of vectors, but the dimension, the span, and the ( $p$ -dimensional) volume of the unit cell are always the same. We write  $\text{vol}_p \Lambda$  for the latter and note that for  $p = d$ , it is the determinant of the matrix whose columns are the vectors. Assuming  $\dim \Lambda = d$ , we write  $U$  for the  $d \times d$  matrix whose columns are the basis vectors. Since  $U^{-1}U$  is the identity matrix,  $U^{-1}$  maps  $\Lambda$  to the standard integer lattice,  $\mathbb{Z}^d$ , and  $U^{-1}$  applied to a lattice vector gives the integer coordinates of that vector with respect to the basis given by  $U$ .

Given a  $d$ -dimensional lattice  $\Lambda$ , a locally finite complex,  $K \subseteq \mathbb{R}^d$ , is  $\Lambda$ -*periodic* if  $\sigma \in K$  and  $u \in \Lambda$  implies  $\sigma + u \in K$ . Similarly, a filter on a  $\Lambda$ -periodic complex  $K$  is a  $\Lambda$ -*periodic* filter,  $F: K \rightarrow \mathbb{R}$ , if it satisfies  $F(\sigma) = F(\sigma + u)$  for every  $\sigma \in K$  and  $u \in \Lambda$ . Periodicity means that information is repeated infinitely often. To avoid infinitely large merge trees, it can therefore be useful to reduce redundancy by building quotients. The *quotient*,  $\mathbb{R}^d/\Lambda$ , is obtained by identifying points  $x, y \in \mathbb{R}^d$  whenever  $y - x \in \Lambda$ . This quotient with topology inherited from the  $d$ -dimensional Euclidean space is usually referred to as the  *$d$ -dimensional torus*. Similarly, we can identify cells of  $K$  if they are translates of each other by a vector of  $\Lambda$ . The resulting *quotient complex*,  $K/\Lambda$ , is a complex on the  $d$ -torus consisting of finitely many cells because  $K$  is locally finite and the  $d$ -torus is compact.<sup>1</sup> We write  $F/\Lambda: K/\Lambda \rightarrow \mathbb{R}$  for the *quotient filter*, which is a convenient representation of  $F: K \rightarrow \mathbb{R}$ . Unfortunately, this representation is not unique since, for example,  $K$  and  $F$  are also periodic with respect to  $2\Lambda$ , or really any sublattice of  $\Lambda$ ; compare the two panels in Figure 2 in which the respective unit cells (a square on the left and a rectangle of twice the area on the right) define the same periodic complex but yield different merge trees. This motivates the introduction of a quantified version of the merge tree; see Figure 4. We first explain how we quantify and then how we apply this idea to modify the merge tree.

### 2.3 Shadow Monomial

Let  $\Lambda \subseteq \mathbb{R}^d$  be a  $d$ -dimensional lattice, and  $F: K \rightarrow \mathbb{R}$  a  $\Lambda$ -periodic filter on a  $\Lambda$ -periodic complex. Write  $\phi: \mathbb{R}^d \rightarrow \mathbb{R}^d/\Lambda$  for the projection that maps every  $x$  to  $x + \Lambda$ . Taking the inverse, a sublevel set of the quotient filter lifts to the corresponding sublevel set of the filter:  $K_t = \phi^{-1}((K/\Lambda)_t)$ . Given a component  $\Gamma \subseteq (K/\Lambda)_t$ , we call a component  $\gamma$  of  $\phi^{-1}(\Gamma)$  a *shadow* of  $\Gamma$ . As an example, consider the sublevel set of the graph in Figure 2 before adding the edge with value 9.0. The quotient is a loop whose shadows are infinite polygonal lines that run parallel to each other in a diagonal direction, as shown in Figure 3. To count the distinct shadows of  $\Gamma$ , we introduce the *periodicity lattice* of all vectors that keep the shadows invariant:

$$\Lambda_\Gamma = \{u \in \Lambda \mid \gamma = \gamma + u \text{ for at least one and therefore every shadow } \gamma \text{ of } \Gamma\}. \quad (2)$$

For example, translating the shadow in Figure 3 along the vector  $(1, 1)$  maps the polygonal line to itself. All other vectors for which this holds are integer multiples of this vector. Hence, the periodicity lattice of the loop is the 1-dimensional lattice spanned by  $(1, 1)$ . With this, counting the shadows of  $\Gamma$

<sup>1</sup> Note, however, that even a periodic complex that is simplicial can have a non-simplicial quotient, which includes the possibilities of more than two edges connecting the same two vertices and an edge connecting a single vertex back to itself; see Figure 2 for an example.

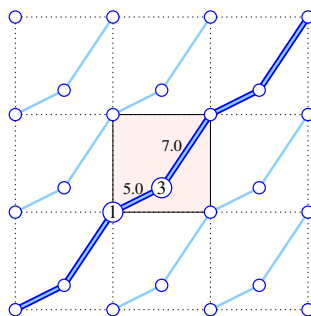
## XX:6 Merge Trees of Periodic Filtrations

reduces to counting the elements of  $\Lambda/\Lambda_\Gamma$ . If the cardinality of this quotient is finite, then this gives the number of shadows. On the other hand, if the cardinality is infinite, we wish to count the elements that appear within a spherical window of radius  $R$ . To this end, we introduce a monomial in the variable  $R$ , such that the coefficient gives the density and the exponent gives the growth-rate. Write  $\nu_q$  for the  $q$ -dimensional volume of the unit ball in  $\mathbb{R}^q$ , so  $\nu_0 = 1$ ,  $\nu_1 = 2$ ,  $\nu_2 = \pi$ ,  $\nu_3 = \frac{4\pi}{3}$ , etc.

► **Definition 2.2** (Shadow Monomial). *Let  $\Lambda \subseteq \mathbb{R}^d$  be a  $d$ -dimensional lattice, and  $F: K \rightarrow \mathbb{R}$  a  $\Lambda$ -periodic filter. Letting  $\Gamma \in \mathbb{M}(F/\Lambda)$  be a connected component of a sublevel set of the quotient filter, and  $p$  the dimension of the corresponding periodicity lattice,  $\Lambda_\Gamma$ , the shadow monomial of  $\Gamma$  is*

$$\mathfrak{m}_\Gamma(R) = \frac{\text{vol}_p \Lambda_\Gamma}{\text{vol}_d \Lambda} \cdot \nu_{d-p} R^{d-p}. \quad (3)$$

Up to lower-order terms, this monomial gives the number of elements of  $\Lambda/\Lambda_\Gamma$  that have points inside the sphere of radius  $R$  centered at the origin in  $\mathbb{R}^d$ . For example, the loop whose shadows are the polygonal lines in Figure 3, we have  $\text{vol}_1 \Lambda_\Gamma = \sqrt{2}$ ,  $\text{vol}_2 \Lambda = 1$ , and  $\nu_1 = 2$ , so  $\mathfrak{m}_\Gamma(R) = 2\sqrt{2}R^1$ , and up to a constant error term this is the number of polygonal lines that pass through a disk of radius  $R$ . This property is fundamental to our results, so we give a formal statement and a proof.



■ Figure 3: Each shadow of the loop in the quotient complex is an infinite polygonal line with periodicity lattice spanned by the vector  $(1, 1)$ . The length of its unit cell is  $\sqrt{2}$ , which implies that its shadow monomial is  $2\sqrt{2}R$ .

► **Lemma 2.3** (Counting Inside Sphere). *Let  $\Lambda \subseteq \mathbb{R}^d$  be a  $d$ -dimensional lattice and  $\Lambda_\Gamma \subseteq \Lambda$  a  $p$ -dimensional sublattice. Then the number of elements of  $\Lambda/\Lambda_\Gamma$  that have a non-empty intersection with the ball of radius  $R$  centered at  $0 \in \mathbb{R}^d$  is  $\mathfrak{m}_\Gamma(R) + \mathcal{O}(R^{d-p-1})$ .*

**Proof.** We begin with the two extreme cases. In the zero-dimensional case, when  $p = 0$ , we have  $\Lambda_\Gamma = \{0\}$  and  $\Lambda/\Lambda_\Gamma = \Lambda$ . We thus count the points of  $\Lambda$  inside the  $d$ -ball of radius  $R$  centered at 0, which up to lower-order terms is the volume of the ball divided by the volume of the unit cell:  $\nu_d R^d / \text{vol}_d \Lambda = \mathfrak{m}_\Gamma(R)$  because  $\text{vol}_0 \Lambda_\Gamma = 1$ . Observe that this estimate also holds if we count the points inside the  $d$ -ball with radius  $R - 1$ . In the full-dimensional case, when  $p = d$ , we have  $\dim \Lambda_\Gamma = \dim \Lambda$ , so  $\Lambda/\Lambda_\Gamma$  is finite, with cardinality  $\text{vol}_d \Lambda_\Gamma / \text{vol}_d \Lambda$ . Indeed, any unit cell of  $\Lambda_\Gamma$  contains a point from each class in the quotient, so for a large enough constant  $R$ , the  $d$ -ball with radius  $R$  contains a point from each class. Furthermore,  $\text{vol}_d \Lambda_\Gamma / \text{vol}_d \Lambda = \mathfrak{m}_\Gamma(R)$  because  $\nu_0 R^0 = 1$ .

In the general case, let  $\Lambda'$  be the maximal sublattice of  $\Lambda$  that satisfies  $\text{span}(\Lambda') = \text{span}(\Lambda_\Gamma)$ . By construction,  $p = \dim \Lambda' = \dim \Lambda_\Gamma$ , so we can apply the result for the full-dimensional case, in which we count exactly  $\text{vol}_p \Lambda_\Gamma / \text{vol}_p \Lambda'$  translates of  $\Lambda_\Gamma$  within  $\Lambda'$ . Next project  $\Lambda$  orthogonally onto the linear subspace orthogonal to  $\text{span}(\Lambda')$ , which gives a  $(d - p)$ -dimensional lattice,  $\Lambda''$ . The unit cells satisfy  $\text{vol}_p \Lambda' \cdot \text{vol}_{d-p} \Lambda'' = \text{vol}_d \Lambda$ . Using the result for the zero-dimensional case, we count  $\nu_{d-p} R^{d-p} / \text{vol}_{d-p} \Lambda'' + \mathcal{O}(R^{d-p-1})$  points of  $\Lambda''$  within the ball of radius  $R$ , and multiplying

the two counts while ignoring the lower order term, we get

$$\frac{\text{vol}_p \Lambda_\Gamma}{\text{vol}_p \Lambda'} \cdot \nu_{d-p} R^{d-p} \frac{1}{\text{vol}_{d-p} \Lambda''} = \frac{\text{vol}_p \Lambda_\Gamma}{\text{vol}_p \Lambda'} \frac{\text{vol}_p \Lambda'}{\text{vol}_d \Lambda} \cdot \nu_{d-p} R^{d-p} = \mathfrak{m}_\Gamma(R). \tag{4}$$

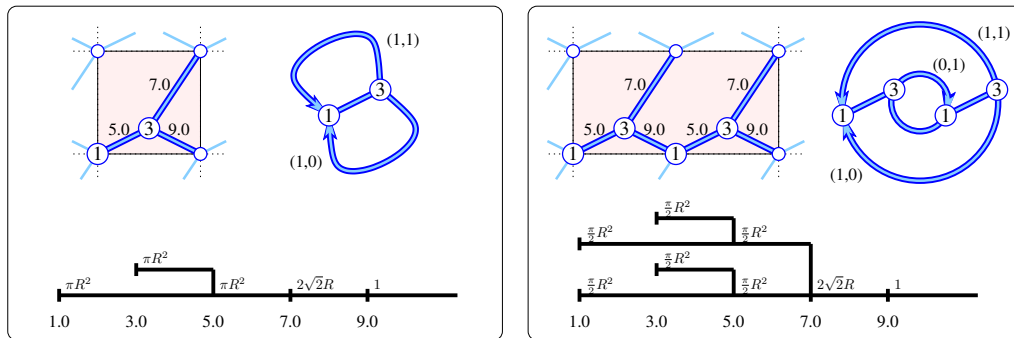
This expression measures the product of the  $(d - p)$ -ball with radius  $R$  and the  $p$ -ball with constant radius. This product is not contained in the  $d$ -ball of radius  $R$ , but if we shrink the  $(d - p)$ -ball to radius  $R - 1$ , then it is contained in the  $d$ -ball, assuming  $R$  is sufficiently large. This changes the count only by a lower-order term, which implies the claim in the general case. ◀

### 2.4 Periodic Merge Tree

To cope with the potentially infinitely many connected components of a periodic complex, we construct the merge tree for a finite representation in the torus and decorate this tree with a shadow monomial at every point. Each shadow monomial is specified by a real coefficient and an integer exponent, and we write  $\mathcal{M}[R]$  for the set of such monomials.

► **Definition 2.4** (Periodic Merge Tree). *Let  $\Lambda \subseteq \mathbb{R}^d$  be a  $d$ -dimensional lattice, and  $F: K \rightarrow \mathbb{R}$  a  $\Lambda$ -periodic filter. The periodic merge tree of  $F$  with respect to  $\Lambda$ , denoted  $\mathbb{M}(F, \Lambda)$ , is the merge tree of the quotient filter,  $\mathbb{M} = \mathbb{M}(F/\Lambda) = [F/\Lambda]_\sim$ , together with its height function,  $\mathfrak{h}: \mathbb{M} \rightarrow \mathbb{R}$ , and the frequency function,  $\mathfrak{f}: \mathbb{M} \rightarrow \mathcal{M}[R]$ , which maps each point  $\Gamma \in \mathbb{M}$  to its shadow monomial.*

Whenever we talk about a periodic merge tree, we will tacitly assume that it comes with its height and frequency functions. The notational ambiguity between merge trees and periodic merge trees is deliberate and emphasizes that every periodic merge tree is also a merge tree. See Figure 4 for two examples, in which we write the shadow monomials above the beams of the two periodic merge trees. For notational reasons, we write  $\mathfrak{m}_\Gamma(R) = \mathfrak{f}(\Gamma)(R)$  when a value of the frequency function is applied to a radius,  $R$ . Since connected components tend to grow with increasing threshold, it is plausible that the shadow monomial can only get smaller when it changes as we move from left to right in the merge tree. To make this intuition concrete, we write  $tR^{d-q} < sR^{d-p}$  if either  $d - q < d - p$  or  $d - q = d - p$  and  $t < s$ , and we write  $tR^{d-q} \leq sR^{d-p}$  if equality is allowed.



■ Figure 4: The same graphs and periodic merge trees as in Figure 2 but with additional information. Edges with non-zero shift vectors (to be defined in Section 3.2) are drawn as (directed) arcs and labeled with their shift vectors, while edges with zero shift vectors remain undirected and without vector. In addition to the appearances at  $t = 1.0, 3.0$  and the merger at  $t = 5.0$ , there are two catenations at  $t = 7.0, 9.0$  that define the shadow monomials decorating the beams of the periodic merge trees in the left panel. Note that the tree in the right panel has twice as many subtrees rooted at the point labeled 7.0, and that the shadow monomials compensate for the increased number of beams. Indeed, we have two events at each of the first four values defining the periodic merge tree, with a merger followed by a catenation at  $t = 7.0$ .

## XX:8 Merge Trees of Periodic Filtrations

► **Lemma 2.5 (Monotonicity).** *Let  $\Lambda \subseteq \mathbb{R}^d$  be a  $d$ -dimensional lattice,  $F: K \rightarrow \mathbb{R}$  a  $\Lambda$ -periodic filter, and  $\mathbb{M} = \mathbb{M}(F, \Lambda)$  the periodic merge tree of  $F$  with frequency function  $\mathfrak{f}: \mathbb{M} \rightarrow \mathcal{M}[R]$ . If a point  $B \in \mathbb{M}$  covers another point  $A \in \mathbb{M}$ , then  $\mathfrak{f}(B) \leq \mathfrak{f}(A)$ .*

**Proof.** Since  $B$  covers  $A$ , the periodicity lattice of  $A$  is a sublattice of that of  $B$ :  $\Lambda_A \subseteq \Lambda_B$ . Indeed, if  $\alpha = \alpha + u$  for every shadow  $\alpha$  of  $A$ , then  $\beta = \beta + u$  for every shadow  $\beta$  of  $B$ , but not necessarily the other way round. The exponents of  $\mathfrak{f}(A)$  and  $\mathfrak{f}(B)$  are  $d - \dim \Lambda_A$  and  $d - \dim \Lambda_B$ . If  $\dim \Lambda_A < \dim \Lambda_B$ , then  $\mathfrak{f}(B) < \mathfrak{f}(A)$ , and if  $\dim \Lambda_A = \dim \Lambda_B$ , then the coefficients determine the order, and the only difference between them is the volume of the periodicity lattice, which is at least as large for  $A$  as it is for  $B$ . We thus get  $\mathfrak{f}(B) \leq \mathfrak{f}(A)$  in general. ◀

Since the shadow monomial can only decrease along a left-to-right trajectory in the merge tree, its exponent progresses from  $d$  down to 0, possibly skipping some integers. We therefore define the  $(d - p)$ -ary era of a beam as the interval of points whose shadow monomials have exponent  $d - p$ . For example, the longest beams of the periodic merge trees in Figure 4 have three eras each, characterized by the exponent of  $R$ , which shrinks from left to right. We further subdivide each era into *epochs*, which are maximal intervals in which the coefficient of the shadow monomial stays constant. From one epoch to the next on the same beam, the periodicity lattice grows to a superlattice, which implies a strengthening of Lemma 2.5, namely that the coefficient drops by division with an integer larger than 1. The graph in Figure 4 is unfortunately too small to have eras consisting of more than one epoch, but see Section 6 for a more elaborate example in  $\mathbb{R}^3$ .

### 3 The Algorithm

This section presents the algorithm we use to construct the periodic merge tree of a  $\Lambda$ -periodic filter,  $F: K \rightarrow \mathbb{R}$ . This tree is but the merge tree of the quotient filter,  $F/\Lambda: K/\Lambda \rightarrow \mathbb{R}$ , equipped with the frequency function. The construction of the merge tree has been amply studied prior to this paper; see e.g. [28], so we will focus on the frequency function.

#### 3.1 Critical Events

Similar to Kruskal's classic algorithm for minimum spanning trees [21], we process the cells of  $K/\Lambda$  in the order of the filter and update the periodic merge tree at each step. The only relevant cells are the vertices and edges, and we distinguish the edges in the quotient complex whose endpoints lie in the same and in different connected components. In the former case, we consider a shadow of the edge and further distinguish when its endpoints belong to the same or to different shadows of the connected component. We therefore have three types of critical events that characterize the evolution of the connected components:

- *appearance*: a vertex is added, which starts a new connected component or, equivalently, a new beam of the periodic merge tree;
- *merger*: an edge with endpoints in two different connected components is added, which merges these components into one or, equivalently, ends a beam by joining it to another in the merge tree;
- *catenation*: an edge with both endpoints in the same connected component is added and the endpoints of its shadow belong to two different shadows of that component, which enlarges the periodicity lattice of the component and shrinks its shadow monomial;

see Figure 4, where the tree in the left panel experiences two appearances, one merger, and two catenations, while the tree in the right panel experiences four appearances, three mergers, and two catenations. To add a new vertex in a appearance, we set its periodicity lattice to  $\{0\}$ . To add a new edge,  $e$ , let  $A, B$  be the connected components that contain its endpoints. If  $A = B$ , then  $e$  together



with a path connecting its endpoints form a loop in the quotient complex, and the shadow of this loop can either be a loop in the periodic complex, or a path that connects different shadows of the same vertex. Hence, the addition of the edge has either no effect or it enlarges the periodicity lattice of the component and thus decreases the shadow monomial. Writing  $\Lambda_e$  for the periodicity lattice of the loop, and writing  $\Gamma$  for the component  $A$  after adding  $e$ , the periodicity lattice of  $\Gamma$  is the smallest common superlattice:  $\Lambda_\Gamma = \Lambda_A + \Lambda_e$ . On the other hand, if  $A \neq B$ , the addition of  $e$  merges the two components, and we write  $\Gamma$  for the new component, which is the union of  $A$ ,  $B$ , and  $e$ . The periodicity lattice of  $\Gamma$  is again the smallest common superlattice:  $\Lambda_\Gamma = \Lambda_A + \Lambda_B$ .

### 3.2 Periodicity Lattice and Loops

There is a direct relation between the periodicity lattice of a connected component and its loops. Since our primary concern are components (and not cycles of dimension 1 or higher), we may assume that a component,  $\Gamma$ , is an undirected graph. Nonetheless, we need to reason about loops in this graph, and this is done more conveniently in the representation of  $\Gamma$  as a directed graph, in which the (undirected) edge connecting  $x$  to  $y$  is replaced by two (directed) arcs, one from  $x$  to  $y$  and the other from  $y$  to  $x$ .

► **Definition 3.1** (Paths and Loops). *In a directed graph, a path from  $x$  to  $y$  is a sequence of arcs,  $a_i$  from  $x_i$  to  $x_{i+1}$ , for  $1 \leq i \leq k$ , with  $x_1 = x$  and  $x_{k+1} = y$ . This path is a loop if, in addition,  $x = y$ . A path or loop is simple if no arc is repeated.*

In our context, each arc is labeled with an integer vector that records the relative position of the unit cells where a trajectory that is a shadow of the arc starts and ends. Specifically, let  $a$  be an arc from  $x$  to  $y$ , write  $x_0, y_0$  for the shadows of the endpoints inside a common unit cell, and let  $x_0 + u, y_0 + w$  be the endpoints of a shadow of the arc. Then the *shift vector* of the arc is  $\text{Shift}(a) = U^{-1}(w - u)$ . Correspondingly, the shift vector of the arc from  $y$  to  $x$  is  $\text{Shift}(-a) = -\text{Shift}(a)$ . The *drift vector* of a path is the sum of shift vectors of its arcs, and similarly for a loop. Assuming the directed graph represents a connected undirected graph,  $\Gamma$ , we can designate one of its vertices as the *root* and extend every loop so it starts and ends at the root without changing its drift vector. Any two loops can therefore be concatenated while adding their drift vectors. The collection of drift vectors thus satisfies the properties of a lattice: for every vector we also have its integer multiples, and for every two vectors we also have their sum. Hence, this collection is isomorphic to a sublattice of  $\mathbb{Z}^d$ , and we write  $V \subseteq \mathbb{Z}^d$  for its basis. The periodicity lattice of the undirected graph,  $\Lambda_\Gamma$ , has basis  $UV$ .

EXAMPLE. Consider the right panel of Figure 4. The edges of the quotient graph with non-zero shift vectors are drawn directed to avoid possible ambiguities. The counterclockwise version of the *upper loop* consists of the edges with filter values 5.0, 7.0, 5.0, 7.0 and drift vector  $v_1 = (0, 0) + (0, 1) + (0, 0) + (1, 1) = (1, 2)$ , which means this loop moves one unit cells to the right and two unit cells up. The basis vectors that span the rectangular unit cell are  $u_1 = (2.0, 0.0)$  and  $u_2 = (0.0, 1.0)$ , so this drift vector corresponds to

$$U \cdot v_1 = \begin{bmatrix} 2.0 & 0.0 \\ 0.0 & 1.0 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 2 \end{bmatrix} = \begin{bmatrix} 2.0 \\ 2.0 \end{bmatrix}, \quad (5)$$

which is a vector in  $\Lambda$ . Its length is  $2\sqrt{2}$ , which explains the shadow monomial decorating the epoch that starts at  $t = 7.0$  and ends at  $t = 9.0$ . After adding the two edges with filter value 9.0, we get additional loops: the *inner loop* with drift vector  $v_2 = (0, 0) - (0, 1) = (0, -1)$ , and the *lower loop* with drift vector  $v_3 = (0, 0, ) + (0, 0) + (0, 0) - (1, 0) = (-1, 0)$ . All other loops are concatenations of these three loops, and thus their drift vectors are sums of these three drift vectors. Writing  $V$  for

## XX:10 Merge Trees of Periodic Filtrations

the matrix whose columns are  $v_1, v_2, v_3$ , the corresponding periodicity lattice is thus spanned by

$$U \cdot V = \begin{bmatrix} 2.0 & 0.0 \\ 0.0 & 1.0 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & -1 \\ 2 & -1 & 0 \end{bmatrix} = \begin{bmatrix} 2.0 & 0.0 & -2.0 \\ 2.0 & -1.0 & 0.0 \end{bmatrix}. \quad (6)$$

The last vector is redundant, so we may choose the first two as a basis. They span a unit cell of area 2.0, which is the same area as the unit cell of  $\Lambda$  (the shaded rectangle). This explains the shadow monomial decorating the last epoch in the periodic merge tree.

### 3.3 Implementing with Union-Find

For the computation of the periodicity lattice, it suffices to look at the drift vectors of a basis of the loop space. We work with a basis defined by a spanning tree of  $\Gamma$ : each edge not in this tree belongs to a unique simple cycle in which all other edges are taken from the spanning tree. Choosing a direction for this edge, we get a corresponding simple loop anchored at the root of  $\Gamma$ , and the basis consists of one such loop for each edge not in the spanning tree. It is convenient to use the spanning tree that consists of the edges corresponding to past merge events for this purpose. The construction of this tree can be reduced to a sequence of *find* and *union* operations—the former are used to decide whether the two endpoints of an edge belong to the same or to different components, and the latter merge the two components that contain the endpoints, assuming they are different.

To put the theory into practice, we use a data structure that supports constant time find and amortized logarithmic time union operations. Faster implementations of the *union-find* data type exist; see e.g. the survey in [14], but they are not required for constructing the periodic merge tree within the desired time bound. The data structure stores the vertices of  $\Gamma$  in a linked list whose first vertex is the root  $r$  of  $\Gamma$ . We write  $\text{Next}(x)$  for the successor of  $x$ , which is `null` if  $x$  is last. In addition,  $x$  stores a link to the root,  $\text{Root}(x) = r$ , and the drift vector of the unique simple path from  $r$  to  $x$  inside the spanning tree, denoted  $\text{Drift}(x)$ . Finally, the root stores the number of vertices in the component, denoted  $\text{Size}(r)$ , the vertex with smallest filter value in  $\Gamma$ , denoted  $\text{Old}(r)$ , as well as an integer matrix,  $\text{Basis}(r) = V$ , such that  $UV$  is a basis of the periodicity lattice of  $\Gamma$ .

Given an edge with endpoints  $x, y$ , the find operation returns their respective roots:  $r = \text{Root}(x)$  and  $s = \text{Root}(y)$ . The vertices belong to the same component iff  $r = s$ , so we can distinguish between the cases in which the edge forms a loop or connects two components in constant time. In the former case, we get a new path from  $r$  to  $y$ , possibly with a different drift vector, which requires updating the periodicity lattice. Let  $a$  be the directed version of the edge, leading from  $x$  to  $y$ :

```
01  if  $r = s$  then  $v = \text{Drift}(x) + \text{Shift}(a) - \text{Drift}(y)$ ;  
02       $\text{Basis}(r) = \text{REDUCE}(\text{Basis}(r), v)$   
03  endif.
```

In the latter case, we form the union of the two components. To do this efficiently, we pick the smaller component, update the information at all its vertices, and insert the updated list right after the root of the larger component. In addition, we compute the periodicity lattice of the union:

```
01  if  $r \neq s$  then assume  $\text{Size}(s) \leq \text{Size}(r)$ ;  
02       $\text{Old}(r) = \min\{\text{Old}(r), \text{Old}(s)\}$ ;  $\text{Basis}(r) = \text{REDUCE}(\text{Basis}(r), \text{Basis}(s))$ ;  
03       $v = \text{Drift}(x) + \text{Shift}(a) - \text{Drift}(y)$ ;  $z = s$ ;  
04      while  $z \neq \text{null}$  do  
05           $\text{Root}(z) = r$ ;  $\text{Drift}(z) = v + \text{Drift}(z)$ ;  $\text{last} = z$ ;  $z = \text{Next}(z)$   
06      endwhile;  
07       $\text{Size}(r) = \text{Size}(r) + \text{Size}(s)$ ;  $\text{Next}(\text{last}) = \text{Next}(r)$ ;  $\text{Next}(r) = s$   
08  endif.
```

The algorithm for updating the periodicity lattice, including its running time, will be discussed in the next subsection. The running time of all other operations is easily analyzed. The periodic merge tree of a  $\Lambda$ -periodic filter,  $F: K \rightarrow \mathbb{R}$ , is constructed incrementally, by adding one vertex or edge at a time, and each addition gives rise to at most two find operations and at most one union operation. Letting  $n$  be the number of vertices of  $K/\Lambda$ , there are  $n - 1$  union operations in total. Each vertex that receives a new drift vector ends up in a component at least twice the size of its old component. Hence, every vertex changes its root and drift vector at most  $\log_2 n$  times. Letting  $m$  be the number of edges of  $K/\Lambda$ , we observe that each find operation takes constant time, which implies that  $\mathcal{O}(n \log n + m)$  time suffices for this part of the algorithm.

### 3.4 Euclid's Algorithm for Lattices

Next, we explain how the bases of two periodicity lattices can be processed to give the basis of their sum. Given sublattices,  $\Lambda', \Lambda'' \subseteq \Lambda$ , the vectors in their bases span  $\Lambda' + \Lambda''$ , which is the smallest sublattice of  $\Lambda$  that contains  $\Lambda'$  and  $\Lambda''$ . Since there are possibly more vectors than necessary, we reduce the matrix of basis vectors to its *Hermite normal form*; see Schrijver [26] for general background. Working with the corresponding integer matrices, this form can be computed in time polynomial in the size of the matrix and its entries; see [18] for the historically first such algorithm. To be self-contained, we give a simple algorithm that reduces the matrix by repeated application of Euclid's algorithm for the greatest common divisor (gcd) of two integers; see also [15]. We focus on constant size matrices and sacrifice the polynomial running time for the simplicity of the algorithm.

Let  $M$  be a  $d \times c$  integer matrix, and interpret its columns as vectors that span a sublattice of  $\mathbb{Z}^d$ . The algorithm reduces  $M$  using only three types of column operations: multiply a column with  $-1$ , exchange two columns, and subtract one column from another. An easy but important observation is that these operations preserve the lattice spanned by the columns. We reduce  $M$  from left to right such that the non-zero columns of the resulting lower triangular matrix define a basis of the lattice:

```

01   $i = j = 1$ ;
02  while  $i \leq d$  do
03      if  $\exists \ell$  with  $j \leq \ell \leq c$  and  $M[i, \ell] \neq 0$  then assume  $M[i, j] > 0$ ;
04      for  $k = j + 1$  to  $c$  do assume  $M[i, k] \geq 0$ ;
05          while  $M[i, j] > 0$  and  $M[i, k] > 0$  do
06              subtract  $\lfloor M[i, j]/M[i, k] \rfloor$  times column  $k$  from column  $j$ ;
07              exchange columns  $j$  and  $k$ 
08          endwhile
09      endfor;
10      for  $k = 1$  to  $j - 1$  do
11          subtract  $\lfloor M[i, k]/M[i, j] \rfloor$  times column  $j$  from column  $k$ 
12      endfor;  $j = j + 1$ 
13  endif;  $i = i + 1$ 
14  endwhile.
```

The while-loop in lines 05 to 08 effectively computes the gcd of the  $i$ -th entries of columns  $j$  and  $k$ ; that is:  $M[i, j] = \gcd(M[i, j], M[i, k])$  and  $M[i, k] = 0$ . This is prepared by ascertaining  $M[i, j] > 0$  and  $M[i, k] \geq 0$  in lines 03 and 04 by the possible exchange of columns  $j$  and  $\ell$  and by possible multiplication with  $-1$ . After finishing the for-loop in lines 04 to 09, all entries that succeed  $M[i, j]$  in its row are zero, and after finishing the for-loop in lines 10 to 12, all entries that precede  $M[i, j]$  in its row are non-negative and smaller than  $M[i, j]$ . After running the algorithm, each column has strictly more leading zeros than the preceding column. Hence, there are at most  $d$

## XX:12 Merge Trees of Periodic Filtrations

non-zero columns left, and they form a basis of the lattice defined by  $M$ . Indeed, the matrix is in Hermite normal form, as defined in [22] for the not necessarily full-rank case.

The running time depends, among other things, on the entries in  $M$ . To study this dependence, we introduce the *magnitude* of  $M$  as the maximum absolute entry in the matrix, denoted  $\|M\|_\infty$ . Similarly, we talk about the magnitude of an integer vector and the magnitude of a basis of integer vectors. Setting  $N = \|M\|_\infty$ , we show that the algorithm reduces  $M$  in time at most logarithmic in  $N$ . Note however that there are faster variants of this algorithm; see e.g. [15, 29].

A key ingredient of the proof is Bézout's identity, which is a statement about Euclid's algorithm for computing the gcd of positive integers  $s \leq t$ . Specifically, it asserts that there are integers  $x$  and  $y$  such that  $xs + yt = \gcd(s, t)$  and  $|x|, |y| \leq t$ . These coefficients are computed by the *extended Euclidean algorithm* from the quotients that appear during the execution of Euclid's algorithm. The latter substitutes 0 and  $\gcd(s, t)$  for  $s$  and  $t$ , so it is not surprising that there are also integers  $x_0$  and  $y_0$  such that  $x_0s + y_0t = 0$  and  $|x_0|, |y_0| \leq t$ , and they can similarly be computed from the quotients in Euclid's algorithm. In summary, the actions of Euclid's algorithm are equivalent to substituting  $xs + yt$  and  $x_0s + y_0t$  for  $s$  and  $t$ , and the magnitude of the four coefficients does not exceed  $t$ .

► **Lemma 3.2** (Time for Reduction). *Given a lattice spanned by the columns of a  $d \times c$  matrix of integer entries with absolute values at most  $N$ , the reduction algorithm takes time at most  $\mathcal{O}(d^2 c^d \log(2N))$  to construct a basis of the lattice.*

**Proof.** The algorithm consists of four loops, of which the outer `while`-loop iterates at most  $d$  times, and the two `for`-loops together iterate  $c - 1$  times. To bound the number of iterations of the inner `while`-loop, let  $\mu_i$  be the maximum absolute entry that appears in row  $i$  at any time during the algorithm. We have  $\mu_1 \leq N$ , because the reduction of the first row does not increase its entries. However, it may increase entries in the other rows. To understand by how much, we consider the side effect of Euclid's algorithm computing the gcd of  $s = M[1, j]$  and  $t = M[1, k]$ , with  $s \leq t$  or  $t \leq s$ . It substitutes  $\gcd(s, t) = xs + yt$  for  $s$  in  $M[1, j]$  and  $0 = x_0s + y_0t$  for  $t$  in  $M[1, k]$ . Since the reduction works on columns, it also substitutes  $xM[2, j] + yM[2, k]$  for  $M[2, j]$  and  $x_0M[2, j] + y_0M[2, k]$  for  $M[2, k]$ . We have  $|x|, |y|, |x_0|, |y_0| \leq N$ , so the first gcd computation in row 1 increases the magnitude of row 2 from at most  $N$  to at most  $2N^2$ . Similarly, every additional gcd computation in row 1, increases the magnitude of row 2 by at most a factor  $2N$ , so after  $c - 1$  such computations, we have  $\mu_2 \leq N(2N)^{c-1} = \frac{1}{2}(2N)^c$ . We iterate this argument and get  $\mu_i \leq \frac{1}{2}(2\mu_{i-1})^c \leq \frac{1}{2}(2N)^{c^{i-1}}$  for  $2 \leq i \leq d$ . Taking the binary logarithm of the final and largest upper bound, we get

$$\log_2 \mu_d \leq \log_2 \left( \frac{1}{2}(2N)^{c^{d-1}} \right) \leq c^{d-1} \log_2(2N). \quad (7)$$

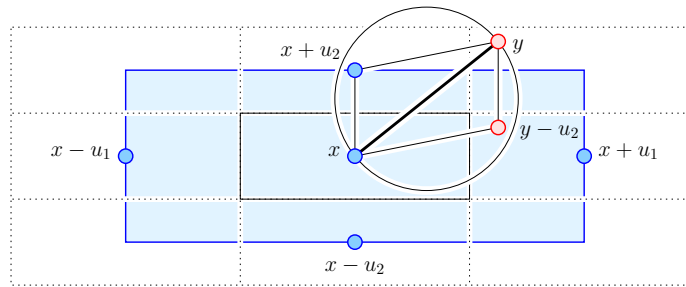
Since Euclid's algorithm takes at most logarithmically many steps to compute the gcd of two integers, this is an upper bound on the number of iterations of the inner `while`-loop. Multiplying with  $d(c - 1)$ , we get  $dc^d \log_2(2N)$  as an upper bound on the total number of iterations. Each iteration takes  $\mathcal{O}(d)$  time to subtract an integer multiple of a column from another, or to exchange two columns, which implies the claimed time bound. ◀

*Remark on magnitude.* While the above bound on the size of temporary entries in the matrix is rather high, it is known that the magnitude of the matrix in Hermite normal form is bounded from above by  $(\sqrt{d}N)^d$  [29, pages 91, 92]. Since the matrix in this form is unique [22, Theorem 1.5.2], it follows that this upper bound also applies to the reduced matrix computed by the above algorithm.

### 3.5 Magnitude and Time for Construction

The size of the entries in the matrices plays an important role in the running time for constructing a periodic merge tree. We thus express  $N$  in terms of the size of  $K/\Lambda$ . Considering a (directed) edge of the quotient complex,  $a$ , we recall that  $\text{Shift}(a)$  records the relative position of the endpoints of a shadow. The magnitude of this vector is its largest absolute entry, and we write  $D$  for the maximum magnitude over all shift vectors of edges in  $K/\Lambda$ , noting that it depends on the chosen basis of  $\Lambda$ .

It is not difficult to see that there are cases in which an unfortunate choice of basis can cause an arbitrarily large  $D$ , even independent of the number of vertices and edges in  $K/\Lambda$ . On the other hand, if  $K$  is a Delaunay triangulation and the basis vectors are pairwise orthogonal, then  $D \leq 1$ . This follows from [10, Theorem 3.1] but can also be proved directly, as illustrated in Figure 5. Let  $x$  and  $y$  be the endpoints of an edge in the Delaunay triangulation, and  $u_i$  a basis vector. The four points,  $x, x + u_i, y, y - u_i$  are the vertices of a parallelogram, and the angles at  $x + u_i$  and  $y - u_i$  are equal and at most  $90^\circ$ . Indeed, if they exceed  $90^\circ$ , then every  $(d - 1)$ -sphere that passes through  $x$  and  $y$  encloses at least one of  $x + u_i$  and  $y - u_i$ , which contradicts that  $x$  and  $y$  are the endpoints of an edge in the Delaunay triangulation. Since this holds for all basis vectors and their negatives,  $\pm u_i$  for  $1 \leq i \leq d$ , we conclude that  $y$  is contained in the hyper-rectangle with vertices  $x \pm u_1 \pm u_2 \pm \dots \pm u_d$ . But this hyper-rectangle overlaps only the unit cell that contains  $x$  and the  $3^d - 1$  neighboring unit cells. It follows that  $x$  and  $y$  lie in neighboring unit cells, so  $D \leq 1$ , as claimed.



■ Figure 5: The edge connecting  $x$  to  $y$  cannot be in the Delaunay triangulation if  $y$  lies outside the hyper-rectangle whose facets are centered at the points  $x \pm u_i$ . This hyper-rectangle has the volume of  $2^d$  unit cells and overlaps  $3^d$  of them, which for the displayed 2-dimensional case are drawn with *dotted lines*.

► **Lemma 3.3 (Magnitude of Basis).** *Let  $\Lambda \subseteq \mathbb{R}^d$  be a  $d$ -dimensional lattice with given matrix of basis vectors,  $U$ , let  $F: K \rightarrow \mathbb{R}$  be a  $\Lambda$ -periodic filter, let  $m$  be the number of edges in  $K/\Lambda$ , and let  $D$  be the maximum magnitude of their shift vectors. For every  $\Gamma \in \mathbb{M}(F, \Lambda)$ , let  $V$  be the Hermite normal form of the lattice  $U^{-1}(\Lambda_\Gamma) \subseteq \mathbb{Z}^d$ . Then, the basis,  $UV$ , of the periodicity lattice of  $\Gamma$  satisfies  $\|V\|_\infty \leq (\sqrt{d}Dm)^d$ .*

**Proof.** Take a spanning tree of  $\Gamma$ , and for each edge not in this tree, take the drift vector of the simple loop for which this edge is the sole edge not in the tree. There are at most  $m$  edges in this loop, so its drift vector has magnitude at most  $Dm$ . We get fewer than  $m$  such vectors, which define the lattice of the periodicity lattice of  $\Gamma$ . Letting  $M$  be the matrix whose columns are these drift vectors, we compute a basis by reducing  $M$  to Hermite normal form. Since  $\|M_\infty\| \leq Dm$ , this implies that the magnitude of the reduced matrix is at most  $(\sqrt{d}Dm)^d$ . ◀

The overall algorithm constructs a periodic merge tree by incrementally adding the vertices and edges of  $K/\Lambda$ . Whenever an edge connects two components (a merge event) or forms a loop (a possible catenation), we use the reduction algorithm to compute a basis from two bases. By Lemma 3.3, the magnitude of the input matrices is at most  $(\sqrt{d}Dm)^d$ . Indeed, as the Hermite normal

## XX:14 Merge Trees of Periodic Filtrations

form of a lattice is unique, the iteratively computed Hermite normal form from our algorithm agrees with the Hermite normal form from Lemma 3.3.

► **Theorem 3.4** (Time for Construction). *Let  $\Lambda \subseteq \mathbb{R}^d$  be a  $d$ -dimensional lattice with given basis,  $F: K \rightarrow \mathbb{R}$  a  $\Lambda$ -periodic filter,  $n$  and  $m$  the number of vertices and edges in the quotient complex,  $K/\Lambda$ , and  $D$  the maximum magnitude of the shift vectors. Then the periodic merge tree of  $F$  can be constructed in time  $\mathcal{O}(n \log n + m2^d d^{d+3} \log(\sqrt{d}Dm))$ .*

**Proof.** The union-find data structure is maintained in constant time per edge and amortized logarithmic time per vertex, so  $\mathcal{O}(n \log n + m)$  time in total. The reduction algorithm is used at most once for each edge, and each time reduces an integer matrix with  $d$  rows and at most  $2d$  columns. By Lemma 3.3, its magnitude is at most  $(\sqrt{d}Dm)^d$ , and by Lemma 3.2, the time to reduce the matrix is at most  $\mathcal{O}(d^2(2d)^d \log(2(\sqrt{d}Dm)^d))$ . Assuming  $(\sqrt{d}Dm)^d \geq 2$ , we can simplify the bound to  $\mathcal{O}(d^3(2d)^d \log(\sqrt{d}Dm))$ . After multiplying with the number of edges, we get  $\mathcal{O}(md^3(2d)^d \log(\sqrt{d}Dm))$  time in total, which implies the claimed bound. ◀

In the important application of our algorithm to crystalline materials, the dimension is a constant,  $d = 3$ . Using the filtration of alpha complexes in the torus, it is typical that each vertex (atom center) is connected to at most some constant number of other vertices, so the number of edges is  $m = \mathcal{O}(n)$ . Similarly typical is that entries in the shift vectors are bounded by a small constant,  $D$ . In this case, the algorithm constructs the periodic merge tree in time at most  $\mathcal{O}(n \log n)$ .

### 4 Properties

This section studies the invariance under different choices of lattices and the stability with respect to perturbations. We begin with the introduction of a notion of distance between periodic merge trees, and follow up with an equivalence relation among periodic merge trees and the corresponding quotient pseudo-metric.

#### 4.1 Interleaving Distance

We adapt the interleaving distance between ordinary merge trees introduced in [23] to periodic merge trees by including the frequency function, which maps each point of the tree to its shadow monomial. Recall that a point  $B \in \mathbb{M}$  covers  $A \in \mathbb{M}$  if  $\mathbf{h}(A) \leq \mathbf{h}(B)$  and  $A \subseteq B$ .

► **Definition 4.1** (Interleaving Distance). *Let  $\mathbb{M}$  and  $\mathbb{M}'$  be two periodic merge trees, with height functions  $\mathbf{h}: \mathbb{M} \rightarrow \mathbb{R}$ ,  $\mathbf{h}': \mathbb{M}' \rightarrow \mathbb{R}$  and frequency functions  $\mathbf{f}: \mathbb{M} \rightarrow \mathcal{M}[R]$ ,  $\mathbf{f}': \mathbb{M}' \rightarrow \mathcal{M}[R]$ . For  $\varepsilon \geq 0$ , continuous maps,  $\varphi: \mathbb{M} \rightarrow \mathbb{M}'$  and  $\psi: \mathbb{M}' \rightarrow \mathbb{M}$  are  $\varepsilon$ -compatible if for all  $\Gamma \in \mathbb{M}$ ,  $\Gamma' \in \mathbb{M}'$ ,*

- (i)  $\mathbf{h}' \circ \varphi(\Gamma) = \mathbf{h}(\Gamma) + \varepsilon$  and  $\mathbf{h} \circ \psi(\Gamma') = \mathbf{h}'(\Gamma') + \varepsilon$ ;
- (ii)  $\psi \circ \varphi(\Gamma)$  covers  $\Gamma$  and  $\varphi \circ \psi(\Gamma')$  covers  $\Gamma'$ ;
- (iii)  $\mathbf{f}' \circ \varphi(\Gamma) \leq \mathbf{f}(\Gamma)$  and  $\mathbf{f} \circ \psi(\Gamma') \leq \mathbf{f}'(\Gamma')$ .

*The interleaving distance between  $\mathbb{M}$  and  $\mathbb{M}'$ , denoted  $I(\mathbb{M}, \mathbb{M}')$ , is the infimum of the  $\varepsilon \geq 0$  for which there exist  $\varepsilon$ -compatible maps from  $\mathbb{M}$  to  $\mathbb{M}'$  and back.*

We focus on *finite* periodic merge trees, for which the infimum in Definition 4.1 is a minimum. Throughout this section, we will therefore assume finite trees, with the benefit of simpler arguments involving the interleaving distance.

As an example, consider the two periodic merge trees in Figure 4. Writing  $\mathbb{M}$  for the tree in the left panel and  $\mathbb{M}'$  for the tree in the right panel, we need continuous maps  $\varphi$  and  $\psi$  whose compositions increase the height of each point by  $2\varepsilon$  in its own tree (i.e. move the point to the right in our drawing).

For small  $\varepsilon$ , this is not possible because  $\mathbb{M}'$  has two subtrees rooted at the point with height 7.0, while  $\mathbb{M}$  has only one such subtree. The minimum height of any point is 1.0, and since  $\varepsilon$  is at least half the height difference, we get  $\varepsilon \geq 3.0$ . We get another constraint from Condition (iii): the shadow monomial cannot increase from a point to its image. All points with height less than 7.0 in  $\mathbb{M}$  have shadow monomials larger than those of the points with height 1.0 in  $\mathbb{M}'$ , hence  $\varepsilon \geq 6.0$ . Indeed, for  $\varepsilon = 6.0$  we get maps  $\varphi$  and  $\psi$  that are  $\varepsilon$ -compatible, which implies  $I(\mathbb{M}, \mathbb{M}') = 6.0$ .

We extend the proof of the interleaving distance being a metric for merge trees [23] to periodic merge trees, but note that [23] missed the verification of positivity, which we add in our extension.

► **Lemma 4.2** (Interleaving Distance is a Metric). *Let  $\mathbb{M}, \mathbb{M}', \mathbb{M}''$  be periodic merge trees. Then*

- $I(\mathbb{M}, \mathbb{M}') \geq 0$  and  $I(\mathbb{M}, \mathbb{M}') > 0$  iff  $\mathbb{M} \neq \mathbb{M}'$  (positivity);
- $I(\mathbb{M}, \mathbb{M}') = I(\mathbb{M}', \mathbb{M})$  (symmetry);
- $I(\mathbb{M}, \mathbb{M}') + I(\mathbb{M}', \mathbb{M}'') \geq I(\mathbb{M}, \mathbb{M}'')$  (triangle inequality).

**Proof.** *Positivity:* Clearly,  $I(\mathbb{M}, \mathbb{M}') = 0$  if  $\mathbb{M} = \mathbb{M}'$ . It remains to show  $I(\mathbb{M}, \mathbb{M}') > 0$  if  $\mathbb{M} \neq \mathbb{M}'$ . Recall that a complex in the  $d$ -dimensional torus is necessarily finite, so there are only finitely many critical events in the construction of the corresponding periodic merge tree. Let  $t_0 < t_1 < \dots < t_k$  be the values of the critical events that arise in the construction of  $\mathbb{M}$  and  $\mathbb{M}'$ . They decompose both trees into a finite collection of intervals, each connecting points at consecutive critical values. There is an adjacency preserving bijection between the two collections of intervals that preserves heights and shadow monomials iff  $\mathbb{M} = \mathbb{M}'$ . But if there is no such bijection, then there are also no  $\varepsilon$ -compatible maps for  $\varepsilon$  smaller than the minimum difference between consecutive critical values. Hence,  $I(\mathbb{M}, \mathbb{M}') > 0$ .

*Symmetry:* Since we can switch  $\varphi$  and  $\psi$  in Definition 4.1, we have  $I(\mathbb{M}, \mathbb{M}') \leq \varepsilon$  iff  $I(\mathbb{M}', \mathbb{M}) \leq \varepsilon$ , for every  $\varepsilon \geq 0$ . Hence,  $I(\mathbb{M}, \mathbb{M}') = I(\mathbb{M}', \mathbb{M})$ .

*Triangle inequality:* Writing  $\varepsilon = I(\mathbb{M}, \mathbb{M}')$  and  $\varepsilon' = I(\mathbb{M}', \mathbb{M}'')$ , there exist  $\varepsilon$ -compatible maps  $\varphi: \mathbb{M} \rightarrow \mathbb{M}'$ ,  $\psi: \mathbb{M}' \rightarrow \mathbb{M}$  and  $\varepsilon'$ -compatible maps  $\varphi': \mathbb{M}' \rightarrow \mathbb{M}''$ ,  $\psi': \mathbb{M}'' \rightarrow \mathbb{M}'$ . It is easy to see that  $\varphi' \circ \varphi: \mathbb{M} \rightarrow \mathbb{M}''$  and  $\psi \circ \psi': \mathbb{M}'' \rightarrow \mathbb{M}$  satisfy (i) and (iii) of Definition 4.1 for  $\varepsilon + \varepsilon'$ . To see (ii), we continuously advance  $\Gamma$  to the point  $\Pi \in \mathbb{M}$  at height  $h(\Pi) = h(\Gamma) + 2\varepsilon'$ . By continuity of  $\varphi$ , and because there is only one point of  $\mathbb{M}'$  whose height is  $h'(\varphi(\Gamma)) + 2\varepsilon'$  that covers  $\varphi(\Gamma)$ , this point must be  $\varphi(\Pi) = \psi' \circ \varphi' \circ \varphi(\Gamma)$ . Hence,  $\psi \circ \varphi(\Pi) = \psi \circ \psi' \circ \varphi' \circ \varphi(\Gamma)$ . Since  $\psi \circ \varphi(\Pi)$  covers  $\Pi$  and therefore  $\Gamma$ , this implies that  $\varphi' \circ \varphi$  and  $\psi \circ \psi'$  satisfy (ii) for  $\mathbb{M}$  and similarly for  $\mathbb{M}''$ , which implies  $I(\mathbb{M}, \mathbb{M}'') \leq \varepsilon + \varepsilon'$ . ◀

## 4.2 Splintering Periodic Merge Trees

While the periodic merge tree is independent of the choice of basis of a given lattice, it heavily depends on the lattice used for the filter; see Figure 4, where we get interleaving distance  $\varepsilon = 6.0$  just because we switch from the integer lattice on the left to a sublattice that drops every other column of integer points on the right. We desire a notion of distance that tolerates different lattices, provided the filter is periodic with respect to both. To this end, we introduce an equivalence relation and work with the corresponding quotient of the interleaving distance. For each point  $\Gamma \in \mathbb{M}$ , write  $\mathbb{M}_\Gamma$  for the subtree with topmost point  $\Gamma$ ; that is:  $\mathbb{M}_\Gamma$  consists of all points  $A \in \mathbb{M}$  covered by  $\Gamma$ .

► **Definition 4.3** (Equivalence by Splintering). *Let  $\mathbb{M}$  and  $\mathbb{M}'$  be two periodic merge trees with height functions  $h: \mathbb{M} \rightarrow \mathbb{R}$ ,  $h': \mathbb{M}' \rightarrow \mathbb{R}$  and frequency functions  $f: \mathbb{M} \rightarrow \mathcal{M}[R]$ ,  $f': \mathbb{M}' \rightarrow \mathcal{M}[R]$ . We say  $\mathbb{M}'$  splinters  $\mathbb{M}$  if there exists a continuous surjection  $\omega: \mathbb{M}' \rightarrow \mathbb{M}$  such that*

- (i) *the surjection preserves height:  $h \circ \omega = h'$ ;*

## XX:16 Merge Trees of Periodic Filtrations

(ii) *it splits subtrees evenly: for every  $\Gamma \in \mathbb{M}$  and all  $A, B \in \omega^{-1}(\Gamma)$ , the subtrees at  $A$  and  $B$  are equal,  $I(\mathbb{M}'_A, \mathbb{M}'_B) = 0$ ,  $\omega$  maps  $\mathbb{M}'_A$  as well as  $\mathbb{M}'_B$  surjectively onto  $\mathbb{M}_\Gamma$ , and the shadow monomials are  $\mathbf{f}'(A) = \mathbf{f}'(B) = \mathbf{f}(\Gamma)/\#\omega^{-1}(\Gamma)$ .*

We call two periodic merge trees equivalent, denoted  $\mathbb{M} \simeq \mathbb{N}$ , if there exists a periodic merge tree that splinters both, and we write  $[\mathbb{M}]$  for the class of periodic merge trees equivalent to  $\mathbb{M}$ .

To be certain that “ $\simeq$ ” is indeed an equivalence relation, we need to verify that it is transitive. First note that splintering is transitive: if  $\mathbb{M}'$  splinters  $\mathbb{M}$  and  $\mathbb{M}''$  splinters  $\mathbb{M}'$ , then  $\mathbb{M}''$  also splinters  $\mathbb{M}$ . Second observe that two splinterings of the same tree can be composed: if  $\mathbb{M}'$  and  $\mathbb{N}'$  both splinter  $\mathbb{N}$ , then there exists  $\mathbb{M}''$  that splinters both  $\mathbb{M}'$  and  $\mathbb{N}'$ . This operation acts like point-wise multiplication: if  $\Gamma \in \mathbb{N}$  has  $i$  preimages in  $\mathbb{M}'$  and  $j$  preimages in  $\mathbb{N}'$ , then it has  $ij$  preimages in  $\mathbb{M}''$ . To see that this is well defined, we traverse the points of the two trees in parallel and in the order of decreasing height, making sure that the split is even whenever we encounter a point of bifurcation. With this we argue the transitivity of “ $\simeq$ ”: assuming  $\mathbb{M} \simeq \mathbb{N} \simeq \mathbb{O}$ , there exist trees  $\mathbb{M}'$  and  $\mathbb{N}'$  splintering  $\mathbb{M}$ ,  $\mathbb{N}$  and  $\mathbb{N}$ ,  $\mathbb{O}$ , respectively. Since  $\mathbb{M}'$  and  $\mathbb{N}'$  both splinter  $\mathbb{N}$ , there exists  $\mathbb{M}''$  that splinters  $\mathbb{M}', \mathbb{N}'$ . By transitivity of splintering,  $\mathbb{M}''$  also splinters  $\mathbb{M}$  as well as  $\mathbb{O}$ , so  $\mathbb{M} \simeq \mathbb{O}$  as required.

Splintering happens, for example, when we construct the periodic merge tree of a  $\Lambda$ -periodic filter for a sublattice of  $\Lambda$ ; see Figure 4. By combining Definitions 4.1 and 4.3, we obtain a pseudo-distance function that considers equivalent such trees the same. For this, we use the quotient pseudo-metric of a metric space with respect to an equivalence relation [2, Definition 3.1.12]. The proof that this yields a pseudo-metric is both standard and trivial.

► **Definition 4.4** (Interleaving Pseudo-distance). *Let  $[\mathbb{M}]$  and  $[\mathbb{M}']$  be two equivalence classes of periodic merge trees. The interleaving pseudo-distance between them is the infimum, over all sequences of equivalence classes and two trees per class, of the sum of interleaving distances:*

$$J([\mathbb{M}], [\mathbb{M}']) = \inf \left\{ \sum_{i=1}^k I(\mathbb{M}_i, \mathbb{M}'_i) \right\}, \quad (8)$$

in which  $\mathbb{M}_1 \in [\mathbb{M}]$ ,  $\mathbb{M}'_k \in [\mathbb{M}']$ , and  $\mathbb{M}'_i \simeq \mathbb{M}_{i+1}$  for all  $1 \leq i \leq k-1$ .

As an example, consider again the two periodic merge trees in Figure 4. Writing  $\mathbb{M}$  and  $\mathbb{M}'$  for the tree in the left and right panels, respectively, we recall that the interleaving distance is  $I(\mathbb{M}, \mathbb{M}') = 6.0$ . The interleaving pseudo-distance is however zero, because  $\mathbb{M}'$  splinters  $\mathbb{M}$  and thus they are in the same equivalence class. Indeed, we get  $\mathbb{M}'$  by duplicating the subtree of  $\mathbb{M}$  below the point at height 7.0 and assigning half the shadow monomial to each point of the two copies of the subtree.

### 4.3 Invariance of Equivalence Classes

As mentioned earlier, splintering occurs if we enlarge the unit cell by substituting a sublattice for the original lattice in the construction of the periodic merge tree. We formalize this claim and prove it.

► **Lemma 4.5** (Splintering from Sublattice). *Let  $\Lambda \subseteq \mathbb{R}^d$  be a  $d$ -dimensional lattice,  $F: K \rightarrow \mathbb{R}$  a  $\Lambda$ -periodic filter, and  $\Lambda' \subseteq \Lambda$  a  $d$ -dimensional sublattice. Then  $\mathbb{M}(F, \Lambda')$  splinters  $\mathbb{M}(F, \Lambda)$ .*

**Proof.** To show that  $\mathbb{M}' = \mathbb{M}(F, \Lambda')$  splinters  $\mathbb{M} = \mathbb{M}(F, \Lambda)$ , we study the projection  $\phi: \mathbb{R}^d/\Lambda' \rightarrow \mathbb{R}^d/\Lambda$  defined by mapping every point  $x + \Lambda'$  to  $x + \Lambda$ . Since  $\dim \Lambda' = \dim \Lambda$ , the degree of the map is finite, namely  $\deg \phi = \text{vol}_d \Lambda' / \text{vol}_d \Lambda$ , and the image of  $\phi$  covers every point in the  $d$ -dimensional torus  $\deg \phi$  times. The projection induces a surjection,  $\omega: \mathbb{M}' \rightarrow \mathbb{M}$ , that maps every point  $\Gamma' \in \mathbb{M}'$  to the point  $\Gamma = \omega(\Gamma') \in \mathbb{M}$  with  $\mathbf{h}(\Gamma) = \mathbf{h}(\Gamma')$  and  $\phi(\Gamma') = \Gamma$ , where we note that  $\Gamma$  and  $\Gamma'$  are connected components of sublevel sets. The continuity of  $\omega$  follows from the continuity of  $\phi$  and the



definition of quotient topology as the final topology with respect to the quotient map. By construction,  $\omega$  preserves height, so it satisfies Condition (i) of Definition 4.3.

To prove Condition (ii), we note that all preimages of  $\Gamma$  are translates of each other and thus have the same periodicity lattice and shadow monomials. Similarly, their subtrees are identical. To show that  $\omega$  maps subtrees surjectively to the appropriate subtrees, recall that  $\phi(\Gamma') = \Gamma$ . Hence, for every component  $A$  in the subtree at  $\Gamma$ , we have  $\phi|_{\Gamma'}^{-1}(A) \neq \emptyset$ , and hence at least one component at height  $\mathfrak{h}(A)$  in the subtree at  $\Gamma'$  that  $\omega$  maps to  $A$ . The only part of Condition (ii) left to prove is the shadow monomial,  $\mathfrak{f}(\Gamma')$ . To this end, note that the cardinality of the preimage,  $\#\omega^{-1}(\Gamma)$ , depends on  $\deg \phi$  but also on the periodicity lattices of  $\Gamma$  and  $\Gamma'$ . By construction, these two lattices have the same dimension,  $p$ . The number of times the projection of  $\Gamma'$  covers  $\Gamma$  is thus finite, namely  $\text{vol}_p \Lambda_{\Gamma'} / \text{vol}_p \Lambda_{\Gamma}$ . Therefore, out of the  $\deg \phi$  preimages of every point in  $\Gamma$ ,  $\text{vol}_p \Lambda_{\Gamma'} / \text{vol}_p \Lambda_{\Gamma}$  stem from the same component, so the number of different components is

$$\#\omega^{-1}(\Gamma) = \deg \phi \cdot \frac{\text{vol}_p \Lambda_{\Gamma}}{\text{vol}_p \Lambda_{\Gamma'}}. \quad (9)$$

This yields the following calculation for the shadow monomial:

$$\mathfrak{f}'(\Gamma') = \frac{\text{vol}_p \Lambda_{\Gamma'}}{\text{vol}_d \Lambda'} \cdot \nu_{d-p} R^{d-p} \quad (10)$$

$$= \frac{\text{vol}_p \Lambda_{\Gamma}}{\text{vol}_d \Lambda} \cdot \nu_{d-p} R^{d-p} / \left[ \frac{\text{vol}_d \Lambda'}{\text{vol}_d \Lambda} \cdot \frac{\text{vol}_p \Lambda_{\Gamma'}}{\text{vol}_p \Lambda_{\Gamma'}} \right] = \frac{\mathfrak{f}(\Gamma)}{\#\omega^{-1}(\Gamma)}, \quad (11)$$

in which the left-hand side of (11) is obtained using trivial substitutions, and the right-hand side follows because the expression in squared brackets is  $1/\#\omega^{-1}(\Gamma)$ . We conclude that  $\omega$  satisfies all conditions of Definition 4.3. ◀

This implies the first main property of periodic merge trees we set out to prove in this section: that their equivalence classes are invariant under different lattices and different bases of the same lattice used in the construction of the trees.

► **Theorem 4.6** (Invariance of Equivalence Classes of Periodic Merge Trees). *Let  $\Lambda \subseteq \mathbb{R}^d$  be a  $d$ -dimensional lattice,  $F: K \rightarrow \mathbb{R}$  a  $\Lambda$ -periodic filter, and  $\Lambda', \Lambda'' \subseteq \Lambda$  two  $d$ -dimensional sublattices. Then  $\mathbb{M}(F, \Lambda')$  and  $\mathbb{M}(F, \Lambda'')$  are equivalent and thus the interleaving pseudo-distance between their equivalence classes vanishes:  $J([\mathbb{M}(F, \Lambda')], [\mathbb{M}(F, \Lambda'')]) = 0$ .*

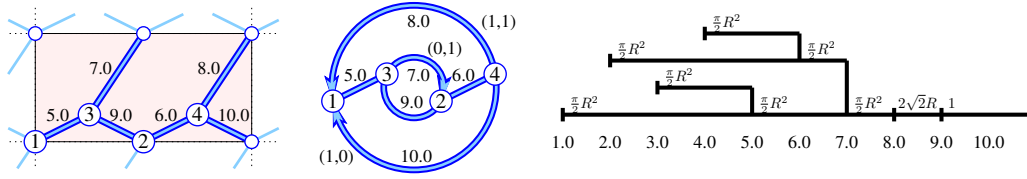
**Proof.** We show that the two trees are equivalent by finding a common splintering. For this, consider the lattice  $\Lambda''' = \Lambda' \cap \Lambda''$ . It is a sublattice of both  $\Lambda'$  and  $\Lambda''$ . By Lemma 4.5,  $\mathbb{M}(F, \Lambda''')$  splinters both,  $\mathbb{M}(F, \Lambda')$  and  $\mathbb{M}(F, \Lambda'')$ , which implies that the interleaving pseudo-distance vanishes. ◀

## 4.4 Stability of Periodic Merge Trees

The second property addressed in this section is the stability of periodic merge trees under perturbations of the filter. More specifically, we show that the interleaving pseudo-distance between the equivalence classes of two periodic merge trees is bounded from above by the  $L_\infty$ -distance between the two filters.

► **Theorem 4.7** (Stability of Periodic Merge Trees). *Let  $\Lambda \subseteq \mathbb{R}^d$  be a  $d$ -dimensional lattice, and  $F, G: K \rightarrow \mathbb{R}$  two  $\Lambda$ -periodic filters. Then  $I(\mathbb{M}(F, \Lambda), \mathbb{M}(G, \Lambda)) \leq \|F - G\|_\infty$  and hence  $J([\mathbb{M}(F, \Lambda)], [\mathbb{M}(G, \Lambda)]) \leq \|F - G\|_\infty$ .*

## XX:18 Merge Trees of Periodic Filtrations



■ Figure 6: From left to right: a perturbation of the periodic graph in the left panel of Figure 4, the corresponding quotient graph shown with filter values and shift vectors, and the resulting periodic merge tree. Its interleaving pseudo-distance to the periodic merge tree in the left panel of Figure 4 is 1.0. This can be seen by first splintering the tree in the left panel to obtain the tree in the right panel of Figure 4, and then comparing the latter to the periodic merge tree in this figure. The intermediate step is necessary because after the perturbation, the graph is no longer periodic with respect to the initially used square lattice.

**Proof.** Writing  $\mathbb{M} = \mathbb{M}(F, \Lambda)$  and  $\mathbb{M}' = \mathbb{M}(G, \Lambda)$  for the two periodic merge trees, we prove  $I(\mathbb{M}, \mathbb{M}') \leq \|F - G\|_\infty$ , and by definition of the interleaving pseudo-distance in terms of the interleaving distance, we have  $J([\mathbb{M}], [\mathbb{M}']) \leq I(\mathbb{M}, \mathbb{M}')$ , which then implies the claimed inequality. Write  $\varepsilon = \|F - G\|_\infty$ , and let  $A$  and  $A'$  be connected components of the sublevel sets of  $F/\Lambda$  and  $G/\Lambda$  at height  $h(A) = h'(A') = t$ , respectively. Since  $\varepsilon$  is the  $L_\infty$ -distance between  $F$  and  $G$ , there are unique connected components  $B$  and  $B'$  of the sublevel sets of  $F/\Lambda$  and  $G/\Lambda$  at height  $t + \varepsilon$ , respectively, such that  $A \subseteq B'$  and  $A' \subseteq B$ . By the stability theorem for merge trees in [23], the maps  $\varphi: \mathbb{M} \rightarrow \mathbb{M}'$  and  $\psi: \mathbb{M}' \rightarrow \mathbb{M}$  defined by  $\varphi(A) = B'$  and  $\psi(A') = B$  are  $\varepsilon$ -compatible as far as the two merge trees are concerned; that is: they satisfy Conditions (i) and (ii) of Definition 4.1.

It remains to prove Condition (iii). Since  $A \subseteq B'$ , an argument analogous to the proof of Lemma 2.5 yields  $f'(B') \leq f(A)$  and, by symmetry,  $f(B) \leq f'(A')$ . Therefore,  $\varphi$  and  $\psi$  are  $\varepsilon$ -compatible with respect to the two periodic merge trees, so  $I(\mathbb{M}, \mathbb{M}') \leq \varepsilon$  by definition of the interleaving distance between these trees. ◀

See Figure 6 for an illustration of the stability in which the perturbation forces a coarser lattice and thus an intermediate step of splintering. With the help of Theorem 4.6, the statement about the interleaving pseudo-distance in Theorem 4.7 can be extended to periodic filters  $F: K \rightarrow \mathbb{R}$  and  $G: L \rightarrow \mathbb{R}$ , provided  $K$  and  $L$  have the same underlying space and a common periodic refinement.

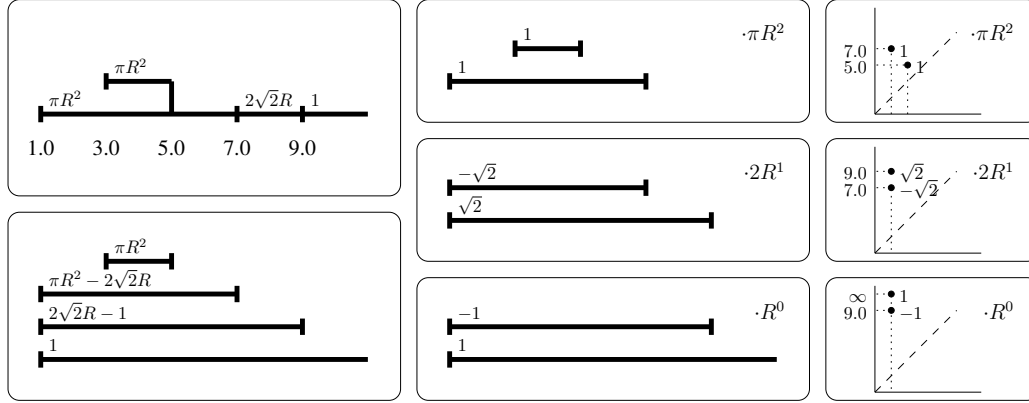
## 5 The Periodic Barcode

Motivated by simplifying the computation of distance, we strip information off the periodic merge tree to construct the periodic analog of the 0-th barcode or persistence diagram. Following the structure of the previous section, we introduce a new notion of distance right after defining the concept and prove invariance and stability thereafter.

### 5.1 Definition by Construction

In a nut-shell, we get the periodic 0-th barcode by calling the beams of the periodic merge tree bars while dropping the vertical segments that connect them. Recall, however, that different epochs along a beam have different shadow monomials, which decrease from left to right, and we need to account for them by possibly substituting more than one bar for each beam. Specifically, when the monomial decreases from  $s\nu_{d-p}R^{d-p}$  to  $t\nu_{d-q}R^{d-q}$  at some interior point of a beam, this point marks the end of an era, if  $d - q < d - p$ , and the end of an epoch inside an era, if  $d - q = d - p$ . This change is brought about by  $s\nu_{d-p}R^{d-p} - t\nu_{d-q}R^{d-q}$  deaths giving rise to polynomial decorations; see the step from the top left panel to the bottom left panel in Figure 7. We recover monomials by introducing two bars that end at this point, one decorated with  $s\nu_{d-p}R^{d-p}$  and the other with  $-t\nu_{d-q}R^{d-q}$ . This

enables the classification of the bars into  $d + 1$  eras, which is desirable since different growth-rates render the coefficients of the corresponding monomials incommensurable. We thus obtain the 0-th barcode as an ensemble of  $d + 1$  collections of bars, one for each era; see the middle panels in Figure 7. The alternative visualization as an ensemble of  $d + 1$  points is displayed in the right panels of the same figure. Within each era we, drop the factor  $\nu_{d-p}R^{d-p}$  of the decoration that all bars share.



■ Figure 7: *Left*: the periodic merge tree of the left panel in Figure 4 and below its decomposition into labeled bars. *Middle*: The periodic barcode split into groups of bars per era. *Right*: the corresponding ensemble of periodic persistence diagrams.

With this in mind, we compute the periodic barcode from the periodic merge tree one epoch at a time. In particular, consider a beam with left endpoint  $\Gamma \in \mathbb{M}$ , and an epoch on this beam that starts at  $A \in \mathbb{M}$  and ends entering  $B \in \mathbb{M}$ . Letting  $f(A) = s\nu_{d-p}R^{d-p}$  be the monomial decorating this epoch, we introduce two bars into the  $(d - p)$ -ary 0-th barcode:  $[\mathfrak{h}(\Gamma), \mathfrak{h}(B))$  with *multiplicity*  $s$  and  $[\mathfrak{h}(\Gamma), \mathfrak{h}(A))$  with *multiplicity*  $-s$ , but note that the second bar might be empty, in which case we skip it. The first bar tells us that  $s\nu_{d-p}R^{d-p}$  (minus the monomial of the next epoch) of the shadows born at  $\mathfrak{h}(\Gamma)$  die at  $\mathfrak{h}(B)$ , and the second bar tells us that the monomial of the previous epoch minus  $s\nu_{d-p}R^{d-p}$  of the shadows born at  $\mathfrak{h}(\Gamma)$  die at  $\mathfrak{h}(A)$ . This straightforward conversion takes time proportional to the number of epochs and gives a collection of bars with possibly negative multiplicities. We refer to the collection of the  $d + 1$  different  $(d - p)$ -ary 0-th barcodes as the *periodic 0-th barcode*, or the *periodic 0-th persistence diagram*, if we choose to visualize the information with points rather than bars. Similar to the notation for the periodic merge tree, we write  $\mathbb{B}_0(F, \Lambda)$  for the periodic 0-th barcode of  $F$  and  $\Lambda$ .

Setting  $\bar{\mathbb{R}} = (-\infty, \infty]$ , we note that for each era, the corresponding bars with multiplicities can be viewed as a function  $\mathbb{R} \times \bar{\mathbb{R}} \rightarrow \mathbb{R}$ , which maps a point  $(b, d)$  to the (possibly negative) sum of the multiplicities of all bars  $[b, d)$ . This sum is 0 if there are no such bars. While some vectorization methods for barcodes require the multiplicities to be non-negative integers, some other standard vectorization methods, such as persistence images [1], do not need such requirements and can thus be used for periodic barcodes.

## 5.2 Alternating Wasserstein Distance

While the interleaving distance between periodic merge trees is difficult to find, it is straightforward to compute a meaningful distance between periodic 0-th barcodes, as we will shortly see. We begin by formulating the Wasserstein distance between two periodic barcodes as a solution to an optimal transport problem. The bars in classic barcodes come with positive integer multiplicities, for which the Wasserstein distance is a solution to an optimal assignment problem; see [11, Section VIII.4]

## XX:20 Merge Trees of Periodic Filtrations

for an algorithm. Contrast this to the periodic setting, in which we have real and possibly negative multiplicities, which requires a solution to the more general optimal transport problem; see [25] for comprehensive background. We will work with the linear version, commonly referred to as the 1-Wasserstein distance, which is a weighted sum of distances between points. The main reason is that this version satisfies  $W_1(\xi, \eta) = W_1(\xi + \zeta, \eta + \zeta)$ , and a similar relation does not hold if we work with powers  $q > 1$  of the distances. This property allows for the removal of negative multiplicities by adding them on both sides, which is also the intuition behind the definition of the alternating 1-Wasserstein distance in (14). Recall that the *support* of a function  $f: \mathbb{R} \times \bar{\mathbb{R}} \rightarrow \mathbb{R}$ , denoted  $\text{supp}(f)$ , are the points in the domain for which  $f$  is non-zero.

► **Definition 5.1** (Alternating Wasserstein Distance). *Let  $\xi, \eta: \mathbb{R} \times \bar{\mathbb{R}} \rightarrow \mathbb{R}$  be two non-negative multiplicity functions, each with finite support. Then the 1-Wasserstein distance between  $\xi$  and  $\eta$  is*

$$W_1(\xi, \eta) = \inf_{T, X, Y} \left[ \sum_{x, y} T(x, y) \|x - y\|_1 + \sum_x X(x) \delta(x) + \sum_y Y(y) \delta(y) \right], \quad (12)$$

in which  $\delta(x) = |x_2 - x_1|$  is the vertical distance of  $x = (x_1, x_2)$  to the diagonal, and the infimum is taken over all functions  $T: \text{supp}(\xi) \times \text{supp}(\eta) \rightarrow [0, \infty)$ ,  $X: \text{supp}(\xi) \rightarrow [0, \infty)$ , and  $Y: \text{supp}(\eta) \rightarrow [0, \infty)$  that satisfy

$$X(x) + \sum_{y \in \text{supp}(\eta)} T(x, y) = \xi(x) \quad \text{and} \quad Y(y) + \sum_{x \in \text{supp}(\xi)} T(x, y) = \eta(y), \quad (13)$$

for all  $x \in \text{supp}(\xi)$  and  $y \in \text{supp}(\eta)$ . Without assuming the non-negativity of the multiplicities, we write  $\xi = \xi^+ - \xi^-$  and  $\eta = \eta^+ - \eta^-$ , in which  $\xi^+, \xi^-, \eta^+, \eta^-$  are non-negative. The more general alternating 1-Wasserstein distance between  $\xi$  and  $\eta$  is

$$W_1^\pm(\xi, \eta) = W_1(\xi^+ + \eta^-, \xi^- + \eta^+). \quad (14)$$

Note that the relation  $W_1(\xi, \eta) = W_1(\xi + \zeta, \eta + \zeta)$  implies that  $W_1^\pm(\xi, \eta)$  does not depend on the choice of  $\xi^+, \xi^-, \eta^+, \eta^-$ . We may therefore assume that  $\xi^+$  and  $\xi^-$  have disjoint supports, and similar for  $\eta^+$  and  $\eta^-$ . We will use this property in the proof of the triangle inequality for the alternating 1-Wasserstein distance.

Write  $\Delta$  for the diagonal in  $\mathbb{R}^2$ ; that is: the points  $x = (x_1, x_2)$  with  $x_1 = x_2$ . For non-negative multiplicity functions whose support avoids  $\Delta$ , the 1-Wasserstein distance is an extended metric, meaning it can also take  $\infty$  as a value, namely between persistence pairs with finite and infinite deaths. We use this property to prove that the alternating 1-Wasserstein distance is also an extended metric.

► **Lemma 5.2** ( $W_1^\pm$  is Extended Metric). *Let  $\xi, \eta, \zeta: \mathbb{R} \times \bar{\mathbb{R}} \rightarrow \mathbb{R}$  be multiplicity functions, each with finite support avoiding  $\Delta$ . Then*

- $W_1^\pm(\xi, \eta) \geq 0$  and  $W_1^\pm(\xi, \eta) > 0$  iff  $\xi \neq \eta$  (positivity);
- $W_1^\pm(\xi, \eta) = W_1^\pm(\eta, \xi)$  (symmetry);
- $W_1^\pm(\xi, \eta) + W_1^\pm(\eta, \zeta) \geq W_1^\pm(\xi, \zeta)$  (triangle inequality).

**Proof.** Write  $\xi = \xi^+ - \xi^-$ ,  $\eta = \eta^+ - \eta^-$ ,  $\zeta = \zeta^+ - \zeta^-$ , in which  $\xi^+, \xi^-$  are non-negative with disjoint supports, and similar for  $\eta^+, \eta^-$  and  $\zeta^+, \zeta^-$ . To show positivity, we note that the right-hand side of (14) can only vanish when  $\xi^+ + \eta^- = \xi^- + \eta^+$ . Since  $\text{supp}(\xi^+) \cap \text{supp}(\xi^-) = \text{supp}(\eta^+) \cap \text{supp}(\eta^-) = \emptyset$ , this implies  $\xi^+ = \eta^+$  and  $\xi^- = \eta^-$  and therefore also  $\xi = \eta$ . Symmetry is implied by the symmetry of Definition 5.1. Finally, we use the triangle inequality for positive

multiplicities to prove the same for possibly negative multiplicities:

$$W_1^\pm(\xi, \eta) + W_1^\pm(\eta, \zeta) = W_1(\xi^+ + \eta^-, \xi^- + \eta^+) + W_1(\eta^+ + \zeta^-, \eta^- + \zeta^+) \quad (15)$$

$$= W_1(\xi^+ + \eta^- + \zeta^-, \xi^- + \eta^+ + \zeta^-) \quad (16)$$

$$+ W_1(\xi^- + \eta^+ + \zeta^-, \xi^- + \eta^- + \zeta^+) \quad (17)$$

$$\geq W_1(\xi^+ + \eta^- + \zeta^-, \xi^- + \eta^- + \zeta^+) \quad (18)$$

$$= W_1(\xi^+ + \zeta^-, \xi^- + \zeta^+), \quad (19)$$

in which the last term is equal to  $W_1^\pm(\xi, \zeta)$ . In summary, the alternating Wasserstein distance satisfies positivity, symmetry, and the triangle inequality and is therefore an extended metric. ◀

We define the alternating Wasserstein distance of two periodic 0-th barcodes as the sum over the alternating Wasserstein distances of the  $(d - p)$ -ary barcodes.

### 5.3 Invariance of Periodic Barcodes

In this subsection, we prove that periodic barcodes do not depend on the lattice used in their construction. This is in contrast to periodic merge trees, for which the splintering relation is needed to get invariant equivalence classes of such trees. We begin by showing that splintering does not affect the periodic barcode derived from a periodic merge tree.

► **Lemma 5.3** (Splintering Preserves Periodic Barcodes). *Let  $\mathbb{B}$  and  $\mathbb{B}'$  be the periodic 0-th barcodes derived from the periodic merge trees  $\mathbb{M}$  and  $\mathbb{M}'$ . If  $\mathbb{M}'$  splinters  $\mathbb{M}$ , then  $\mathbb{B}' = \mathbb{B}$ .*

**Proof.** To get  $\mathbb{B}$ , we collect two bars from each epoch of  $\mathbb{M}$ , one with positive and the other with negative multiplicity, but note that the second bar may be empty. Let  $A, B \in \mathbb{M}$  delimit an epoch decorated with  $s\nu_{d-p}R^{d-p}$ , and let  $\Gamma \in \mathbb{M}$  be the leftmost point on the same beam. By the elder rule applied in the construction of the merge tree,  $h(\Gamma)$  is the minimum height of any point in the subtree rooted at  $A$ . The two bars are  $[h(\Gamma), h(B))$  with multiplicity  $s$  and  $[h(\Gamma), h(A))$  with multiplicity  $-s$ . If we subdivide the epoch into intervals we call *ages* and construct the bars from each age, we get the same result because all bars ending at points strictly between  $A$  and  $B$  cancel in pairs. This motivates us to refine the epochs of  $\mathbb{M}$  and  $\mathbb{M}'$  into ages such that

- (i) none of the ages contains the value of critical event in its interior;
- (ii) the preimages of an age in  $\mathbb{M}$  are ages in  $\mathbb{M}'$ .

Consider an age with shadow monomial  $s\nu_{d-p}R^{d-p}$  in  $\mathbb{M}$ , let  $N$  be the number of its preimages in  $\mathbb{M}'$ , and observe that each of these preimages is an age with shadow monomial  $\frac{s}{N}\nu_{d-p}R^{d-p}$ . The bars generated by the age in  $\mathbb{M}$  and its preimages in  $\mathbb{M}'$  are the same, except that the former have multiplicities  $\pm s$ , while the latter have multiplicities  $\pm \frac{s}{N}$ . Adding the contributions of the  $N$  preimages amounts to the same two bars with the same multiplicities. Since this is true for all ages, we conclude that  $\mathbb{B} = \mathbb{B}'$ . ◀

The invariance of periodic barcodes is an immediate consequence of Lemma 5.3 and Lemma 4.5. Indeed, if  $\Lambda', \Lambda'' \subseteq \Lambda$  are two  $d$ -dimensional sublattices, and  $\mathbb{M}(F, \Lambda')$ ,  $\mathbb{M}(F, \Lambda'')$  are the corresponding periodic merge trees, then they both splinter  $\mathbb{M}(F, \Lambda)$  by Lemma 4.5. By Lemma 5.3, the periodic barcodes derived from these periodic merge trees satisfy  $\mathbb{B}_0(F, \Lambda') = \mathbb{B}_0(F, \Lambda) = \mathbb{B}_0(F, \Lambda'')$ . We formulate for later reference:

► **Corollary 5.4** (Invariance of Periodic Barcodes). *Let  $\Lambda \subseteq \mathbb{R}^d$  be a  $d$ -dimensional lattice,  $F: K \rightarrow \mathbb{R}$  a  $\Lambda$ -periodic filter, and  $\Lambda', \Lambda'' \subseteq \Lambda$  two  $d$ -dimensional sublattices. Then  $\mathbb{B}_0(F, \Lambda') = \mathbb{B}_0(F, \Lambda'')$ .*

## 5.4 Size of Multiplicities

To prepare the proof of stability for periodic barcodes, this section studies how large the multiplicities can get. We thus consider the maximum multiplicity of a bar in the 0-barcode of  $F$  and  $\Lambda$ ,

$$\mu(F, \Lambda) = \max \left\{ \frac{\text{vol}_p \Lambda_\Gamma}{\text{vol}_d \Lambda} \mid \Gamma \in \mathbb{M}(F, \Lambda) \right\}, \quad (20)$$

and prove an upper bound that depends on the dimension,  $d$ , the number of edges in the quotient complex,  $m$ , the magnitude of the shift vectors,  $D$ , but also on the basis of  $\Lambda$ . As before, we write  $U$  for the matrix whose columns are the vectors in the basis of  $\Lambda$ . Recall that  $U^{-1}$  maps  $\Lambda$  to  $\mathbb{Z}^d$ , and that the operator norm,  $\|U^{-1}\|_{\text{op}} = \max_{x \in \mathbb{R}^d} \|U^{-1}(x)\|_2 / \|x\|_2$ , is the largest singular value of  $U^{-1}$  and quantifies the maximum stretching experienced by any vector  $x \in \mathbb{R}^d$ .

► **Lemma 5.5 (Maximum Multiplicity).** *Let  $\Lambda \subseteq \mathbb{R}^d$  be a  $d$ -dimensional lattice,  $U$  the matrix of basis vectors,  $F: K \rightarrow \mathbb{R}$  a  $\Lambda$ -periodic filter on a  $\Lambda$ -periodic complex,  $m$  the number of edges, and  $D$  the maximum magnitude of their shift vectors in  $K/\Lambda$ . Then  $\mu(F, \Lambda) \leq (d^{2.5} Dm \|U^{-1}\|_{\text{op}})^d$ .*

**Proof.** Write  $\mathbb{M} = \mathbb{M}(F, \Lambda)$ , let  $\Gamma \in \mathbb{M}$  be arbitrary, set  $p = \dim \Lambda_\Gamma$ , and let  $w_1, w_2, \dots, w_p$  be a basis of  $\Lambda_\Gamma$ . Write  $v_i = U^{-1}(w_i)$ . There are different ways to choose the basis. For example, the columns of the reduced matrix in Hermite normal form would be a valid choice. However, to get a better bound, we choose  $v_i$  as the output of a different reduction algorithm, described in [19], which we apply to the matrix of size  $d \times m$  whose columns are the drift vectors of the simple cycles that correspond to the fewer than  $m$  catenation events in the subtree rooted at  $\Gamma$ . As argued in the proof of Lemma 3.3, the magnitude of this matrix is at most  $Dm$ , and according to [19], the magnitude of the reduced matrix—which is not necessarily in Hermite normal form—is at most  $d^2 Dm$ . It follows that the Euclidean norm of each column vector satisfies  $\|v_i\|_2 \leq \sqrt{d}(d^2 Dm) = d^{2.5} Dm$ . In order to write  $\text{vol}_p \Lambda_\Gamma$  as a  $d$ -dimensional volume, let  $w_{p+1}, w_{p+2}, \dots, w_d$  be unit vectors orthogonal to each other and to  $\Lambda_\Gamma$ , so that  $\text{vol}_p \Lambda_\Gamma = \text{vol}_d \Lambda'_\Gamma$ , with  $\Lambda'_\Gamma$  spanned by  $w_1, w_2, \dots, w_d$ . Since a linear transformation changes the  $d$ -dimensional volume of any subset of  $\mathbb{R}^d$  in the same way, we get

$$\frac{\text{vol}_d \Lambda'_\Gamma}{\text{vol}_d \Lambda} = \frac{\text{vol}_d U^{-1}(\Lambda'_\Gamma)}{\text{vol}_d U^{-1}(\Lambda)} = \text{vol}_d U^{-1}(\Lambda'_\Gamma) \leq \prod_{i=1}^d \|v_i\|_2 \quad (21)$$

$$\leq (d^{2.5} Dm)^p \cdot \|U^{-1}\|_{\text{op}}^{d-p} \leq (d^{2.5} Dm \|U^{-1}\|_{\text{op}})^d, \quad (22)$$

in which we get the third term in (21) by recalling that  $U^{-1}(\Lambda) = \mathbb{Z}^d$ , and we get the first term in (22) by using Lemma 3.3 for the first  $p$  and the operator norm for the last  $d - p$  basis vectors. ◀

*Remark about dependence on basis.* Both  $D$  and  $\|U^{-1}\|_{\text{op}}$  depend on the basis we choose for  $\Lambda$ , but by Corollary 5.4, the barcode and therefore the multiplicities of its bars do not. To gain a better understanding of the size of the multiplicities, it would therefore be useful to know how small  $D$  and  $\|U^{-1}\|_{\text{op}}$  can be made by careful choice of a basis. Indeed, the bound in Lemma 5.5 seems excessively pessimistic. It would be interesting to find out what the maximum multiplicity is in practice, e.g. on average for the periodic structures collected in popular crystallographic databases.

## 5.5 Stability of Periodic Barcodes

The cellular  $\ell_1$ -distance between filters  $F/\Lambda, G/\Lambda: K/\Lambda \rightarrow \mathbb{R}$  is the sum of absolute differences over all simplices,  $\|F/\Lambda - G/\Lambda\|_1 = \sum_{\sigma \in K/\Lambda} |F/\Lambda(\sigma) - G/\Lambda(\sigma)|$ , but note that this is different from the  $L_1$ -distance between the two functions, which is the integral of their point-wise differences. By comparing the 1-Wasserstein distance of two non-periodic barcodes with the cellular  $\ell_1$ -distance

between the corresponding filters, Skraba and Turner [27] prove stability with Lipschitz constant 1. Following this approach, we prove stability with respect to the cellular  $\ell_1$ -distance in the periodic setting, but with worse Lipschitz constant. Observe first that the periodic 0-th barcode is invariant under changing how ties in the ordering of the simplices are broken. Indeed, the periodic merge tree is invariant, and the asymmetry introduced by the elder rule has no effect on the periodic barcode computed from the periodic merge tree. This invariance is of course necessary for the stability of periodic barcodes, which we prove next.

► **Theorem 5.6** (Stability of Periodic Barcodes). *Let  $\Lambda \subseteq \mathbb{R}^d$  be a  $d$ -dimensional lattice,  $U$  the matrix of a basis of  $\Lambda$ , and  $F, G: K \rightarrow \mathbb{R}$  two  $\Lambda$ -periodic filters. Then*

$$W_1^\pm(\mathbb{B}_0(F, \Lambda), \mathbb{B}_0(G, \Lambda)) \leq 2(d+1)\mu_0 \cdot \|F/\Lambda - G/\Lambda\|_1, \quad (23)$$

in which  $\mu_0 = (d^{2.5} Dm \|U^{-1}\|_{\text{op}})^d$  is the upper bound on the maximum multiplicity from Lemma 5.5, with  $m$  the number of edges of  $K/\Lambda$ , and  $D$  the maximum magnitude of any of their shift vectors.

**Proof.** The proof is analogous to that of [27, Lemma 4.7 and Theorem 4.8], which itself follows the strategy of the proof of [5, Combinatorial Stability Theorem]. It consists of two steps: the first assumes that  $F$  and  $G$  are compatible to a common ordering of the simplices, and the second analyzes the transpositions necessary to make them compatible to such an ordering, if they are not.

*Step 1.* Assume the filters  $F$  and  $G$  allow for a total order of the simplices that is compatible to both; that is:  $\sigma_i \subseteq \sigma_j$ ,  $F(\sigma_i) < F(\sigma_j)$ , and  $G(\sigma_i) < G(\sigma_j)$  all imply  $i < j$ . Let  $\text{index}: K/\Lambda \rightarrow \mathbb{R}$  be the filter that assigns to each simplex its index in this order, and construct  $\mathbb{M}_{\text{index}} = \mathbb{M}(\text{index}, \Lambda)$ . Replacing the height  $h(\Gamma) = i$  of a critical event by  $h(\Gamma) = F(\sigma_i)$  and  $h(\Gamma) = G(\sigma_i)$ , respectively, we obtain two periodic merge trees,  $\mathbb{M}_F$  and  $\mathbb{M}_G$ , whose only difference is the height function. However, since  $F(\sigma_i) = F(\sigma_j)$  is possible even if  $i \neq j$ , some epochs in  $\mathbb{M}_{\text{index}}$  may correspond to empty epochs in  $\mathbb{M}_F$ , and similarly for  $\mathbb{M}_G$ . When we construct the periodic barcodes,  $\mathbb{B}_F = \mathbb{B}_0(F, \Lambda)$  and  $\mathbb{B}_G = \mathbb{B}_0(G, \Lambda)$ , we use the empty epochs as well, while noting that each contributes two bars that cancel each other. The bijection between the epochs of  $\mathbb{M}_F$  and  $\mathbb{M}_G$  (both empty and non-empty) thus gives a bijection between  $\mathbb{B}_F$  and  $\mathbb{B}_G$ , which is a transportation plan,  $T: \text{supp}(\xi) \times \text{supp}(\eta) \rightarrow \mathbb{R}$ , as in Definition 5.1.<sup>2 3</sup> We prove that the cost of this particular transportation plan is bounded from above by  $2(d+1)\mu_0 \cdot \|F/\Lambda - G/\Lambda\|_1$ , which will imply

$$W_1^\pm(\mathbb{B}_F, \mathbb{B}_G) \leq 2(d+1)\mu_0 \cdot \|F/\Lambda - G/\Lambda\|_1. \quad (24)$$

By construction, we compare two bars if they are born and die at the hands of the same two simplices. Their contribution to  $T$  is therefore the difference in birth values plus the difference in death values, times the absolute multiplicity. Depending on the type of critical event a simplex causes, its value may mark more or fewer births or deaths. If  $\sigma$  causes a catenation or merge event, its value marks the death of at most three bars, with the absolute multiplicity of each bounded by  $\mu_0$ . So the total cost for transporting the corresponding death values from  $F$  to  $G$  is bounded by  $3\mu_0|F(\sigma) - G(\sigma)|$ . If  $\sigma$  causes an appearance, its value may mark the births of any number of bars. However, within each era, absolute multiplicities form a telescoping series whose sum is bounded by  $2\mu_0$ . Since there are  $d+1$  eras, the cost for transporting the corresponding birth values from  $F$  to  $G$  is bounded by  $2(d+1)\mu_0 \cdot |F(\sigma) - G(\sigma)|$ . Adding the cost for the death and the birth values implies (24).

<sup>2</sup> Whenever the transportation plan maps to or from a specific point on the diagonal, we could instead use  $X$  or  $Y$  to or from the closest point on the diagonal. For the sake of a simpler argument, we refrain from using  $X$  and  $Y$  and rely solely on  $T$ , which leads to a possible over-estimation of the 1-Wasserstein distance.

<sup>3</sup> Note also that instead of canceling the two bars contributed by an empty epoch, we keep both. This corresponds to looking at  $W_1(\xi + \zeta, \eta + \zeta)$  instead of  $W_1(\xi, \eta)$ , but recall that they agree.

## XX:24 Merge Trees of Periodic Filtrations

*Step 2.* If  $F$  and  $G$  are not necessarily compatible with the same total order, we look at the straight-line homotopy between the two filters:  $F_t = (1 - t)F + tG$  for  $t \in [0, 1]$  and observe that

- $F_0 = F$  and  $F_1 = G$ ;
- each  $F_t$  is a periodic filter with respect to  $\Lambda$ ;
- for each  $0 \leq s \leq t \leq 1$ ,  $\|F_s/\Lambda - F_t/\Lambda\|_1 = (t - s)\|F/\Lambda - G/\Lambda\|_1$ ;
- there are only finitely many values  $0 = t_0 < t_1 < \dots < t_k = 1$  at which the total order compatible with  $F_t$  changes.

With this, we finish the proof by taking the sum of the bounds in (24) for each interval between two such values:

$$W_1^\pm(\mathbb{B}_F, \mathbb{B}_G) \leq \sum_1^k W_1^\pm(\mathbb{B}_0(F_{t_{i-1}}, \Lambda), \mathbb{B}_0(F_{t_i}, \Lambda)) \quad (25)$$

$$\leq 2(d + 1)\mu_0 \sum_1^k \|F_{t_{i-1}}/\Lambda - F_{t_i}/\Lambda\|_1 \quad (26)$$

$$= 2(d + 1)\mu_0 \cdot \|F/\Lambda - G/\Lambda\|_1, \quad (27)$$

which is the claimed inequality. ◀

*Remark about invariance.* When a perturbation breaks a symmetry and thus forces a coarser lattice, like in Figure 6, Theorem 5.6 can be applied to the filter before and after the perturbation, both considered under the coarse lattice. By Corollary 5.4, passing to an unnecessarily coarse lattice does not change the periodic barcode.

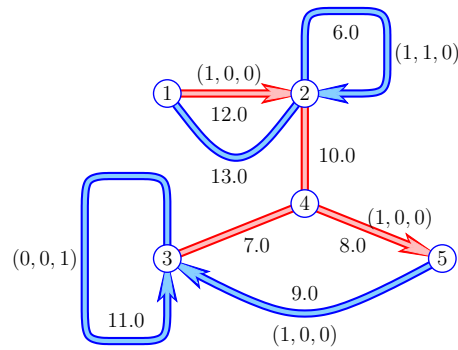
*Remark about distortion.* For a  $\Lambda$ -periodic  $F: K \rightarrow \mathbb{R}$ , and an arbitrarily small  $\varepsilon > 0$ , a uniform scaling of  $K$  and  $\Lambda$  by  $1 + \varepsilon$  that preserves the filter values yields a periodic merge tree with slightly smaller shadow monomials. Hence, there is no arbitrarily small interleaving with the original periodic merge tree. However, the 1-Wasserstein distance between the corresponding periodic barcodes is less sensitive to small differences in the multiplicities and therefore the better choice in coping with the effect of slight scalings, or more general affine linear transformations with singular values close to 1.

*Remark about Delaunay triangulations.* Note that the perturbation of the vertices of a 2-dimensional Delaunay triangulation may change the filter values but possibly also the underlying graph, which happens only when four points lie on a common circle so one diagonal of a convex quadrangle *flips* to the other diagonal. In this case, neither diagonal contributes to the merge trees, because their endpoints are connected via shorter edges of the quadrangle. Hence, a diagonal flip does not affect the periodic merge tree. Since Theorems 4.7 and 5.6 apply to each part of the straight line homotopy between Delaunay triangulations during which the graph structure does not change—and there is no contribution due to the diagonal flips—the two theorems can be used to bound the distance between our descriptors of two Delaunay triangulations with possibly non-isomorphic edge-skeletons.

### 6 A Working Example

This section illustrates the concepts introduced in Sections 3 to 5 using the periodic graph in Figure 9 as a running example. Its quotient in the 3-dimensional torus has five vertices, labeled from 1 to 5, and eight edges, each labeled with its filter value; see Figure 8. If an edge has shadows whose endpoints lie in different copies of the unit cell, we choose an arbitrary direction and label the directed edge with the shift vector that locates the unit cell of the target vertex relative to that of the source vertex. In Figure 8, two of the blue edges connect to the same vertex twice, and they correspond to two families of parallel lines in  $\mathbb{R}^3$ : with diagonal direction defined by the drift vector  $(1, 1, 0)$ , and with vertical direction defined by the drift vector  $(0, 0, 1)$ . The loop formed by the two edges connecting vertices 1 and 2 corresponds to another family of lines with direction defined by  $(1, 0, 0)$ .





■ Figure 8: The quotient of the graph in Figure 9. Each edge is labeled by its value in the filter, and some also by their shift vectors. The spanning tree edges, which merge components, are drawn straight and *red*, and the others are drawn curved and *blue*. Each edge whose shadow has endpoints in different copies of the unit cell has a non-zero shift vector associated to a directed version of the edge.

More interesting is the loop formed by the three edges connecting vertices 3, 4, and 5. The drift vector is  $(2, 0, 0)$ , which implies that two copies of the strand (drawn curved in Figure 9 for better visibility) form a double helix that passes through a linear sequence of unit cells.

Next we go through the motion of constructing the periodic merge tree incrementally, by adding one vertex or edge at a time in the order of increasing filter value. The top panel of Figure 10 shows the final tree. We pay special attention to the periodicity lattices, which are maintained incrementally and used to compute the shadow monomials decorating the beams of the tree. For simplicity, we assume that the underlying lattice is  $\Lambda = \mathbb{Z}^3$ , whose unit cell is the unit cube with unit volume.

STEPS 1 TO 5: adding the five vertices, we get five connected components, each with trivial periodicity lattice,  $\Lambda_1 = \Lambda_2 = \Lambda_3 = \Lambda_4 = \Lambda_5 = \{0\}$ , in which we use the labels of the vertices as indices. Each of these lattices has 0-dimensional volume 1, which implies that the shadow monomial is  $\frac{4\pi}{3} R^3$ ; see the left ends of the beams in the top panel of Figure 10.

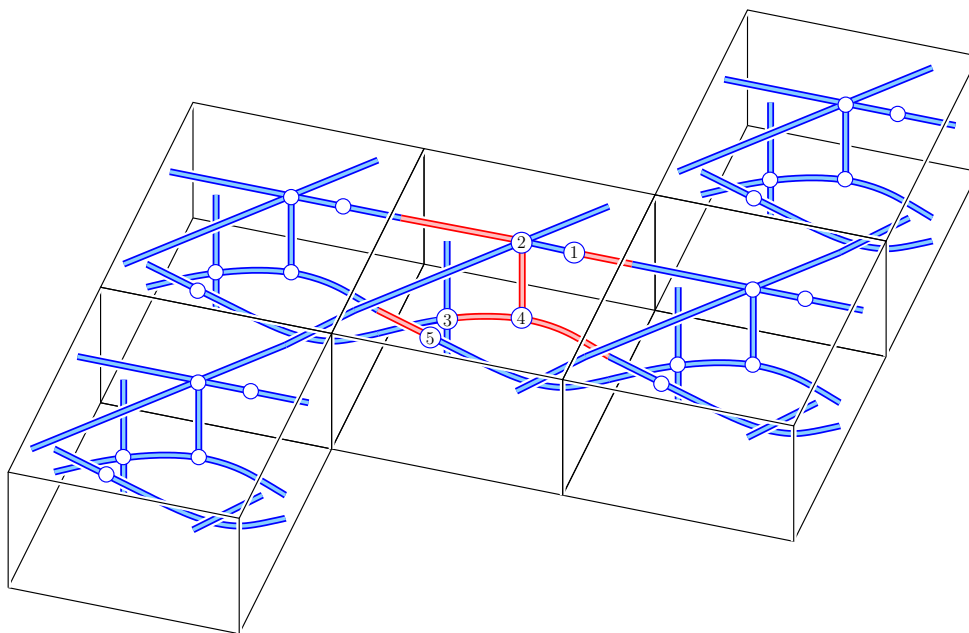
STEP 6: the edge with filter value 6.0 connects vertex 2 to itself, and its addition changes the periodicity lattice of the corresponding component to  $\Lambda_2 = \Lambda((1, 1, 0))$ . Its unit cell has 1-dimensional volume  $\sqrt{2}$ . Note that the dimension of the periodicity lattice increases from 0 to 1, and the shadow monomial decreases from cubic to quadratic. The drop of the exponent means the component leaves the cubic and enters the quadratic era.

STEPS 7 TO 9: the edges with filter values 7.0, 8.0, 9.0 form the helix in the periodic graph. The addition of the first edge merges the components of vertices 3 and 4, and the addition of the second edge merges this component with that of vertex 5. These mergers do not affect the periodicity lattices. However, the third edge completes the helix with periodicity lattice  $\Lambda_3 = \Lambda((2, 0, 0))$  and shadow monomial  $2\pi R^2$ . The factor 2 indicates that shifting a strand by  $(1, 0, 0)$  produces a second strand, so we are dealing with a double and not a single helix.

STEP 10: the edge with filter value 10.0 connects the line from Step 6 with the helix from Step 9. The corresponding periodicity lattice is spanned by the vectors  $(1, 1, 0)$  and  $(2, 0, 0)$ , which after reduction becomes  $\Lambda_2 = \Lambda((1, 1, 0), (0, 2, 0))$ . The 2-dimensional volume of its unit cell is 2, and so is the 1-dimensional volume of the unit ball in  $\mathbb{R}^1$ , which is why the shadow monomial is  $4R$ ; see Figure 10. Note that the component leaves the quadratic and enters the linear era.

STEP 11: the edge with filter value 11.0 connects vertex 3 to itself. Its addition combines the 2-dimensional structures from Step 10 to form a bigger, 3-dimensional structure. Indeed, the periodicity lattice is spanned by vectors  $(1, 1, 0)$ ,  $(0, 2, 0)$ , and  $(0, 0, 1)$ , whose unit cell has 3-dimensional volume 2. The fact that the volume is 2 and not 1 suggests that the two strands of

## XX:26 Merge Trees of Periodic Filtrations

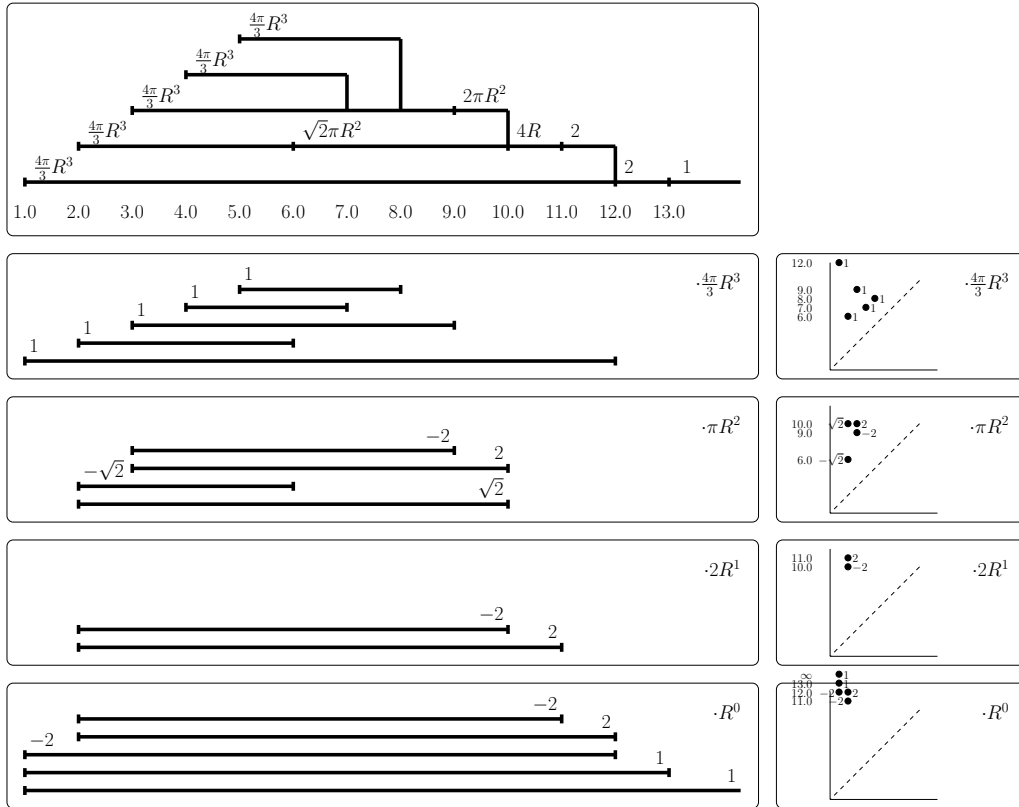


■ Figure 9: A periodic graph in  $\mathbb{R}^3$  serving as a running example to illustrate the periodic merge tree and its construction. The portion of the graph inside five copies of the unit cell is shown. Inside the middle unit cell, the vertices are shown with labels and the edges in the spanning tree are drawn *red*. The motif within this cell consists of a cross above a double helix, with a single edge connecting the two, and, in addition, a vertical line sharing a point with the helix. This motif repeats periodically. The two lines of the cross meet at vertex 2, and together with their periodic translates form a series of 2-dimensional grids. The two strands of the double helix are periodic translates of each other and pass from left to right through copies of vertices 3, 4, 5, in this cyclic sequence. For convenience, the vertex labels are also used as filter values, and the values of the edges are shown in Figure 8, which displays the corresponding quotient, i.e. the graph in the 3-dimensional torus.

the double helix are still in separate components. The corresponding shadow monomial is 2, so the component leaves the linear and enters the constant era.

STEPS 12 AND 13: until now, vertex 1 was isolated, with its own beam in the periodic merge tree. The first edge with endpoints 1 and 2 merges the two components, keeping the larger of the two periodicity lattices unchanged, but assigning it now to beam 1, by the elder rule. The second such edge forms a loop with drift vector  $(1, 0, 0)$ . We use the reduction algorithm to select a basis from the three vectors spanning the lattice constructed in Step 11 and the drift vector of the new loop. The result is  $\Lambda_1 = \Lambda((1, 0, 0), (0, 1, 0), (0, 0, 1))$ , and the shadow monomial is 1, so this step moves the one remaining component from a first to the second and final epoch in the constant era; see the right end of the tree in the top panel of Figure 10.

*Remark about stability.* If we alter the value of the second vertex from  $2$  to  $1 + \varepsilon$ , for an arbitrarily small  $\varepsilon > 0$ , the periodic merge tree and periodic barcode do not change combinatorially, but if we alter it to  $1 - \varepsilon$ , then this vertex swaps with the current first vertex and thus changes the combinatorics of the tree and barcode. While the adjustment from  $1 + \varepsilon$  to  $1 - \varepsilon$  mostly affects birth values in the arbitrarily small interval between these value, there is one change worth noting: the two epochs with shadow monomial 2—which belong to different beams in the tree shown in Figure 10—merge into one epoch on one beam. Therefore, the points  $(2, 12)$  with multiplicity 2 and  $(1, 12)$  with multiplicity  $-2$  disappear. This illustrates the importance of negative multiplicities, which are essential for stability as they facilitate the matching of such pairs using the alternating 1-Wasserstein distance.



■ Figure 10: *Top*: the periodic merge tree of the filtered graph in Figure 9. The left endpoints of the beams correspond to vertices of the quotient graph, and the vertical segments correspond to edges in its spanning tree. The other edges form loops and thus do not affect the set of components but possibly alter the shadow monomials, which decorate the beams. *Below the top on the left*: the four collections of bars labeled with their possibly negative multiplicities. *On the right*: the corresponding four persistence diagrams with similarly labeled points. The bottommost constant diagram has a point with infinite vertical coordinate, which is drawn above its window.

## 7 Discussion

The main contribution of this paper is the extension of the persistent homology framework to the periodic setting, albeit only for connected components; that is: for homological dimension 0. The crucial ingredient in this extension is the notion of a *shadow monomial*, which quantifies the growth-rate and density of the translates of a component. The two new data structures are the *periodic merge tree* and the *periodic 0-barcode*, with the former containing strictly more information and the latter offering computational advantages and an easier connection to machine learning tools; see the text by Peyré and Cuturi [25] for optimal transport algorithms and their application to data science. Both data structures are invariant under isometries and a change of basis, which can be seen directly from the definition of the shadow monomial. While the periodic merge tree splinters when we pass to a sublattice of the original lattice, the periodic 0-th barcode is invariant under such changes, and so are the equivalence classes of periodic merge trees. Both data structures are stable under perturbations.

The reported work suggests several avenues of future study aimed at broadening and deepening the capabilities of the method and thus make it more compelling in applications to the sciences.

- For the sake of convenience, we restricted ourselves to  $d$ -periodic filtrations in  $\mathbb{R}^d$ . However, the theory extends to the setting of  $k$ -periodic filtrations in  $\mathbb{R}^d$ , by replacing all  $d$  by  $k$  in the

- definition of the shadow monomial. The non-periodic case,  $k = 0$ , then yields shadow monomials equal to 1 and thus the classical merge tree and 0-th persistent homology. To what extent can the theory, and in particular Definition 2.2 and Lemma 2.3, be further extended to groups acting on topological spaces different from  $\mathbb{R}^d$ ?
- Further extend the results of this paper to homological dimensions beyond 0. Some such capability can already be achieved by applying the periodic merge tree to the dual filtration, describing the complements of the original filtration. While this idea is not new [8], a key difference is that the infinite periodic space is not compact, and therefore Alexander duality does not hold in the usual way. As a consequence, we not only get information about  $(d - 1)$ - but also about  $(d - 2)$ -dimensional homology. In  $\mathbb{R}^3$ , 2-dimensional homology (voids) of the original filtration corresponds to the cubic eras of the dual filtration. In contrast, the quadratic eras, for example, describe infinitely long tunnels, so a particular kind of 1-dimensional homology.
  - Assuming success in extending periodic barcodes to higher homological dimension, it will be interesting to combine the periodic with the chromatic setting recently developed in [7]. Indeed, crystalline materials are periodic and typically consist of more than one type of atoms, which suggests both concepts in its analysis.

We conclude this paper with a concrete geometric question about the Delaunay triangulation of a locally finite  $\Lambda$ -periodic set in  $\mathbb{R}^d$ : does there exist a constant  $D = D(d)$  and a basis of  $\Lambda$  such that the shift vector of every edge in the Delaunay triangulation has magnitude at most  $D$ ? We recall that this constant is  $D = 1$  if  $\Lambda$  has a basis with pairwise orthogonal vectors, as proved in Section 3.5. However, the proof does not extend to the case in which the basis vectors enclose arbitrary angles. Is there a similar bound for nearly orthogonal bases constructed with the Lenstra–Lenstra–Lovász (LLL) lattice basis reduction algorithm?

## Acknowledgements

The authors thank Georg Osang and Dmitriy Morozov for their motivating words, Alexey Garber, János Pach, Morteza Saghafia, Nicolò Zava and Manuel Soriano Trigueros for technical discussions on the topic of this paper, and most of all Chiara Martyka for discussions on lattices and software implementing an earlier version of the algorithm.

---

## References

- 1 H. ADAMS ET AL. Persistence images: a stable vector representation of persistent homology. *J. Machine Learning Res.* **18** (2017), 1–35.
- 2 D. BURAGO, Y. BURAGO AND S. IVANOV. *A Course in Metric Geometry*. Amer. Math. Soc., Providence, Rhode Island, 2001.
- 3 G. CARLSSON. Topology and data. *B. Am. Math. Soc.* **46** (2009), 255–308.
- 4 J.W.S. CASSELS. *An Introduction to the Geometry of Numbers*. Classics Math., Springer, Berlin, 1997.
- 5 D. COHEN-STEINER, H. EDELSBRUNNER, AND D. MOROZOV. Vines and vineyards by updating persistence in linear time. In “Proc. 22nd Ann. Sympos. Comput. Geom., 2006”, 119–126.
- 6 E. COHEN AND N. MEGIDDO. Recognizing properties of periodic graphs. In *Applied Geometry and Discrete Mathematics*, DIMACS Ser. Discrete Math. Theoret. Comput. Sci. **4**, 135–146, 1990.
- 7 S. CULTRERA DI MONTESANO, O. DRAGANOV, H. EDELSBRUNNER, AND M. SAGHAFIAN. Chromatic alpha complexes. [arXiv:2212.03128 \[math.AT\]](https://arxiv.org/abs/2212.03128), 2024.
- 8 C.J.A. DELFINADO AND H. EDELSBRUNNER. An incremental algorithm for Betti numbers of simplicial complexes on the 3-sphere. *Comput. Aided Geom. Design* **12** (1995), 771–784.
- 9 W. DICKS AND M.J. DUNWOODY. *Groups Acting on Graphs*. Cambridge Univ. Press, Cambridge, England, 1989.
- 10 N.P. DOLBILIN AND D.H. HUSON. Periodic Delone tilings. *Period. Math. Hung.* **34** (1997), 57–64.
- 11 H. EDELSBRUNNER AND J.L. HARER. *Computational Topology. An Introduction*. American Mathematical Society, Providence, Rhode Island, 2010.
- 12 H. EDELSBRUNNER, T. HEISS, V. KURLIN, P. SMITH AND M. WINTRAECKEN. The density fingerprint of a periodic point set. In “Proc. 37th Ann. Sympos. Comput. Geom., 2021”, 32:1–32:16.
- 13 E.S. FEDOROV. Symmetry of regular systems of figures. *Zap. Mineral. Obch.* **28** (1891), 1–146.

- 14 Z. GALIL AND G. ITALIANO. Data structures and algorithms for disjoint set union problems. *ACM Computing Surveys* **23** (1991), 319–344.
- 15 G. HAVAS, B.S. MAJEWSKY AND K.R. MATHEWS. Extended gcd and Hermite normal form algorithms via lattice basis reduction. *Exp. Math.* **7** (1998), 125–136.
- 16 Y. HIRAOKA, T. NAKAMURA, A. HIRATA, E.G. ESCOLAR, K. MATSUE AND Y. NISHIURA. Hierarchical structures of amorphous solids characterized by persistent homology. *PNAS* **113** (2016), 7035–7040.
- 17 N.R. HOWES. *Modern Analysis and Topology*. Univeritext, Springer-Verlag, Berlin and New York, 1995.
- 18 R. KANNAN AND A. BACHEM. Polynomial algorithms for computing the Smith and Hermite normal forms of an integer matrix. *SIAM J. Comput.* **8** (1979), 499–507.
- 19 K.-M. KLEIN AND J. REUTER. Simple lattice basis computation—the generalization of the Euclidean algorithm. [arXiv:2311.15902 \[cs.DS\]](https://arxiv.org/abs/2311.15902), 2023.
- 20 A.S. KRISHNAPRIYAN, M. HARANCZYK AND D. MOROZOV. Topological descriptors help predict guest adsorption in nanoporous materials. *J. Phys. Chem. C* **124** (2020), 9360–9368.
- 21 J.B. KRUSKAL. On the shortest spanning subtree of a graph and the traveling salesman problem. *Proc. Amer. Math. Soc.* **7** (1956), 48–50.
- 22 A. MADER. *Almost Completely Decomposable Groups*. Gordon and Breach, Amsterdam, Netherlands, 2000.
- 23 D. MOROZOV, K. BEKETAYEV AND G. WEBER. Interleaving distance between merge trees. *Discrete Comput. Geom.* **49** (2013), 22–45.
- 24 A. ONUS AND V. ROBINS. Quantifying the homology of periodic cell complexes. [arXiv:2208.09223 \[math.AT\]](https://arxiv.org/abs/2208.09223), 2022.
- 25 G. PEYRÉ AND M. CUTURI. *Computational Optimal Transport: With Applications to Data Science*. Foundations and Trends in Machine Learning **11**, Now Publisher, Boston–Delft, 2019.
- 26 A. SCHRIJVER. *Theory of Linear and Integer Programming*. Wiley-Interscience, Chichester, England, 1998.
- 27 P. SKRABA AND K. TURNER. Wasserstein stability for persistence diagrams. [arXiv:2006.16824 \[math.AT\]](https://arxiv.org/abs/2006.16824), 2020.
- 28 D. SMIRNOV AND D. MOROZOV. Triplet merge trees. In *Topological Methods in Data Analysis and Visualization V (TopoInVis’17)*, 2020.
- 29 A. STORJOHANN. *Algorithms for Matrix Canonical Forms*. Ph.D. thesis ETH No. 13922, Swiss Federal Institute of Technology, Zurich, Switzerland, 2000.
- 30 S.S. THYGESEN, A.I. ABRIKOSOV, P. STENETEG, T. BIN MASOOD AND I. HOTZ. Level of detail visual analysis of structures in solid-state materials. In “Proc. 25th EG Conf. Visualization, 2023”, 55–59.
- 31 S. WEINBERGER. What is . . . persistent homology? *Notices Amer. Math. Soc.* **58** (2011), 36–39.
- 32 B. ZHILINSKII. *Introduction to Louis Michel’s Lattice Geometry Through Group Action*. EDP Science, CNRS edition, Paris, France, 2015.
- 33 THE CGAL PROJECT. *CGAL, Computational Geometry Algorithms Library*. <https://www.cgal.org>.

**A** Notation

$\mathbb{R}^d$	$d$ -dimensional Euclidean space
$\sigma, \tau \in K$	cells, cell complex
$K_t = F^{-1}(-\infty, t]$	sublevel set
$F: K \rightarrow \mathbb{R}$	filter
$h: \mathbb{M} \rightarrow \mathbb{R}$	height function on merge tree
$f: \mathbb{M} \rightarrow \mathcal{M}[\mathbb{R}]$	frequency function to the set of monomials
$\Lambda = \Lambda(u_1, \dots, u_d); U$	lattice spanned by vectors; basis matrix
$u, w \in \Lambda, \lambda_i \in \mathbb{Z}$ ,	lattice points; integer coefficients
$\text{span}(\Lambda); \text{vol}_p \Lambda, \nu_p$	linear space; volume of unit cell and unit ball
$F/\Lambda: K/\Lambda \rightarrow \mathbb{R}$	quotient filter on quotient complex
$A, B, \Gamma$	connected components
$\alpha, \beta, \gamma$	corresponding shadows
$\Lambda_\Gamma, \mathfrak{m}_\Gamma(R)$	periodicity lattice, shadow monomial
$\Lambda_\Gamma = \Lambda_A + \Lambda_B$	sum, smallest common superlattice
$r, s; x, y, z$	roots; points, vertices, or nodes
$\text{Root}(x), \text{Size}(r)$	root of component, #vertices in component
$\text{Next}(x)$	successor in linear list
$a, \text{Shift}(a), \text{Drift}(x)$	arc, shift vector, drift vector of path from root
$n, m$	number of vertices, edges in quotient complex
$V; v_1, \dots, v_p$	integer basis matrix; integer vectors
$M[i, j]; \ M\ _\infty; \mu$	integer basis matrix; max magnitude; multiplicity
$\mathbb{M} = \mathbb{M}(F, \Lambda)$	periodic merge tree
$\mathbb{M}, \mathbb{M}', \mathbb{M}''$	splintering periodic merge trees
$\mathbb{N}, \mathbb{N}', \mathbb{O}$	extra periodic merge trees
$\varphi, \psi; \omega$	maps between periodic merge trees
$\mathbb{M} \simeq \mathbb{M}', [\mathbb{M}]$	equivalence between, class of periodic merge trees
$I(\mathbb{M}, \mathbb{M}')$	interleaving distance
$J([\mathbb{M}], [\mathbb{M}'])$	interleaving pseudo-distance
$\xi, \eta, \zeta: \mathbb{R} \times \overline{\mathbb{R}} \rightarrow \mathbb{R}$	multiplicity functions
$W_1(\xi, \eta), W_1^\pm(\xi, \eta)$	(alternating) 1-Wasserstein distance
$T, X, Y$	transportation plans used in the optimization
$\xi = \xi^+ - \xi^-$	partition into positive, negative multiplicities
$\mathbb{B} = \mathbb{B}_0(F, \Lambda)$	periodic 0-th barcode
$\mu_0, \mu(F, \Lambda)$	upper bound, maximum multiplicity of bars

■ Table 1: Notation used in the paper.