Decomposing filtered chain complexes: Geometry behind barcoding algorithms

Wojciech Chachólski, Barbara Giunti *, Alvin Jin, Claudia Landi

ABSTRACT

In Topological Data Analysis, filtered chain complexes enter the persistence pipeline between the initial filtering of data and the final persistence invariants extraction. It is known that they admit a tame class of indecomposables, called interval spheres. In this paper, we provide an algorithm to decompose filtered chain complexes into such interval spheres. This algorithm provides geometric insights into various aspects of the standard persistence algorithm and two of its runtime optimizations. Moreover, since it works for any filtered chain complexes, our algorithm can be applied in more general cases. As an application, we show how to decompose filtered kernels with it.

© 2022 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

1. Introduction

Starting from the 1990s, new methods of topological shape analysis arose [15,21,22,36], and developed in what is nowadays known as *persistent homology* [8,17], with applications varying from quantum physics [35] to medicine [7], biology [39], and geology [26] (see [25] for an extended list of references to applications). The underlying idea is that homology captures shape features such as components and voids. Measuring their persistence along a filtration makes it possible to infer their significance. A popular method to apply persistence to data sampled from an underlying shape is to build its *Vietoris–Rips simplicial filtration*. This construction comes with guarantees about the actual presence of the homological features detected by persistence in the underlying shape [17,34].

Recently, the authors of [11] have proposed a method to study not only the non-contractible parts of the data but also the contractible ones by using directly *filtered chain complexes* without applying homology to them. Among the results of [11], the most relevant here is the structure theorem stating that filtered chain complexes can be written as direct sums of indecomposables called *interval spheres*. In plain words, this means that a filtered chain complex contains algebraic shadows of geometry in the form of algebraic disks and spheres glued together. It is also interesting that filtered chain complexes do not arise only as chain complexes of filtered simplicial complexes. For example, the cover construction in [11] approximates any tame parametrised chain complex by a filtered chain complex. So, filtered chain complexes provide also invariants of (forward) commutative ladders [19] and zigzag modules [9]. Thus, an algorithm decomposing filtered chain complexes can be applied in various scenarios.

The main goal of this paper is to propose an algorithm based on geometry to decompose filtered chain complexes into direct sums of interval spheres. With this goal in mind, we first identify *quasi-minimal sets of generators* as a satisfactory trade-off between the complexity of the problem of pre-computing minimal sets of generators, which is equivalent to that

---

of the decomposition into interval spheres itself, and more general sets of generators that may contain an arbitrarily large number of elements. In the case of a filtered chain complex built from a simplicial complex, simplices themselves form a quasi-minimal set of generators of the filtered chain complex. Then we show how to split interval spheres off a filtered chain complex by choosing appropriate pairs of generators and how this split reflects in the boundary matrix of the filtered chain complex. In particular, each split is performed by modifying and shrinking the original matrix, first via a row Gaussian elimination and then by deleting two rows and two columns. It is worth pointing out that our method does not rely on a definite ordering of generators and can process boundary matrices with rows and columns in any arbitrary order, thus leaving the possibility of choosing a preferred ordering for convenience.

Since our algorithm works for arbitrary filtered chain complexes, not only for those arising from filtered simplicial complexes, we can apply it in novel settings. As a demonstration of this, we introduce the notion of a *filtered kernel*. This is a filtered chain complex given by the kernel of an epimorphism between two filtered simplicial complexes. We remark that filtered kernels are in general only partially related to kernel persistence [13,33,37], but arise naturally when studying, e.g., metric contractions. In this context, we determine a quasi-minimal set of generators for a filtered kernel and its corresponding boundary matrix. In this way, simply applying our decomposition algorithm produces the decomposition of filtered kernels.

As the third contribution of this paper, we provide geometrical insights on the standard persistence algorithm [16], viewed as a decomposition of filtered chain complexes, and its clear and compress optimisations [5,12], viewed as peeling off interval spheres from a filtered chain complex. Indeed, from our decomposition of filtered chain complexes into interval spheres, we straightforwardly obtain the decomposition of the corresponding persistence module into interval modules [10,34] simply forgetting all the intervals of length zero. Vice versa, since most of the persistence algorithms work in practice at the chain complex level, it is possible to recover the decomposition of filtered chain complexes from them. Therefore, our method can be seen as a geometric interpretation of some persistence algorithms [2,4,6,17,18,29,32] and their operations on the boundary matrix. Combining this with the possibility of applying our algorithm to arbitrary filtered chain complexes, one can extend the application of persistence algorithms beyond filtered cell complexes.

Outline In Section 2, we introduce the necessary background on filtered chain complexes, and we review the relevant persistence algorithms. In Section 3, we present our decomposition algorithm for filtered chain complexes, and we show how to apply it to filtered kernels. In Section 4, we highlight the connections to persistence algorithms and elucidate the geometry behind them. Finally, in Section 5, we discuss some further questions that may be interesting to explore.

2. Preliminaries

Throughout the work, $\mathbb{F}$ denotes a finite prime field of characteristic p for a prime integer p.

2.1. *Chain complexes*. Let $\mathbf{N} = \{0, 1, \ldots\}$ be the set of natural numbers. A (non-negatively graded) chain complex of $\mathbb{F}$-vector spaces is a sequence of linear maps $C = \{\delta_{n+1} \colon C_{n+1} \to C_n\}_{n \in \mathbf{N}}$ of $\mathbb{F}$-vector spaces, called **differentials**, such that $\delta_n \delta_{n+1} = 0$ for all n in $\mathbb{N}$. A chain complex C is called **compact** if $\bigoplus_{n \in \mathbf{N}} C_n$ is finite dimensional as a vector space [1]. This happens if and only if C_n is finite dimensional for all n and C_n is trivial for all $n > N \in \mathbf{N}$. We denote by **ch** the category of compact chain complexes. Throughout this work, all considered chain complexes are compact.

We call an *n***-sphere**, or simply a **sphere**, the chain complex S^n that is zero in all degrees but in degree n, where it is isomorphic to $\mathbb{F}$. We call an *n***-disk**, or simply a **disk**, the chain complex D^n, given by

$$D_k^n = \begin{cases} \mathbb{F} & \text{if } k = n, n-1 \\ 0 & \text{otherwise,} \end{cases} \qquad \delta_k = \begin{cases} \mathbf{1} & \text{if } k = n \\ 0 & \text{otherwise} \end{cases}$$

Explicitly:

$$
\begin{array}{ccccccccc}
 & \cdots & n+1 & n & n-1 & n-2 & \cdots \\
S^n \cong & \cdots \longrightarrow 0 \longrightarrow & \mathbb{F} \longrightarrow & 0 \longrightarrow & 0 \longrightarrow & \cdots \\
D^n \cong & \cdots \longrightarrow 0 \longrightarrow & \mathbb{F} \xrightarrow{\ \mathbf{1}\ } & \mathbb{F} \longrightarrow & 0 \longrightarrow & \cdots
\end{array}
$$

Since we consider only non-negatively graded chain complexes, $D^0 \cong S^0$.

2.2. *Homology*. The following vector spaces are called respectively the *n*-th **cycles** and the *n*-th **boundaries** of a chain complex C:

$$Z_n C := \begin{cases} C_0 & \text{if } n = 0 \\ \ker(\delta_n \colon C_n \to C_{n-1}) & \text{if } n \geq 1, \end{cases} \qquad B_n C := \mathrm{im}(\delta_{n+1} \colon C_{n+1} \to C_n)$$

Since $\delta_n \delta_{n+1} = 0$, the space of n-th boundaries $B_n C$ form a vector subspace of that of the n-th cycles $Z_n C$. The quotient $Z_n C / B_n C$ is called the n-th **homology** of C and is denoted by $H_n C$. We write ZC, BC, and HC to denote the chain complexes $\{Z_n C\}_{n \in \mathbf{N}}$, $\{B_n C\}_{n \in \mathbf{N}}$, and $\{H_n C\}_{n \in \mathbf{N}}$. Note that they all have trivial differentials. As an example, consider S^n and D^n defined in Paragraph 2.1. We have $HD^n = 0$, $H_n S^n = \mathbb{F}$, and $H_k S^n = 0$ for all $k \neq n$.

2.1. Filtered chain complexes

The symbols $[0, \infty)$ and $[0, \infty]$ denote the ordered set of non-negative reals and extended non-negative reals, respectively. Functors of the form $X \colon [0, \infty) \to \mathbf{ch}$ are also referred to as **parametrised chain complexes**. The value of X at t in $[0, \infty)$ is denoted by X^t, and $X^{s \leq t} \colon X^s \to X^t$ denotes the morphism in $\mathbf{ch}$ that X assigns to $s \leq t$. The morphism $X^{s \leq t}$ is also referred to as the **transition morphism** in X from s to t. The symbol X_n denotes the n-th degree of X, for all $t \in [0, \infty)$, i.e. a sequence of vector spaces and maps between them.

Definition 2.3. An increasing sequence $0 = \tau_0 < \cdots < \tau_k$ in $[0, \infty)$ **discretises** $X \colon [0, \infty) \to \mathbf{ch}$ if $X^{s \leq t} \colon X^s \to X^t$ may fail to be an isomorphism only when there is $a \in \{1, \ldots, k\}$ such that $s < \tau_a \leq t$. A parametrised chain complex $X \colon [0, \infty) \to \mathbf{ch}$ is called **tame** if there is a sequence that discretises it.

We are especially interested in the following class of tame parametrised chain complexes.

Definition 2.4. A tame parametrised chain complex X is called a **filtered chain complex** if, for all $s < t \in [0, \infty)$, the transition morphism $X^{s < t}$ is a monomorphism.

The following definition provides an interesting class of filtered chain complexes, parametrised by a natural number n and an element of $\Omega := \{(s, e) \in [0, \infty) \times [0, \infty] \mid s \leq e\}$.

Definition 2.5. Let $n \in \mathbf{N}$ and $(s, e) \in \Omega$. An (n-**dimensional**) **interval sphere** is a filtered chain complex isomorphic to $I^n[s, e]$, with:

$$
I^n[s, e]^t = \begin{cases} 0 & \text{if } t < s \\ S^n & \text{if } s \leq t < e \\ D^{n+1} & \text{if } e \leq t < \infty, \end{cases}
\qquad
I^n[s, e]^{u \leq t} = \begin{cases} 0 \hookrightarrow 0 & \text{if } u \leq t < s \\ 0 \hookrightarrow S^n & \text{if } u < s \leq t < e \\ \mathbf{1}_{S^n} & \text{if } s \leq u < t < e \\ S^n \hookrightarrow D^{n+1} & \text{if } u < e \leq t \\ \mathbf{1}_{D^{n+1}} & \text{if } e \leq u < \infty. \end{cases}
$$

Note that, in the previous definition, we allow for $s = e$ when $e < \infty$.

We now recall the decomposition theorem of filtered chain complexes from [11]. We remark that the structure theorem for filtered chain complexes appeared in the literature in different settings in [3,14,30,38].

Theorem 2.6 ([11, Th. 4.2]). *Any filtered chain complex is isomorphic to a finite direct sum $\bigoplus_{i=1}^l I^{n_i}[s_i, e_i)$, where l could possibly be 0. Moreover, if $\bigoplus_{i=1}^l I^{n_i}[s_i, e_i) \cong \bigoplus_{j=1}^{l'} I^{n'_j}[s'_j, e'_j)$, then $l = l'$ and there is a permutation σ of the set $\{1, \ldots, l\}$ such that $n_i = n'_{\sigma(i)}$, $s_i = s'_{\sigma(i)}$, and $e_i = e'_{\sigma(i)}$ for $1 \leq i \leq l$.*

2.7. *Relation between filtered chain complexes and persistence modules.* Applying homology to a filtered chain complex, we obtain classical persistence modules [10]. In particular, applying H_n to an n-dimensional interval sphere, we obtain either the trivial persistence module or a so called **interval module** [34]: $H_n I^n[s, e) \cong 0$ for $s = e$, and $H_n I^n[s, e) \cong \mathbb{I}_{[s,e)}$ for $s < e$. In general, by the additivity of the homology functor, we can state the following result linking the decomposition of filtered chain complexes into interval spheres to the decomposition of persistence modules into interval modules.

Proposition 2.8. *Let X be a filtered chain complex. Any decomposition of X into interval spheres induces the decomposition of the persistence module $H_n X$ into interval modules:*

$$
X \cong \bigoplus_{i=1}^l I^{n_i}[s_i, e_i) \implies H_n X \cong \bigoplus_{i \colon n_i = n, \ s_i < e_i} \mathbb{I}_{[s_i, e_i)} \,.
$$

As a consequence, any algorithm for decomposing filtered chain complexes into indecomposables provides as a by-product another way of computing the barcode decomposition of the persistence module HX, simply by forgetting all the intervals $[s_j, e_j)$ for which $s_j = e_j$.

Theorem 2.6 is an encouraging mathematical result, but if one aims to use it in computations, one needs to provide an appropriate translation from the mathematical formulation to feasible computer input. With this goal in mind, we first provide the right mathematical setting using the notion of generators and then restrict to a special class of them, the quasi-minimal sets of generators, which correspond to a feasible input – in fact, they correspond to the simplices in a simplicial complex.

2.9. *Generators.* Let X be a filtered chain complex. With a little abuse of notation, in what follows we write $I^{-1}[s,s)$ for $I^0[s,\infty)$. By tameness, there is a finite collection of interval spheres $\{I^{n_i}[s_i,s_i)\}_{i=1,\dots,m}$ that **generates** X, i.e. such that there is an epimorphism

$$\Phi \colon \bigoplus_{i=1}^{m} I^{n_i}[s_i,s_i) \to X. \tag{1}$$

The morphism Φ uniquely defines $\Phi_i \colon I^{n_i}[s_i,s_i) \to X$, for all $i=1,\dots,m$. In turn, for all $i=1,\dots,m$, each morphism $\Phi_i \colon I^{n_i}[s_i,s_i) \to X$ can be uniquely described by the vector $x_i := (\Phi_i)_{n_i+1}^{s_i}(1) \in X_{n_i+1}^{s_i}$ (see [11, Paragraph 4.5]). Therefore, $\mathcal{G} = \{x_1,\dots,x_m\}$ is also called a **set of generators of** X. On $\mathcal{G}$, we can define a **degree** function deg and an **entrance time** function ent such that, given the generator $x = \Phi_i^t(1)$, they return $\deg(x) = n$ and $\mathrm{ent}(x) = t$. A set of generators $\mathcal{G}$ is called **minimal** if there is no other set of generators $\mathcal{G}'$ with $|\mathcal{G}'| < |\mathcal{G}|$.

Retrieving a minimal set of generators for a filtered chain complex is as hard as decomposing it into interval spheres. So, it is more convenient to consider larger sets of generators. With this goal in mind, we recall that, for each $n \in \mathbf{N}$, the radical of X_n is the subfunctor $\mathrm{rad}X_n \subset X_n$ whose value $\mathrm{rad}(X_n)^t$ is the subspace of X_n^t given by the sum of all the images of $X_n^{s<t}$ for all $s < t$. A set of generators $\mathcal{G}$ of X is called **quasi-minimal** if, for all $n \in \mathbf{N}$ and $t \in [0,\infty)$, the set

$$\{x \in \mathcal{G} \mid \mathrm{ent}(x) = t \text{ and } \deg(x) = n\}$$

is a basis of $\left(X_n/\mathrm{rad}X_n\right)^t$.

As an example, take X to be an interval sphere $I^n[s,e)$, with $n \in \mathbf{N}$ and $s \leq e < \infty$. A finite set of generators of X is given by $\mathcal{G} = \{x_n^s, x_{n+1}^e\}$ with $x_n^s \in X_n^s \setminus \{0\}$, $x_{n+1}^e \in X_{n+1}^e \setminus \{0\}$ arbitrarily chosen. If $s < e$, then $\mathcal{G}$ is minimal. If $s = e$, $\mathcal{G}$ is not minimal, since x_n^s is redundant. However, it is quasi-minimal.

We use quasi-minimal sets of generators of X to give a matrix encoding of X.

2.10. *Total boundary matrix.* Let X be a filtered chain complex with differential δ, and $\mathcal{G} = \{x_1,\dots,x_m\}$ be an arbitrarily totally ordered quasi-minimal set of generators of X. From the definition of the radical of X_n, and the assumptions that X is tame and all its transition morphisms are monomorphisms, it follows that, for each $n \in \mathbf{N}$ and $t \in [0,\infty)$,

$$X_n^t \cong \bigoplus_{s \leq t \in [0,\infty)} \left(X_n/\mathrm{rad}X_n\right)^s. \tag{2}$$

By definition, a quasi-minimal set of generators $\mathcal{G}$ of X provides a basis of X_n^t. Thus, for each $x_j \in \mathcal{G}$ with $\deg(x_j) = n+1$ and $\mathrm{ent}(x_j) = t$, we can write $\delta_{n+1}^t(x_j)$ uniquely as a linear combination of elements $x_i \in \mathcal{G}$ such that $\deg(x_i) = n$ and $\mathrm{ent}(x_i) \leq t$. The total boundary matrix of X is the matrix $d = (d_j^i)$, $1 \leq i, j \leq |\mathcal{G}|$, such that d_j^i is equal to the coefficient of x_i in $\delta_{n+1}^t(x_j)$. Hereafter, we adopt the notational convention by which superscripts denote row indices and subscripts column indices. Accordingly, d^i denotes the i-th row and d_i the i-th column of d.

By construction, we can assume the rows and columns of d are labelled by the degree and entrance time of the corresponding generator in $\mathcal{G}$. The sub-matrix of d corresponding to the rows and columns associated with generators with degree $n-1$ and n, respectively, is the boundary matrix of the n-th differential δ_n of X. In the case when the total order on $\mathcal{G}$ is given first by degree, d is a diagonal block matrix with blocks given by the differential δ_n.

In practical situations, we often obtain the filtered chain complex X from a filtered simplicial complex Σ. In this case, the set of simplices of Σ provides a quasi-minimal set of generators of X, and the total boundary matrix of X coincides with the boundary matrix of Σ.

Since, in what follows, we constructively decompose a filtered chain complex by modifying its (quasi-minimal) set of generators, we now review in detail how the change of generators affects the corresponding boundary matrix.

2.11. *Change of generators.* Let X be a filtered chain complex and $\mathcal{G}$ a totally ordered quasi-minimal set of generators of X labelled by degree and entrance time. Suppose now we change one of the generators $x_i \in \mathcal{G}$ to a new element

$$\tilde{x}_i = \alpha x_i + \sum_{j \neq i} \beta_j x_j$$

where $\alpha, \beta_j \in \mathbb{F}$ are such that $\alpha \neq 0$, and $\beta_j = 0$ when $\text{ent}(x_j) > \text{ent}(x_i)$ or $\deg(x_j) \neq \deg(x_i)$. Taking $\tilde{\mathcal{G}} = \{\tilde{x}_1, \ldots, \tilde{x}_m\}$ with $\tilde{x}_k = x_k$ for $k \neq i$, and $\tilde{x}_i$ as described, we claim that $\tilde{\mathcal{G}}$ is again a quasi-minimal set of generators of X. Indeed, the change from $\mathcal{G}$ to $\tilde{\mathcal{G}}$ corresponds to a change of basis in $(X_n / \text{rad} X_n)^t$, with $n = \deg(x_i)$ and $t = \text{ent}(x_i)$. Hence, it is realised by an isomorphism $\psi \colon X \to X$, showing that $\tilde{\mathcal{G}}$ is also a set of generators of X. Moreover, in $\tilde{\mathcal{G}}$ the degree and the entrance time functions operate as follows:

$$\deg(\tilde{x}_k) = \deg(x_k) \qquad\qquad \text{for all } k = 1, \ldots, m;$$
$$\text{ent}(\tilde{x}_j) = \text{ent}(x_j) \qquad\qquad \text{for all } j = 1, \ldots, m \text{ and } j \neq i;$$
$$\text{ent}(\tilde{x}_i) = \max\{\text{ent}(x_i), \max\{\text{ent}(x_j) \mid j = 1, \ldots, m \text{ such that } j \neq i \text{ and } \beta_j \neq 0\}\} = \text{ent}(x_i).$$

Thus, for each $n \in \mathbf{N}$ and $t \in [0, \infty)$, the number of generators with degree n and entrance time t is the same in $\mathcal{G}$ and $\tilde{\mathcal{G}}$, yielding that $\tilde{\mathcal{G}}$ is quasi-minimal as well.

The new total boundary matrix $\tilde{d} = (\tilde{d}_j^i)$ of X relative to $\tilde{\mathcal{G}}$ can be deduced from the original total boundary matrix $d = (d_j^i)$ relative to $\mathcal{G}$ by Gaussian elimination. In particular, the substitution of x_i by $\tilde{x}_i = \alpha x_i$, with $\alpha \neq 0 \in \mathbb{F}$, is realized by the row operation $d^i \to (\tilde{d})^i = (\alpha)^{-1} d^i$ and by the column operation $d_i \to (\tilde{d})_i = \alpha d_i$. Similarly, the substitution of x_i by $\tilde{x}_i = x_i + \beta x_k$, with $\beta \neq 0 \in \mathbb{F}$ and $k \neq i$, is realized by the row operation $(\tilde{d})^k = d^k - \beta d^i$ and by the column operation $(\tilde{d})_i = d_i + \beta d_k$. Note that, as degrees and entrance times of generators have not changed, the rows and columns of $\tilde{d}$ are still labelled with the same degrees and entrance times as in d, respectively.

2.2. Persistence algorithms

In view of relating our algorithm for filtered chain complexes decomposition to persistence algorithms, we recall the **standard persistence algorithm** [16, Sec. VII.1], in short the SPA, and two of its runtime optimisations known, respectively, as clear and compress [5,12].

Let d be the total boundary matrix of X associated with the differential δ with respect to $\mathcal{G}$. In the SPA and its variants, the generators in $\mathcal{G}$ are sorted according to first entrance times, then degrees, and finally randomly to break the ties. This entails the fact that the row index of the lowest non-zero element in a column, called a **pivot**, corresponds to the youngest face of the column simplex. The boundary matrix is called **reduced** if there are no two non-zero columns with the same pivot. Suppose i is a pivot of column j. In this case, the generator corresponding to the i-th row gives birth to a homology class (hence it is said to be **positive**), while the generator corresponding to the j-th column causes the death of that homology class (hence it is said to be **negative**). The indices (i, j) form a **persistence pair**. The collection of all persistence pairs with $i < j$ of the reduced boundary matrix of X provides the persistence diagram of HX, i.e. its barcode decomposition into interval modules $\mathbb{I}[i, j)$ [10,16,17].

The standard persistence algorithm [16], as well as its cohomological version [14], achieve such reduction using left-to-right column operations. The worst-case runtime complexity is cubic in the number of generators (see [31] for an example where the worst-case is attained). To obviate this problem, time optimisation strategies have been introduced.

2.12. *Clear and compress [12].* The **clear** optimisation is based on the observation that if i is the pivot of a reduced column (i.e. it is positive), then during the reduction the i-th column will eventually turn out to be trivial. Thus, instead of explicitly performing all the operations to make it trivial, it can be directly cleared, i.e. set to zero. Symmetrically, the **compress** optimisation is based on the observation that if the j-th column contains a pivot (i.e. it is negative), then the j-th row cannot be a pivot. Thus, one can avoid unnecessary computation and set the j-th row directly to zero.

Since the clear strategy requires processing columns in decreasing degree order while the compress runs in increasing degree order, they cannot be easily combined. A technique to combine them is [5], where the clear is applied first, but only limitedly to pairs not exceeding a threshold persistence, followed by the compress on the remaining columns.

From the point of view of decomposing persistence modules, the clear and compress are clever runtime optimisations that take advantage of the property of differentials. On the other hand, from the point of view of decomposing filtered chain complexes, the clear and compress have a geometrical explanation corresponding to the splitting of the pair of generators of an interval sphere (see Paragraph 4.1).

3. Decomposition method for filtered chain complexes

In this section, we present a computational strategy to decompose filtered chain complexes. We begin by proving results ensuring the correctness of the algorithm, and then we provide its pseudo-code. We then run the algorithm on two examples coming from the Vietoris–Rips filtration of two point clouds to clarify it. Finally, we show how to apply the algorithm to filtered kernels in a situation where the filtered chain complex to be decomposed does not correspond to a filtered simplicial complex.

Let X be a filtered chain complex, and assume that we are given an arbitrarily totally ordered quasi-minimal set of generators $\mathcal{G} = \{x_1, x_2, \ldots, x_m\}$ of X. We denote by $d = (d_j^i)$ the total boundary matrix associated with the differential δ

with respect to $\mathcal{G}$, i.e. $\delta(x_j) = \sum_{i=1}^{m} d_j^i x_i$. All generators, and hence the corresponding rows and columns of d, are assumed to be labelled by degree and entrance time.

A pair (x_i, x_j) of generators in $\mathcal{G}$ is said to satisfy the **split conditions SC1-2-3** if

$$\textbf{SC1: } d_j^i \neq 0; \qquad \textbf{SC2: } x_i \in \arg\max_{h \,|\, d_j^h \neq 0} \text{ent}(x_h); \qquad \textbf{SC3: } x_j \in \arg\min_{h \,|\, d_h^i \neq 0} \text{ent}(x_h).$$

In plain words, x_i is any of the possibly many "youngest" elements hit by the differential of x_j, and x_j is any of the possibly many "oldest" elements whose image under δ hits x_i. As we will discuss in Paragraph 4.3, these conditions are related to the so-called apparent and emergent pairs.

The reason for the name of the above conditions is that they allow us to split off an interval sphere (see Proposition 3.2). Splitting interval spheres makes the differential more and more trivial. Thus, the strategy for the decomposition is to prune off elements to simplify the differential, similarly to what is done in [23] for single chain complexes.

If $\mathcal{G}$ is sorted as in the SPA (see Section 2.2), the existence of a pair (x_i, x_j) satisfying **SC1-2-3** is straightforwardly ensured, provided $d \neq 0$. Indeed, in that case, the generator x_j can be taken to correspond to the first non-zero column of d (which exists by the assumption $d \neq 0$), and the generator x_i can be taken to correspond to the lowest non-zero element in such column.

We now show that the existence of such a pair is still guaranteed, as long as there is at least one non-trivial boundary, even if the generators are ordered arbitrarily.

Proposition 3.1. *Let X be a filtered chain complex and $\mathcal{G} = \{x_1, x_2, \ldots, x_m\}$ a quasi-minimal set of generators of X. Then there exists a pair (x_i, x_j) of generators in $\mathcal{G}$ that satisfy the split conditions SC1-2-3 if and only if there is at least one non-trivial differential.*

Proof. If there exists such a pair, then by **SC1** X has at least one non-trivial differential.

We now prove the other implication. Since X has at least one non-trivial differential, there exists an index $i_0 \in \{1, \ldots, m\}$ and a degree $n \in \mathbb{N}$ such that $x_{i_0} \in \text{im}(\delta_n)$, implying the existence of h such that $d_h^{i_0} \neq 0$. Choose x_{j_0} in $\{\arg\min_{h|d_h^{i_0} \neq 0} \text{ent}(x_h)\}$. Observe that $\text{ent}(x_{j_0}) \geq \text{ent}(x_{i_0})$. Now, if $x_{i_0} \in \{\arg\max_{h|d_{j_0}^h \neq 0} \text{ent}(x_h)\}$, the claim is proved with $i = i_0$ and $j = j_0$. Otherwise, choose $x_{i_1} \in \{\arg\max_{h|d_{j_0}^h \neq 0} \text{ent}(x_h)\}$. We have $\text{ent}(x_{i_1}) > \text{ent}(x_{i_0})$. If $x_{j_0} \in \{\arg\min_{h|d_h^{i_1} \neq 0} \text{ent}(x_h)\}$, the claim is proved with $i = i_1$ and $j = j_0$. Otherwise, choose $x_{j_1} \in \{\arg\min_{h|d_h^{i_1} \neq 0} \text{ent}(x_h)\}$ and note that $\text{ent}(x_{j_1}) < \text{ent}(x_{j_0})$. We iterate the process, selecting at the k-th step generators two generators such that one of the following happens: *(i)* $x_{i_k} \in \{\arg\max_{h|d_{j_k}^h \neq 0} \text{ent}(x_h)\}$, so the claim is proved with $i = i_k$ and $j = j_k$; *(ii)* $x_{j_{k-1}} \in \{\arg\min_{h|d_h^{i_k} \neq 0} \text{ent}(x_h)\}$, so the claim is proved with $i = i_k$ and $j = j_{k-1}$; *(iii)* none of the above and the process iterates to step $k + 1$. Since X is tame, this procedure eventually ends, proving the claim. $\square$

We are interested in a pair satisfying **SC1-2-3** because it determines generators of an interval sphere that can be split, thus shrinking a filtered chain complex X to a smaller filtered chain complex X'. In the next proposition, we show how to identify the generators to be split using the boundary matrix of X and retrieve the boundary matrix of X' by a row Gaussian elimination on the boundary matrix of X.

Proposition 3.2. *Let X be a filtered chain complex with at least one non-trivial differential, and $\mathcal{G} = \{x_1, x_2, \ldots, x_m\}$ be a totally ordered quasi-minimal set of generators of X. Let (x_i, x_j) be a pair of generators in $\mathcal{G}$ satisfying the split conditions SC1-2-3. Then there exists a quasi-minimal set of generators $\tilde{\mathcal{G}} = \{\tilde{x}_1, \ldots, \tilde{x}_m\}$ of X for which*

1. *$\tilde{x}_j = x_j$, $\tilde{x}_i = \delta(x_j)$, and the pair $(\tilde{x}_i, \tilde{x}_j)$ satisfies the split conditions SC1-2-3;*
2. *Each $\tilde{x}_k \in \tilde{\mathcal{G}}$ is labelled with the same degree and entrance time as $x_k \in \mathcal{G}$, $k = 1, \ldots, m$;*
3. *$X \cong I^n[s, e] \oplus X'$, with $n = \deg(\tilde{x}_i)$, $s = \text{ent}(\tilde{x}_i)$, $e = \text{ent}(\tilde{x}_j)$, and X' is the filtered chain complex generated by $\mathcal{G}' = \tilde{\mathcal{G}} \setminus \{\tilde{x}_i, \tilde{x}_j\}$ with differential δ' induced by δ;*
4. *The $(m - 2) \times (m - 2)$ total boundary matrix d' of X' with respect to $\mathcal{G}'$ is obtained from the $m \times m$ total boundary matrix d of X with respect to $\mathcal{G}$ by adding, for each k such that $d_j^k \neq 0$, to its k-th row the i-th row multiplied by $-(d_j^i)^{-1} d_j^k$, and next removing its i-th and j-th columns, as well as its i-th and j-th rows.*

Proof. Recall from [11, Thm. 4.2] that a sufficient condition to split an interval sphere $I^n[s, e]$ out of a filtered chain complex X with non-trivial differential δ is to take two non-zero elements $x \in X_n^s$ and $y \in X_{n+1}^e$ that satisfy the following properties:

(i) $\delta_n(x) = 0$;

(ii) $\delta_{n+1}(y) = X_n^{s \leq e}(x)$;

(iii) $x \notin \text{im}(X_n^{t < s})$ for any $t < s$;

(iv) $y \notin \operatorname{im}(X_{n+1}^{t<e})$ for any $t < e$;

(v) $X_n^{s<t}(x) \notin \operatorname{im}(\delta_{n+1}^t)$ for any $t < e$.

Under these assumptions, X is isomorphic to $I^n[s,e] \oplus X'$ for a filtered chain complex X' uniquely determined up to isomorphism by quotienting out x and y.

The existence of a pair of generators (x_i, x_j) of $\mathcal{G}$ satisfying the split conditions **SC1-2-3**, ensured by Proposition 3.1, is not enough to guarantee property (ii). Let us therefore construct a possibly different set of quasi-minimal generators $\tilde{\mathcal{G}} := \{\tilde{x}_1, \dots, \tilde{x}_m\}$ so that the pair $(\tilde{x}_i, \tilde{x}_j)$ satisfies properties (i–v).

The first step is to define $\overline{\mathcal{G}} := \{\overline{x}_1, \dots, \overline{x}_m\}$ with

$$\overline{x}_k := \begin{cases} \sum_r d_j^r x_r & \text{if } k = i \\ x_k & \text{otherwise.} \end{cases} \tag{3}$$

In other words, we take $\overline{x}_i$ to be the differential of x_j. Let $\overline{d} = (\overline{d}_h^k)$ be the matrix of δ with respect to $\overline{\mathcal{G}}$. By the discussion in Paragraph 2.11, $\overline{\mathcal{G}}$ is a quasi-minimal set of generators. Moreover, $\deg(\overline{x}_k) = \deg(x_k)$ for all k, and $\operatorname{ent}(\overline{x}_k) = \operatorname{ent}(x_k)$ for all k since $\operatorname{ent}(\overline{x}_i) = \max\{\operatorname{ent}(x_k) \mid d_j^k \neq 0\} = \operatorname{ent}(x_i)$ by **SC2**.

The second step is to possibly change again the set of generators to $\tilde{\mathcal{G}} := \{\tilde{x}_1, \dots, \tilde{x}_m\}$ by putting $\tilde{x}_k := \overline{x}_k$ for $k = j$, and $\tilde{x}_k := \overline{x}_k - \overline{d}_k^i \overline{x}_j$ otherwise. $\tilde{\mathcal{G}}$ is still a quasi-minimal set of generators, again by the discussion of Paragraph 2.11, and $\deg(\tilde{x}_k) = \deg(\overline{x}_k)$ and $\operatorname{ent}(\tilde{x}_k) = \operatorname{ent}(\overline{x}_k)$ for all k. Moreover, by construction, also $(\tilde{x}_i, \tilde{x}_j)$ satisfies **SC1-2-3**. So, claims 1 and 2 are verified.

To prove claim 3, it is sufficient to show that properties (i–v) are satisfied by taking $x := \tilde{x}_i$ and $y := \tilde{x}_j$. Property (i) follows by **SC1-2** of $(\tilde{x}_i, \tilde{x}_j)$. Property (ii) holds by construction since $\delta_n(\tilde{x}_j) = \tilde{x}_i$. Finally, properties (iii–v) follow by **SC2-3** of $(\tilde{x}_i, \tilde{x}_j)$.

To prove claim 4, let us examine how the change of generators from $\mathcal{G}$ to $\tilde{\mathcal{G}}$ via $\overline{\mathcal{G}}$ modifies the total boundary matrix (see Paragraph 2.11). We see that

$$\overline{d}_j = (0, \dots, 0, \underset{i\text{-th}}{1}, 0, \dots, 0)^T \tag{4}$$

$$\overline{d}^k = d^k - d_j^k d^i \qquad \text{for } k \neq i \tag{5}$$

$$\overline{d}_i = 0 \tag{6}$$

The equations (4) and (5) are induced by the change of generators (3), and equation (6) follows from $\delta^2 = 0$. Indeed, $\delta^2 = 0$ if and only if $\overline{d}^2 = 0$, which implies $(\overline{d}^2)_j^k = 0$ for all k. On the other hand, by construction we have $\overline{d}_j^i = 1$ and $\overline{d}_j^k = 0$ for $k \neq i$. So, $(\overline{d}^2)_j^k = \sum_h \overline{d}_h^k \overline{d}_j^h = \overline{d}_i^k$ by (4). Hence, $\overline{d}_i^k = 0$ for all k, and thus the column $\overline{d}_i$ is trivial.

Next, let $\tilde{d}$ be the matrix associated with δ with respect to $\tilde{\mathcal{G}}$. The change of the generators from $\overline{\mathcal{G}}$ to $\tilde{\mathcal{G}}$ affects only the j-th row, so that $\tilde{d}^j = \overline{d}^j + \overline{d}_k^i \overline{d}^k$, and the columns $\tilde{d}_k$ with $k \neq j$ and $\overline{d}_k^i \neq 0$. In particular, equations (4)–(6) hold also for $\tilde{d}$, and in addition we have:

$$\tilde{d}^i = (0, \dots, 0, \underset{j\text{-th}}{1}, 0, \dots, 0) \tag{7}$$

$$\tilde{d}^j = 0 \tag{8}$$

Equality (7) follows by the choice of the generators. Indeed, we have $\tilde{d}_k = \overline{d}_k - \overline{d}_h^i \overline{d}_j$ for $k \neq j$, and thus, by (4), $\tilde{d}_k^i = \overline{d}_k^i - \overline{d}_k^i \overline{d}_j^i = \overline{d}_k^i - \overline{d}_k^i = 0$ for all $k \neq j$. Equality (8) follows by similar arguments as in the proof of (6). Claim 4 now follows by observing that the total boundary matrix with respect to $\mathcal{G}' = \tilde{\mathcal{G}} \setminus \{\tilde{x}_i, \tilde{x}_j\}$ corresponds to $\tilde{d}$ with the i-th row and column and the j-th row and column deleted. $\square$

We stress that in the proof we relied on the assumption $\delta^2 = 0$. Without this assumption, the i-th column and the j-th row need not be zero and thus cannot be directly deleted. This is why we cannot generalise such an argument to the decomposition of two-parameter persistence modules.

Moreover, we remark that, in actual computations, it is sufficient to perform only the row reduction to determine the boundary matrix of the complex X' that remains after the splitting. Indeed, the column reduction is necessary for the proof of the correctness of the procedure but does not affect the retrieving of the boundary of X' (see Section 3.1).

If a generator is a cycle and does not contribute to the differential, its corresponding interval sphere can be directly split.

Proposition 3.3. *Let X be a filtered chain complex and $\mathcal{G} = \{x_1, x_2, \dots, x_m\}$ an ordered quasi-minimal set of generators of X. Let d be the boundary matrix associated with δ with respect to $\mathcal{G}$. For each generator x_i in $\mathcal{G}$ such that (a) $d^i = 0$, and (b) $d_i = 0$, it holds that:*

1. X is isomorphic to $I^n[s, \infty) \oplus X'$ for $n = \deg(x_i)$, with $s = \mathrm{ent}(x_i)$, and X' is the filtered chain complex whose set of generators is $\mathcal{G}' = \mathcal{G} \setminus \{x_i\}$, with same degrees and entrance times of $\mathcal{G}$, and differential induced by δ;
2. The $(m-1) \times (m-1)$ total boundary matrix d' of X' with respect to $\mathcal{G}'$ is obtained from the $m \times m$ total boundary matrix d of X with respect to $\mathcal{G}$ by removing its i-th row and column.

Proof. Consider the following exact sequence:

$$0 \longrightarrow I^n[s, \infty) \overset{\iota}{\longrightarrow} X \longrightarrow X' \cong \left. X \middle/ \mathrm{im}(\iota) \right. \longrightarrow 0 \tag{9}$$

where ι is such that $1 \in (I^n[s, \infty))_s^n \mapsto x_i$, with $s = \mathrm{ent}(x_i)$. The hypotheses *(a)* and *(b)* are equivalent to fact that the differential of x_i is trivial and $x_i \notin \mathrm{im}(\delta)$. Thus, they guarantee the existence of a retract $\rho \colon X \to I^n[s, \infty)$. We can then apply the characterisation of split exact sequences [28, Prop. 4.3] to (9) and conclude that $X \cong I^n[s, \infty) \oplus X'$, proving the first claim. The second claim follows immediately since $d^i = d_i = 0$ by hypothesis. □

We are now ready to present the pseudo-code for the decomposition of filtered chain complexes. Algorithm 1 takes as input an ordered quasi-minimal set of generators of X and returns as output the list of interval sphere direct summands of X.

Algorithm 1: Interval sphere decomposition of filtered chain complexes.

I: Totally ordered quasi-minimal set of generators $\mathcal{G}$ of X, labelled by degree and entrance time
O: List of interval spheres
1 List $= \emptyset$
2 $d = \mathrm{BOUNDARY}(\mathcal{G})$
3 while *∃ i such that $d^i \neq 0$* **do**
4 $i, j = \mathrm{PAIR}(d, i)$
5   Append $I^{\deg(i)}[\mathrm{ent}(i), \mathrm{ent}(j))$ to List
6 $\mathrm{SPLIT}(i, j, d)$
7 for *all indices i of remaining rows in d* **do**
8   Append $I^{\deg(i)}[\mathrm{ent}(i), \infty)$ to List
9 Return List

Algorithm 1 hinges on three methods: BOUNDARY, SPLIT, and PAIR. BOUNDARY generates the boundary matrix d from a minimal set of generators and is not discussed here as it depends on the data structure at hand. The boundary matrix d is then a global variable and will be updated at each call of SPLIT. SPLIT performs the splitting by reducing the boundary matrix, as stated in Proposition 3.2(*4*). PAIR selects the pair of generators to be split following the constructive proof of Proposition 3.1, which guarantees that the while loop eventually terminates. We underline that PAIR does not need columns and rows of d to be sorted by an a-priori-chosen order, differently to what happens for usual persistence algorithms. For actual implementation, the most efficient approach is likely to order the generators as in the SPA (see Section 2.2). Indeed, with that order, the method PAIR is much more efficient than the pseudo-code we present below. However, from a theoretical point of view, we show that we can reduce the boundary matrices and obtain the barcode decomposition without an a-priori-fixed order. Finally, we note that Proposition 3.1, and consequently the PAIR method, would also work by exchanging the role of rows and columns, i.e. by first picking an arbitrary column j, and then looking for a row i to pair it with.

Method 2: SPLIT(i, j, d).

I: Boundary matrix d, indices i, j
O: Updated boundary matrix d
1 for *k with $d_j^k \neq 0$* **do**
2   Add to the k-th row the i-th row multiplied by $-(d_j^i)^{-1} d_j^k$
3 Delete the j-th row from d
4 Delete the i-th row and the j-th column from d
5 Delete the i-th column from d
6 Return d

In conclusion, Proposition 3.2 and Proposition 3.3 guarantee the overall correctness of Algorithm 1 to obtain the decomposition of a filtered chain complex, and hence, as a by-product, the barcode decomposition of a one-parameter persistence module.

Method 3: PAIR(d, i).

I: Boundary matrix d, index i of a non-zero row of d
O: i, j satisfying **SC1-2-3**

1 Pick $j \in \arg\min_{h|d_h^i \neq 0} \text{ent}(h)$

2 **while** TRUE **do**

3   **if** $i \in \arg\max_{h|d_j^h \neq 0} \text{ent}(h)$ **then**

4   Return i, j

5   **else**

6   Pick $k \in \arg\max_{h|d_j^h \neq 0} \text{ent}(h)$

7          $i = k$

8   **if** $j \in \arg\min_{h|d_h^i \neq 0} \text{ent}(h)$ **then**

9   Return i, j

10   **else**

11   Pick $k \in \arg\min_{h|d_h^i \neq 0} \text{ent}(h)$

12         $j = k$

3.1. Examples

To illustrate the above methods, we provide two running examples built on the Vietoris–Rips complexes of two point clouds. Consider the l_∞ metric on the following two spaces: (*a*) four points on a line and (*b*) four points in a grid. We call these points x, y, z, w as illustrated in Figure 1.

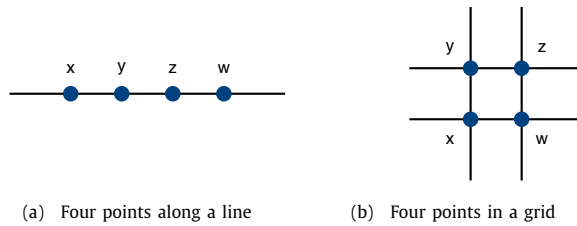(a) Four points along a line   (b) Four points in a grid

Fig. 1. The two point clouds used in Section 3.1

The corresponding distance matrices are given in (10) (hereafter, red colour is used for the case (*a*) and blue colour for (*b*)).

$$
D^a = \begin{array}{c} \\ x \\ y \\ z \\ w \end{array}
\begin{array}{c} \begin{array}{cccc} x & y & z & w \end{array} \\
\begin{bmatrix} 0 & 1 & 2 & 3 \\ 1 & 0 & 1 & 2 \\ 2 & 1 & 0 & 1 \\ 3 & 2 & 1 & 0 \end{bmatrix} \end{array}, \quad
D^b = \begin{array}{c} \\ x \\ y \\ z \\ w \end{array}
\begin{array}{c} \begin{array}{cccc} x & y & z & w \end{array} \\
\begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix} \end{array}. \tag{10}
$$

We illustrate the decomposition Algorithm 1 for the filtered chain complexes whose quasi-minimal sets of generators are given by the simplices of the Vietoris–Rips complexes on the two finite metric spaces given by these two matrices. First, we find a pair of generators to split using PAIR, and then we prune them off according to SPLIT. As a field, we choose to work in $\mathbb{Z}/2\mathbb{Z}$. For simplicity, we will not display the total boundary matrices but only the block of degree 2. As a consequence, we will not perform Line 3 and Line 5 of Method 2 and we will shrink the matrix by only one column and one row.

As by Line 2 of Algorithm 1, we first build the boundary matrices ∂_2^a and ∂_2^b, whose columns correspond to the generators of degree 2 and whose rows correspond to generators of degree 1. Observe that, since by Proposition 3.2 we can use any total order, we can pick the lexicographic order for the simplices so that the matrices have the same elements (i.e. $\partial_2^a = \partial_2^b$ – even if the labels of the rows and columns given by the entrance times will differ. In the following matrix, the most-external labels denote the simplices, and the innermost labels denote their entrance times.

$$
\partial_2^a = \begin{array}{c} \\ xy \\ xz \\ xw \\ yz \\ yw \\ zw \\ \end{array}
\begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 1 \\ 2 \\ 1 \\ \text{Line} \end{array}
\begin{array}{cccc} \begin{array}{cccc} xyz & xyw & xzw & yzw \\ 1 & 1 & 1 & 1 \end{array} & & & \\ \left(\begin{array}{cccc} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{array} \right) & \begin{array}{c} \text{Grid} \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{array} \\ \begin{array}{cccc} 2 & 3 & 3 & 2 \end{array} \end{array} = \partial_2^b
$$

Let us now use PAIR to find a row and a column to split. As required, we begin with a non-zero row: we choose the top one. We now need to look at the entrance time labels of the non-zero elements in row xy and choose one of the smallest ones (Line 1 of PAIR). We pick column xyz for both matrices. Next, we need to check if row xy has one of the greatest entrance times among the rows of the non-zero elements in column xyz. This is true for (b) but not for (a), as the one with the greatest entrance time is row xz. Thus, we exit the loop according to Line 3 of PAIR with the generators (xy, xyz) for (b) and we repeat the procedure for (a). Taking xz in place of xy, we see that condition of Line 3 of PAIR is met by the pair of simplices (xz, xyz) for (a). Thus, the routine PAIR ends, and we have the generators of the interval sphere to be split also for (a), namely xz and xyz.

Next, we apply the SPLIT method. This means that we first need to add row xz to rows xy and yz in ∂_2^a, and row xy to rows xz and yz in ∂_2^b (Line 2 of Method 2). The matrices after these operations turns out to be:

$$
\partial_2^a = \begin{array}{c} \\ xy \\ xz \\ xw \\ yz \\ yw \\ zw \\ \end{array}
\begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 1 \\ 2 \\ 1 \\ \text{Line} \end{array}
\begin{array}{c} \begin{array}{cccc} xyz & xyw & xzw & yzw \end{array} \\ \left(\begin{array}{cccc} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{array} \right) \\ \begin{array}{cccc} 2 & 3 & 3 & 2 \end{array} \end{array}
\qquad
\partial_2^b = \begin{array}{c} \\ \\ xy \\ xz \\ xw \\ yz \\ yw \\ zw \end{array}
\begin{array}{c} \begin{array}{cccc} xyz & xyw & xzw & yzw \\ 1 & 1 & 1 & 1 \end{array} \\ \left(\begin{array}{cccc} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{array} \right) \begin{array}{c} \text{Grid} \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{array} \end{array}
$$

At this point, we are in the position to remove row xz, resp. xy, and column xyz from ∂_2^a, resp. ∂_2^b, thus shrinking the boundary matrices (Line 4 of SPLIT). Finally, we obtain the interval sphere $I^1[1,1]$ for the grid point-cloud and $I^1[2,2]$ for the line one. Indeed, these are the entrance time labels for row xz, resp. xy, and column xyz in the line case (resp. the grid case). Note that both these interval spheres have zero homology: their difference goes undetected at the homological level. The degree 2 boundary matrices now are:

$$
\partial_2^a = \begin{array}{c} \\ xy \\ xw \\ yz \\ yw \\ zw \\ \end{array}
\begin{array}{c} \\ 1 \\ 3 \\ 1 \\ 2 \\ 1 \\ \text{Line} \end{array}
\begin{array}{c} \begin{array}{ccc} xyw & xzw & yzw \end{array} \\ \left(\begin{array}{ccc} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \end{array} \right) \\ \begin{array}{ccc} 3 & 3 & 2 \end{array} \end{array}
\qquad
\partial_2^b = \begin{array}{c} \\ \\ xz \\ xw \\ yz \\ yw \\ zw \end{array}
\begin{array}{c} \begin{array}{ccc} xyw & xzw & yzw \\ 1 & 1 & 1 \end{array} \\ \left(\begin{array}{ccc} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \end{array} \right) \begin{array}{c} \text{Grid} \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{array} \end{array}
$$

and we can look for the next pair to split. After completely running Algorithm 1, also in the other degrees, the decomposition of the filtered chain complexes turns out to be

$$
\text{(a)} \bigoplus_{i=1}^{3} I^0[0,1) \oplus \bigoplus_{i=1}^{2} I^1[2,2) \oplus I^1[3,3) \oplus I^2[3,3), \quad \text{(b)} \bigoplus_{i=1}^{3} I^0[0,1) \oplus \bigoplus_{i=1}^{3} I^1[1,1) \oplus I^2[1,1).
$$

3.2. Filtered kernels

In the previous section, we applied our algorithm on filtered Vietoris–Rips complexes. However Algorithm 1 is more general and can be applied to arbitrary filtered chain complexes. To convey the idea, in this section, we illustrate one such application. We consider the kernels of morphisms between filtered chain complexes, which we call **filtered kernels**. Note that the homology of filtered kernels is generally different from kernel persistence [13,33,37] as $\ker Hf \not\cong H \ker f$ in general.

3.4. *Simplicial maps and induced morphisms of filtered chain complexes.* Let $f : K \to L$ be a simplicial map, and assume filtrations on K and L, respectively, are given. To obtain quasi-minimal set of generators, as in Paragraph 2.9, we label each simplex

σ with a degree $\deg(\sigma)$ (given by its simplicial dimension) and an entrance time $\text{ent}(\sigma)$ (given by the filtration). Since f is simplicial, for every simplex $\sigma \in K$, we have $\text{ent}(\sigma) \geq \text{ent}(f(\sigma))$ and $\deg(\sigma) \geq \deg(f(\sigma))$. We can now take X and Y to be the two filtered chain complexes induced by the given filtrations on K and L, respectively. Then the map f induces a morphism of filtered chain complexes, called again f, defined by

$$f_n^t := \begin{cases} X^{\text{ent}(f(\sigma)) \leq \text{ent}(\sigma)}(f(\sigma)) & \text{if } \dim f(\sigma) = \dim(\sigma) = n \\ 0 & \text{otherwise}. \end{cases}$$

The filtered chain complex given by the kernel of f, hence called filtered kernel and denoted by $\ker f$, is defined by $(\ker f)_n^t := \ker f_n^t$ for all $n \in \mathbb{N}$ and $t \in [0, \infty)$, with the transition morphisms and differentials being those induced by X. Note that we can analogously define the filtered image $\text{im} f$ as $(\text{im} f)_n^t := \text{im} f_n^t$ and the transition morphisms and differentials induced by Y. As for filtered kernels, this is in general different from image persistence [13,33,37].

Our goal is to use quasi-minimal sets of generators of K and L, precisely their simplices, to describe a quasi-minimal set of generators of the filtered kernel of f, in the case when f admits a section. This way, Algorithm 1 can be applied to compute a decomposition of $\ker f$.

3.5. *Quasi-minimal set of generators for* $\ker f$. We now fix a simplicial map $f : K \to L$ between two filtered simplicial complexes admitting a section $s : L \to K$ (see [27] for conditions that guarantee this requirement). In particular, f is an epimorphism. Note that, for such s, for every $\tau \in L$, $\text{ent}(\tau) = \text{ent}(s(\tau))$ and $\deg(\tau) = \deg(s(\tau))$. The section s induces a morphism of filtered chain complexes denoted again by s (Paragraph 3.4). We now define a quasi-minimal set of generators for $\ker f$ (see Paragraph 2.9): for every $n \in \mathbb{N}$ and $t \in [0, \infty)$, a base of $\ker f_n^t$ is given by

$$V = \{\sigma - sf(\sigma) \mid \deg(\sigma) = n, \ \text{ent}(\sigma) = t \text{ and } \sigma \notin \text{im}(s)\}.$$

Indeed, since f is an epimorphism and s its section, we have that the dimension of the space generated by V is equal to

$$\dim \text{span}(V) = \dim X_n^t - \dim(\text{im} s_n^t) = \dim X_n^t - \dim(\text{im} f_n^t) = \dim(\ker f_n^t),$$

where the first equality holds because $\sigma \notin \text{im}(s)$ implies $\sigma \neq sf(\sigma)$. Thus, it is left to prove that the elements in V are linearly independent as vectors. But this follows directly from the fact that each element is given by the difference between $\sigma \notin \text{im}(s)$ and something which is in the image of s.

We can thus easily compute the quasi-minimal set of generators of $\ker f$ from the quasi-minimal sets of generators of K and L, once a section s is known. Next, we show that retrieving the boundary matrix of $\ker f$ requires almost no computation.

3.6. *Boundary matrix of* $\ker f$. To retrieve the boundary matrix of $\ker f$, we denote by $\ker f_n$ the parametrised vector space generated by all the elements in $\ker f$ with degree n. Consider a quasi-minimal generator $\sigma - sf(\sigma)$, with $\sigma \in K$. Assume the boundary of σ in X is $\sum_{i=0}^n (-1)^i \tau_i$. Then the boundary of $\sigma - sf(\sigma)$ is given by:

$$\delta(\sigma - sf(\sigma)) = \delta\sigma - sf(\delta\sigma) = \sum_{i=0}^n (-1)^i \tau_i - sf(\sum_{i=0}^n (-1)^i \tau_i) = \sum_{i=0}^n (-1)^i (\tau_i - sf(\tau_i)),$$

where s and f are considered as morphisms of filtered chain complexes (Paragraph 3.4). Since the elements of the form $(\tau_i - sf(\tau_i))$ either belong to a basis of $\ker f_{n-1}$ (if $\tau_i \neq sf(\tau_i)$) or are equal to zero, the boundary matrix ∂ of $\ker f$ is obtained from that of X removing the columns and rows corresponding to simplices $\sigma \in K$ for which $\sigma = sf(\sigma)$.

In conclusion, with our decomposition algorithm, we can efficiently decompose $\ker f$, which is a filtered chain complex that does not come from a simplicial complex, showing that the algorithm can be applied in more general scenarios.

4. The structural geometry behind the decomposition

One of the contributions of our algorithm is to highlight the geometry behind the SPA (see Section 2.2). Indeed, this algorithm, together with its variants such as [4–6,14,18], works at the chain complex level, but it is usually described at the persistence module level.

4.1. *Relation with persistence algorithms.* Each interval sphere $I^n[s, e)$ has two generators, namely the one of the sphere S^n at step s and one of the disk D^{n+1} capping such sphere at step e. These generators form the pair that in the SPA is given by the pivot pairing. As explained in Paragraph 2.7, by forgetting (resp. retaining) the interval spheres whose generators

have the same entrance time, i.e. the contractible parts, we can retrieve the interval modules (resp. spheres) decomposition from the interval spheres (resp. modules) decomposition. Thus, our algorithm explains the geometry behind the SPA, as we explicitly split the corresponding spheres and disks.

Moreover, we can now see that the clear and compress optimisations are implicit methods to quotient out both the generators: in the former the identification of the generator of D^{n+1} allows us to skip the reduction of the one of S^n, while in the latter the opposite happens. In our algorithm, this is explicit: since the two generators generate the same indecomposable, we have to remove both in the decomposition. We can read the compress and clear strategy in our pseudo-code at Line 3 and at Line 5 of Method 2, respectively. Thus, our method provides a geometric explanation of the optimisations at the chain complex level: rather than mere strategic optimisations, they are structural.

4.2. *Row versus column reductions.* The SPA computes the barcode by reducing the boundary matrix by column operations, thus performing operations on the chains. On the other hand, the authors of [14] obtain the barcode by reducing (still by column operations) the coboundary matrix, thus performing operations on the cochains. According to [6], the use of cohomology with respect to homology turns out to be particularly convenient, when combined with the clear strategy, in the computation of the barcode of Vietoris–Rips complexes.

To properly dualise the optimisation obtained in cohomology by the clear strategy, the boundary matrix has to be reduced via row rather than column operations [24]. This is precisely what Algorithm 1 does. In doing so, when generators are primarily sorted by degree, the combination of the row reductions with the compress strategy achieves in homology the same advantages as in cohomology with the clear, for Vietoris–Rips complexes.

In [4], an explicit calculation quantifies the computational speed-up that the column reduction with clear on the coboundary matrix provides in the barcode computation of Vietoris–Rips complexes. We now display the analogous computation for row operations in homology. We underline that what is computed is the number of iterations. Each iteration has to be multiplied by the reduction cost, which depends on the algorithm used.

Let X be a filtered Vietoris–Rips complex, whose quasi-minimal set of generators is ordered first by degree and then by a total order of choice. Let v be the number of generators in degree 0, i.e. the number of vertices, N the maximal non-trivial degree of X, and d_n, for $1 \le n \le N+1$, the blocks of the boundary matrix d of X. To reduce d without compress, we need to process one row per generator from degree 0 to N. In total, this sums up to

$$\sum_{n=0}^{N} \underbrace{\binom{v}{n+1}}_{\dim(C_n X)} = \sum_{n=0}^{N} \underbrace{\binom{v-1}{n}}_{\dim(B_{n-1} X)} + \sum_{n=0}^{N} \underbrace{\binom{v-1}{n+1}}_{\dim(Z_n X)}, \tag{11}$$

where the equation follows by the fact that the final step of a Vietoris–Rips filtration is a full-complex, the definition of homology, and the rank-nullity theorem, see also [4, Sec. 3.3]. Note that $\dim(B_{n-1} X)$ is the number of negative generators of degree n. Applying the compress strategy, since the negative rows of d_n, for $n > 0$, are removed when processing d_{n-1}, (11) decreases to

$$\underbrace{\binom{v-1}{0}}_{\dim(B_{-1} X)} + \sum_{n=0}^{N} \underbrace{\binom{v-1}{n+1}}_{\dim(Z_n X)} = 1 + \sum_{n=0}^{N} \binom{v-1}{n+1} = 1 + \sum_{n=1}^{N+1} \binom{v-1}{n}$$

that coincides with the number of generators processed in [4].

4.3. *Apparent and emergent pairs.* According to [4], in the SPA where rows and columns of the boundary matrix are ordered first by entrance time and then by degree, the pairs of indices (i, j) such that the lowest non-zero element of the j-th column is in the i-th row, and the leftmost non-zero element of the i-th row is in the j-th column, are called **apparent pairs** (see [4, Sec. 3.6] for an overview of the use of apparent pairs in the computation of persistent homology). Thus, the apparent pairs are precisely given by pairs of generators (x_i, x_j) that satisfy the split conditions **SC1-2-3** before the beginning of the reduction. In other words, they are all and only the pairs that always exit the loop in PAIR at the first iteration. Note that the generators in an apparent pair depend on the chosen total order: there can be more than one pair of generators with the same degree and entrance time, and the order between them decides which one belongs to the apparent pair. The interval decomposition is, clearly, independent from this choice.

More generally, an **emergent (co)facet pair** is a pair of generators (x_i, x_j) that get split and such that x_i is the youngest facet of x_j (resp. x_j is the oldest cofacet of x_i). Thus, emergent pairs are pairs of generators that immediately exit the loop in PAIR, when PAIR is called after at least one iteration.

5. Discussion and conclusions

In this work, we analysed persistence algorithms based on matrix reductions from the standpoint of filtered chain complexes. Using this approach, we have been able to geometrically explain the decomposition as well as the clear and compress

optimisations. Nevertheless, several questions remain open. For example, a more efficient implementation of the PAIR method, ideally tailored for different chain complexes, needs to be studied to see if it can provide some speed-up in computation. One could also ask if, due to the independence of the order, there is a more efficient way to sort the generators to allow for quick access to the memory or a faster way to calculate the (co)boundary.

We also introduced filtered kernels and provided a setting in which they can be decomposed with our algorithm. This setting can be realised in different situations. For example, suppose we have a distance space M that we want to enlarge by glueing non-empty fibers to its points. The resulting distance space M' is called an extension of M if the projection $f : M' \to M$ is 1-Lipschitz. Any such extension leads to a split short exact sequence $0 \to \ker \to X \to Y \to 0$, where X and Y are the filtered chain complexes built from filtrations on M' and M, respectively. Thus, in such a setting, the decomposition of X can be obtained from the decompositions of Y and ker, where the latter may not come, in any natural way, from a filtered simplicial complex. Decompositions of filtered chain complexes of all extensions of M can be analysed in this way, without the need to recomputing the decomposition of Y, possibly leading to more effective calculations.

Yet another aspect to be investigated is the information carried by the points on the diagonal. There is some empirical evidence that these points are not always noise. In the example in Section 3.1, we see that interval modules do not distinguish between the two metric spaces, whereas interval spheres do. This indicates that filtered chain complexes retain more geometric information that would be lost applying homology. We also have that, in the case of a Vietoris–Rips complex, the latest-appearing interval sphere in degree 1 has an interval of length zero, and the value of the starting point of the interval is the diameter of the point cloud. For another example, the authors of [40] noted that the presence and multiplicity of points on and very near the diagonal of persistence diagrams obtained from studying the diffraction of different materials are related to the density of the materials (see Fig. 11 (A) and (B) in [40]). Moreover, in [20], the zero-length intervals are necessary to ensure the proper reconstruction of the simplicial complex. These different examples suggest that the zero-length intervals (which are the points on the diagonal of a persistence diagram) carry some meaningful information. We believe that they should be investigated in a more structured study, and we hope that our algorithm will be helpful to this end.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

References

[1] Jiří Adámek, Jiří Rosický, Locally Presentable and Accessible Categories, London Mathematical Society Lecture Note Series, vol. 189, Cambridge University Press, Cambridge, 1994.
[2] Henry Adams, Andrew Tausz, Mikael Vejdemo-Johansson, Javaplex: a research software package for persistent (co)homology, in: Mathematical Software – ICMS 2014, in: Lecture Notes in Computer Science, vol. 8592, Springer Berlin Heidelberg, 2014, pp. 129–136.
[3] Serguei Barannikov, The Framed Morse Complex and Its Invariants, Advances in Soviet Mathematics, vol. 21, American Mathematical Society, 1994, pp. 93–115.
[4] Ulrich Bauer, Ripser: efficient computation of Vietoris–Rips persistence barcodes, Journal of Applied and Computational Topology 5 (2021) 391–423.
[5] Ulrich Bauer, Michael Kerber, Jan Reininghaus, Clear and compress: computing persistent homology in chunks, in: Topological Methods in Data Analysis and Visualization III, 2013, pp. 103–117.
[6] Ulrich Bauer, Michael Kerber, Jan Reininghaus, Hubert Wagner, Phat – persistent homology algorithms toolbox, J. Symbolic Comput. 78 (2017) 76–90.
[7] Francisco Belchi, Mariam Pirashvili, Joy Conway, Michael Bennett, Ratko Djukanovic, Jacek Brodzki, Lung topology characteristics in patients with chronic obstructive pulmonary disease, Scientific Reports 8 (1) (2018-03-28) 5341.
[8] Gunnar Carlsson, Topology and data, Bull. Amer. Math. Soc. (N.S.) 46 (2) (2009) 255–308.
[9] Gunnar Carlsson, Vin de Silva, Zigzag persistence, Found. Comput. Math. 10 (4) (2010) 367–405.
[10] Gunnar Carlsson, Afra Zomorodian, Computing persistent homology, Discrete Comput. Geom. 33 (2) (2005) 249–274.
[11] Wojciech Chachólski, Barbara Giunti, Claudia Landi, Invariants for tame parametrised chain complexes, Homology, Homotopy, and Applications 23 (2) (2021) 183–213.
[12] Chao Chen, Michael Kerber, Persistent homology computation with a twist, in: Proceedings 27th European Workshop on Computational Geometry, 2011, pp. 1–4.
[13] David Cohen-Steiner, Herbert Edelsbrunner, John Harer, Dmitriy Morozov, Persistent homology for kernels, images, and cokernels, in: Proceedings of the Twentieth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA '09, Society for Industrial and Applied Mathematics, USA, 2009, pp. 1011–1020.
[14] Vin de Silva, Dmitriy Morozov, Mikael Vejdemo-Johansson, Dualities in persistent (co)homology, Inverse Problems 27 (12) (2011) 124003, 17.
[15] Cecil Jose A. Delfinado, Herbert Edelsbrunner, An incremental algorithm for Betti numbers of simplicial complexes on the 3-sphere, Computer Aided Geometric Design 12 (7) (1995) 771–784.

[16] Herbert Edelsbrunner, John L. Harer, Computational Topology: An Introduction, vol. 69, American Mathematical Society, Providence, RI, 2010.
[17] Herbert Edelsbrunner, David Letscher, Afra Zomorodian, Topological persistence and simplification, Discrete & Computational Geometry 28 (4) (Nov 2002) 511–533.
[18] Herbert Edelsbrunner, Katharina Ölsböck, Tri-partitions and bases of an ordered complex, Discrete & Computational Geometry 64 (3) (Oct 2020) 759–775.
[19] Emerson G. Escolar, Yasuaki Hiraoka, Persistence modules on commutative ladders of finite type, Discrete Comput. Geom. 55 (1) (2016) 100–157.
[20] Brittany Terese Fasy, Samuel Micka, David L. Millman, Anna Schenfisch, Lucia Williams, A faithful discretization of the augmented persistent homology transform, arXiv:1912.12759, 2022.
[21] Massimo Ferri, Patrizio Frosini, Alessandro Verri, Claudio Uras, On the use of size functions for shape analysis, Biological Cybernetics 70 (2) (Dec 1993) 99–107.
[22] Patrizio Frosini, Measuring shapes by size functions, in: David P. Casasent (Ed.), Intelligent Robots and Computer Vision X: Algorithms and Techniques, in: International Society for Optics and Photonics, vol. 1607, SPIE, 1992, pp. 122–133.
[23] Étienne Gallais, Combinatorial realization of the thom-smale complex via discrete Morse theory, Annali della Scuola Normale Superiore di Pisa - Classe di Scienze, Ser. 5 9 (2) (2010) 229–252.
[24] Barbara Giunti, Notes on pivot pairings, in: Proceedings of the 37th European Workshop on Computational Geometry, 2021, pp. 11:1–11:6.
[25] Barbara Giunti, Jānis Lazovskis, Database of TDA real-world applications, Available at www.zotero.org/groups/tda-applications, 2020.
[26] Fei Jiang, Takeshi Tsuji, Tomoyuki Shirai, Pore geometry characterization by persistent homology theory, Water Resources Research 54 (6) (2018) 4150–4163.
[27] Yukihiro Kodama, On a right inverse mapping of a simplicial mapping, Proceedings of the Japan Academy 33 (3) (1957) 136–138.
[28] Saunders MacLane, Homology, first edition, Die Grundlehren der mathematischen Wissenschaften, vol. 114, Springer-Verlag, Berlin-New York, 1967.
[29] Clément Maria, Jean-Daniel Boissonnat, Marc Glisse, Mariette Yvinec, The GUDHI library: simplicial complexes and persistent homology, in: Mathematical Software – ICMS 2014, in: Lecture Notes in Computer Science, vol. 8592, Springer Berlin Heidelberg, 2014, pp. 167–174.
[30] Killian Meehan, Andrei Pavlichenko, Jan Segert, On the structural theorem of persistent homology, Discrete and Computational Geometry 62 (2017) 945–989.
[31] Dmitriy Morozov, Persistence algorithm takes cubic time in worst case, in: BioGeometry News, Dept. Comput. Sci., Duke Univ., 2005.
[32] Dmitriy Morozov, Dionysus, a C++ library for computing persistent homology, 2007.
[33] Dmitriy Morozov, Homological Illusions of Persistence and Stability, PhD thesis, Duke University, 2008.
[34] Steve Y. Oudot, Persistence Theory: From Quiver Representations to Data Analysis, Mathematical Surveys and Monographs, vol. 209, American Mathematical Society, Providence, RI, 2015.
[35] Alessandra di Pierro, Stefano Mancini, Laleh Memarzadeh, Riccardo Mengoni, Homological analysis of multi-qubit entanglement, EPL (Europhysics Letters) 123 (3) (2018-09) 30006.
[36] Vanessa Robins, Towards computing homology from finite approximations, in: Proceedings of the 14th Summer Conference on General Topology and Its Applications, Brookville, NY, 1999, vol. 24, 1999, pp. 503–532.
[37] Álvaro Torras Casas, Distributing persistent homology via spectral sequences, arXiv:1907.05228, 2020.
[38] Michael Usher, Jun Zhang, Persistent homology and Floer–Novikov theory, Geometry and Topology 20 (2016) 3333–3430.
[39] Kelin Xia, Guo-Wei Wei, Persistent homology analysis of protein structure, flexibility, and folding, International journal for Numerical Methods in Biomedical Engineering 30 (8) (2014) 814–844.
[40] Onodera Yohei, Kohara Shinji, Tahara Shuta, Atsunobu Masuno, Inoue Hiroyuki, Motoki Shiga, Akihiko Hirata, Koichi Tsuchiya, Yasuaki Hiraoka, Ippei Obayashi, Koji Ohara, Akitoshi Mizuno, Osami Sakata, Understanding diffraction patterns of glassy, liquid and amorphous materials via persistent homology analyses, Journal of the Ceramic Society of Japan 127 (12) (2019) 853–863.