Computing Persistent Homology with Various Coefficient Fields in a Single Pass

Jean-Daniel Boissonnat, Clément Maria

COMPUTING PERSISTENT HOMOLOGY WITH VARIOUS COEFFICIENT FIELDS IN A SINGLE PASS

JEAN-DANIEL BOISSONNAT AND CLÉMENT MARIA

Abstract. This article introduces an algorithm to compute the persistent homology of a filtered complex with various coefficient fields in a single matrix reduction. The algorithm is output-sensitive in the total number of *distinct* persistent homological features in the diagrams for the different coefficient fields. This computation allows us to infer the prime divisors of the torsion coefficients of the integral homology groups of the topological space at any scale, hence furnishing a more informative description of topology than persistence in a single coefficient field. We provide theoretical complexity analysis as well as detailed experimental results. The code is part of the `Gudhi` software library, and is available at [21].

This article appeared in the Journal of Applied and Computational Topology 2019 [6]. An extended abstract of this article appeared in the proceedings of the European Symposium on Algorithms 2014 [5].

1. Introduction

Persistent homology [12, 24] is an invariant measuring the topological features of the sublevel sets of a function defined on a topological space. Its generality and stability [9] with regard to noise have made it a widely used tool in applied topology. When considering homology with field coefficients—in opposition to integer coefficients—persistent homology admits an algebraic decomposition that can be represented by a *persistence diagram* [24]. The persistence diagram contains rich information about the topology of the studied space and very efficient methods exist to compute it. However, the integral homology groups of a topological space are strictly more informative than the homology groups with field coefficients, in particular because they convey information about *torsion* in homology. Algebraically, torsion is characterized by cyclic subgroups of the integral homology groups, and appears in the range of application of computational topology, such as topological data analysis—where, for example, Klein bottles appear naturally [7, 22]—or the study of random complexes—where a burst of torsion subgroups of large order are found [17].

When homology is computed with field coefficients, these torsion subgroups may either vanish or contribute to the homology, depending on their (unknown) orders. This consequently obfuscate the study of the topology of data and complexes. A simple approach to distinguish between the two cases is to compute persistent homology with different coefficient fields and track the differences in the persistence diagrams.

We build on this idea and describe an efficient algorithm to compute persistent homology with various coefficient fields $\mathbb{Z}/q_1\mathbb{Z}, \cdots, \mathbb{Z}/q_r\mathbb{Z}$ in a single pass of the matrix reduction algorithm. To do so, we introduce a method we call *modular reconstruction* consisting of using the *Chinese Remainder Isomorphism* to encode an element of $\mathbb{Z}/q_1\mathbb{Z} \times \cdots \times \mathbb{Z}/q_r\mathbb{Z}$ with an element of $\mathbb{Z}/(q_1 \cdots q_r)\mathbb{Z}$. This is a simple solution to implement a simple idea. However, it requires the introduction of technical tools for dedicated arithmetic operations, and the solution is tailored for persistent homology computations.

Specifically, we describe algorithms to perform elementary row/column operations in a matrix with $\mathbb{Z}/(q_1 \cdots q_r)\mathbb{Z}$ coefficients, corresponding to simultaneous elementary row/column operations in r distinct matrices with coefficients in the fields $\mathbb{Z}/q_1\mathbb{Z}, \cdots, \mathbb{Z}/q_r\mathbb{Z}$ respectively. The method results in an algorithm with an output-sensitive complexity in the total number of *distinct* pairs in the echelon forms of the matrices with $\mathbb{Z}/q_1\mathbb{Z}, \cdots, \mathbb{Z}/q_r\mathbb{Z}$ coefficients, plus an overhead due to arithmetic operations on big numbers in $\mathbb{Z}/(q_1 \cdots q_r)\mathbb{Z}$. We present the method for computing persistent homology with several coefficient fields using the original persistence algorithm [13, 24], but the methodology and generic tools developed may be applied to other persistent homology algorithms relying on elementary row/column operations, such as the persistent cohomology algorithms of [4, 10, 11]. Finally, we describe how to infer the torsion coefficients of the integral homology using the *Universal Coefficient Theorem for Homology*, and how to integrate this information in a *multi-field persistence diagram*, that could be used in application pipelines.

We discuss applications of the algorithm, and provide experimental analysis that on practical examples of interest, our method is significantly faster than the brute-force approach consisting in reducing separately r matrices with coefficients in $\mathbb{Z}/q_1\mathbb{Z}, \cdots, \mathbb{Z}/q_r\mathbb{Z}$. It is important to note that the method does not pretend to scale to large r, as the arithmetic complexity of operations in $\mathbb{Z}/(q_1 \cdots q_r)\mathbb{Z}$ becomes problematic.

Computing persistent homology with different coefficients has been mentioned in the literature [24] in order to verify if a persisting feature was due to an actual "hole" (or high-dimensional equivalent) or to torsion (and consequently existed only for a certain coefficient field). The issues caused by homological torsion in the study of data using persistent homology is also discussed in [10]. To the best of our knowledge, this is the first work describing an efficient and practical algorithm to compute persistence with various coefficient fields in order to detect and analyse torsion subgroups in persistent homology.

2. Background

For simplicity, we focus in the following on simplicial complexes and their homology. However, the approach and the algorithms do not rely on the simplicial structure, and apply to general complexes.

2.1. Simplicial Homology with General Coefficients.

We refer the reader to [16] for an introduction to homology and to [12] for an introduction to persistent homology.

A *simplicial complex* $\mathbf{K}$ on a set of *vertices* $V = \{1, \cdots, n\}$ is a collection of simplices $\{\sigma\}$, $\sigma \subseteq V$, such that $\tau \subseteq \sigma \in \mathbf{K} \Rightarrow \tau \in \mathbf{K}$. The dimension $d = |\sigma| - 1$ of σ is its number of elements minus 1. For a ring $\mathcal{R}$, the group of d-chains, denoted by $\mathbf{C}_d(\mathbf{K}, \mathcal{R})$, of $\mathbf{K}$ is the group of formal sums of d-simplices with $\mathcal{R}$ coefficients. The *boundary operator* is a linear operator $\partial_d : \mathbf{C}_d(\mathbf{K}, \mathcal{R}) \to \mathbf{C}_{d-1}(\mathbf{K}, \mathcal{R})$ such that $\partial_d \sigma = \partial_d [v_0, \cdots, v_d] = \sum_{i=0}^{d} (-1)^i [v_0, \cdots, \widehat{v_i}, \cdots, v_d]$, where $\widehat{v_i}$ means v_i is deleted from the list. It will be convenient to consider later the endomorphism $\partial_* : \bigoplus_d \mathbf{C}_d(\mathbf{K}, \mathcal{R}) \to \bigoplus_d \mathbf{C}_d(\mathbf{K}, \mathcal{R})$ extended by linearity to the external sum of chain groups. Denote by $\mathbf{Z}_d(\mathbf{K}, \mathcal{R})$ and $\mathbf{B}_{d-1}(\mathbf{K}, \mathcal{R})$ the kernel and the image of ∂_d respectively. Observing $\partial_d \circ \partial_{d+1} = 0$, we define the d^{th} homology group $\mathbf{H}_d(\mathbf{K}, \mathcal{R})$ of $\mathbf{K}$ by the quotient $\mathbf{H}_d(\mathbf{K}, \mathcal{R}) = \mathbf{Z}_d(\mathbf{K}, \mathcal{R}) / \mathbf{B}_d(\mathbf{K}, \mathcal{R})$.

If $\mathcal{R}$ is the *ring of integers* $\mathbb{Z}$, $\mathbf{H}_d(\mathbf{K}, \mathbb{Z})$ is an abelian group and, according to the *fundamental theorem of finitely generated abelian groups* [16], admits a *primary decomposition*:

$$\mathbf{H}_d(\mathbf{K}, \mathbb{Z}) \cong \mathbb{Z}^{\beta_d(\mathbb{Z})} \bigoplus_{q \text{ prime}} \left(\mathbb{Z}/q^{k_1}\mathbb{Z} \oplus \cdots \oplus \mathbb{Z}/q^{k_{t(d,q)}}\mathbb{Z} \right) \tag{2.1}$$

for a uniquely defined integer $\beta_d(\mathbb{Z})$, called the d^{th} *integral Betti number*, and integers $t(d,q) \geq 0$ and $k_i > 0$ for every prime number q. If $t(d,q) > 0$, the integers $q^{k_1}, \cdots, q^{k_{t(d,q)}}$ are called *torsion coefficients*, and they admit q as unique *prime divisor*. Intuitively, in dimension 0, 1 and 2, the integral Betti numbers count the number of connected components, the number of holes and the number of voids respectively. Torsion captures features such as non-orientability in surfaces ; see Section 3.2 and Figure 1 for the example of the Klein bottle. If $\mathcal{R}$ is a *field* $\mathbb{F}$, $\mathbf{H}_d(\mathbf{K}, \mathbb{F})$ is a vector-space and decomposes into

$$\mathbf{H}_d(\mathbf{K}, \mathbb{F}) \cong \mathbb{F}^{\beta_d(\mathbb{F})}$$

where $\beta_d(\mathbb{F})$ is the d^{th} *field Betti number*. The field Betti numbers $(\beta_d(\mathbb{F}))_d$ are entirely determined by the characteristic of $\mathbb{F}$ and the integral homology ; see Section 3.1. Hence, the integral homology is more informative than homology in $\mathbb{F}$.

We suggest in Section 7 the study of the $\mathbb{Z}$-homology of geometric data and random complexes. It is unclear how often integral homology is more informative that field homology in general geometric data, but important cases where torsion is fundamental in the study of data have been observed [7, 22]. The analysis of torsion is however fundamental in the study of random complexes [20].

2.2. **Persistent Homology with Field Coefficients.** A *filtration* of a complex is a function $f : \mathbf{K} \to \mathbb{R}$ satisfying $f(\tau) \leq f(\sigma)$ whenever $\tau \subseteq \sigma$. Ordering the simplices of $\mathbf{K}$ by strictly increasing f-value, we get an increasing sequence of complexes

$$\emptyset = \mathbf{K}_0 \subsetneq \mathbf{K}_1 \subsetneq \cdots \subsetneq \mathbf{K}_{m-1} \subsetneq \mathbf{K}_m = \mathbf{K}$$

where all simplices in $\mathbf{K}_i \setminus \mathbf{K}_{i-1}$ have same filtration value. Without loss of generality, we suppose in the following that all f-values are distinct, and that successive complexes differ by exactly one simplex, i.e., $\mathbf{K}_i = \mathbf{K}_{i-1} \cup \{\sigma_i\}$. The *size* of a filtration is the number of simplices m in the complex $\mathbf{K}$.

A filtration induces a sequence of d-homology groups

$$0 = \mathbf{H}_d(\mathbf{K}_0, \mathcal{R}) \to \mathbf{H}_d(\mathbf{K}_1, \mathcal{R}) \to \cdots \to \mathbf{H}_d(\mathbf{K}_m, \mathcal{R}) = \mathbf{H}_d(\mathbf{K}, \mathcal{R})$$

connected by homomorphisms, induced by the inclusions. In the following, we denote simply by $\mathbf{K}$ the filtration $(\mathbf{K}, f)$. When $\mathcal{R}$ is a field, the latter sequence admits an algebraic decomposition that can be described in terms of a family of intervals $\{(i,j)\}$, called an *indexed persistence diagram*, where a pair (i,j) belongs to $\{1 \dots m\} \times \{1 \dots, m, \infty\}$ and is interpreted as a homology feature that *is born* at index i and *dies* at index j (homology features which never die have death ∞). Note than, for simplicial complexes, an index $i \in \{1 \dots n\}$ belongs to exactly one pair (as birth or death) of the indexed persistence diagram. We assume this property true in the remainder of the article. For a fixed field of coefficients $\mathbb{F}$, computing the persistent homology of a filtration consists of computing the persistence diagram of the induced sequence of $\mathbb{F}$-homology groups.

3. Multi-Field Persistent Homology

We call the algorithmic problem of computing persistent homology for a family of coefficient fields $\mathbb{Z}/q_1\mathbb{Z}, \dots, \mathbb{Z}/q_r\mathbb{Z}$ *multi-field persistent homology*. As explained in the next section, computing multi-field persistence allows us to infer a more informative description of the topology of a space, compared to persistence in a single field.

3.1. **Inference of Torsion.** For a topological space $\mathbb{X}$, the *Universal Coefficient Theorem for Homology* [16] establishes the relationship between the homology groups $\mathbf{H}_d(\mathbb{X},\mathbb{Z})$ with $\mathbb{Z}$ coefficients and the homology groups $\mathbf{H}_d(\mathbb{X},\mathbb{Z}/q\mathbb{Z})$ with coefficients in the field $\mathbb{Z}/q\mathbb{Z}$ (of characteristic q), for q prime. We use the following corollary:

Corollary 3.1 (Universal Coefficient Theorem [16][Corollary 3A.6.(b)])**.** *Denote by $\beta_d(\mathbb{Z})$ and $\beta_d(\mathbb{Z}/q\mathbb{Z})$ the Betti numbers of $\mathbf{H}_d(\mathbb{X},\mathbb{Z})$ and $\mathbf{H}_d(\mathbb{X},\mathbb{Z}/q\mathbb{Z})$ respectively, and $t(j,q)$ the number of $\mathbb{Z}/q^{k_i}\mathbb{Z}$ summands in the primary decomposition of the homology group $\mathbf{H}_j(\mathbb{X},\mathbb{Z})$ as in Equation (2.1), we have:*

$$\beta_d(\mathbb{Z}/q\mathbb{Z}) = \beta_d(\mathbb{Z}) + t(d,q) + t(d-1,q)$$

Suppose $\{q_1,\cdots,q_r\}$ are the first r prime numbers and q_r is a strict upper bound on the prime divisors of the torsion coefficients of $\mathbb{X}$. Consequently, according to Corollary 3.1, $\beta_d(\mathbb{Z}/q_r\mathbb{Z}) = \beta_d(\mathbb{Z})$ for all dimensions d. Moreover, there is no torsion in 0-homology [16], and $t(0,q) = 0$ for all primes q. Given the Betti numbers of $\mathbb{X}$ in all fields $\mathbb{Z}/q_s\mathbb{Z}, 1 \leq s \leq r$, we deduce from Corollary 3.1 the recurrence formula $t(d,q_s) = \beta_d(\mathbb{Z}/q_s\mathbb{Z}) - \beta_d(\mathbb{Z}/q_r\mathbb{Z}) - t(d-1,q_s)$, from which we compute the value of $t(d,q)$ for every dimension d and prime q. For any dimension d, we consequently infer the integral Betti numbers and the number $t(d,q)$ of $\mathbb{Z}/q^{k_i}\mathbb{Z}$ summands in the primary decomposition of $\mathbf{H}_d(\mathbb{X},\mathbb{Z})$.

It is important to notice two limitations of this approach. First, the universal coefficient theorem does not allow us to infer powers k_i from the summands $\mathbb{Z}/q_i^{k_i}\mathbb{Z}$ in the decomposition of the homology groups with $\mathbb{Z}$-coefficient, as in Equation (2.1), by computing homology with field coefficients. Consequently, a summand $\mathbb{Z}/q^{k_i}\mathbb{Z}$ is detected as a summand $\mathbb{Z}/q^*\mathbb{Z}$, for an unknown power of q. Second, determining an upper bound q_r on the prime divisors of the torsion coefficients of a complex is a difficult task in general. However, computing separately persistent homology with $\mathbb{Q}$-coefficients provides the Betti numbers $\beta_d(\mathbb{Q})$ that are equal to $\beta_d(\mathbb{Z})$, and can be used in the formula of Corollary 3.1. This allows us to detect correctly the summands $\mathbb{Z}/q^{k_i}\mathbb{Z}$ for all $q \leq q_r$, even when q_r is not an upper bound on the prime divisors of the torsion coefficients.

We discuss the question of upper bounds of prime divisors of torsion coefficients in the experimental Section 7 for different types of data sets.

3.2. **Representation of the Multi-Field Persistence Diagram.** Persistence diagrams are represented by sets of points in the plane, where to every persistent pair (i,j) of the diagram corresponds a point with coordinates (i,j) in the plane ; see Figure 1. We generalize this representation to multi-field persistence diagram by plotting the *superimposition* of the persistence diagrams in each coefficient field, and by infering an expression of the integral homology group in each cell of the diagram.

We refer to Figure 1 for an example. It pictures the multi-field persistence diagram of the 1-homology of a filtration $\mathbf{K}$ approximating a Klein bottle (for field coefficients $\mathbb{Z}/2\mathbb{Z}$ and $\mathbb{Z}/3\mathbb{Z}$). The integral 1-homology of the Klein bottle is $\mathbf{H}_1(\mathbf{K},\mathbb{Z}) = \mathbb{Z} \oplus \mathbb{Z}/2\mathbb{Z}$, and $\mathbf{H}_1(\mathbf{K},\mathbb{Z}/2\mathbb{Z}) = (\mathbb{Z}/2\mathbb{Z})^2$ and $\mathbf{H}_1(\mathbf{K},\mathbb{Z}/3\mathbb{Z}) = \mathbb{Z}/3\mathbb{Z}$, and the integral homology appears clearly in the multi-field persistence diagram.

Notion of distances, such as *bottleneck distance* and *Wasserstein distance*, and stability [9], extend naturally to this presentation, by defining the distance between two multi-field persistence diagram for coefficients $\mathbb{Z}/q_1,\ldots,\mathbb{Z}/q_r\mathbb{Z}$ as the maximal distance between the corresponding standard persistence diagrams over all coefficient fields $\mathbb{Z}/q_s\mathbb{Z}$, $1 \leq s \leq r$.

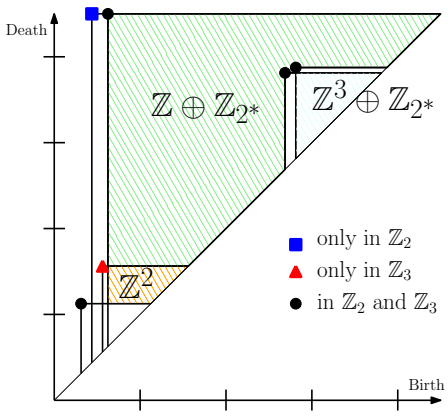

FIGURE 1. Multi-field persistence diagram of the most persistent features of $\mathbf{H}_1$ for a Rips complex reconstructing a Klein bottle. The "$*$" in $\mathbb{Z}/2^*\mathbb{Z}$ indicates that the persisting homology admits a torsion summand $\mathbb{Z}/2^k\mathbb{Z}$ for some unknown $k \geq 1$. The 1-homology $\mathbb{Z} \oplus \mathbb{Z}/2\mathbb{Z}$ of the underlying Klein bottle appears clearly as persisting.

4. Algorithm for Multi-Field Persistent Homology

In this section we design an efficient algorithm to compute multi-field persistent homology. For a filtered complex $\mathbf{K}$ of size m, denote by $P_{\mathbb{F}}$ the number of pairs of indices $(i, j) \in \{1, \dots, m\} \times \{1, \dots, m, \infty\}$ forming the index persistence diagram with coefficients in a field $\mathbb{F}$. For a set of coefficient fields $\mathbb{Z}/q_1\mathbb{Z}, \dots, \mathbb{Z}/q_r\mathbb{Z}$, denote by P_r the number of *distinct* pairs of indices appearing the persistence diagram of $\mathbf{K}$ for every coefficient field $\mathbb{Z}/q_s\mathbb{Z}$.

We design an algorithm, called *modular reconstruction* algorithm, of complexity

$$O\left(\left[r \times (P_r - P_{\mathbb{F}}) + P_r^3\right] \times \mathsf{A}_r\right)$$

where A_r is a bound on the time complexity of arithmetic operations on large integers in $\mathbb{Z}/(q_1 \cdots q_r)\mathbb{Z}$ (see Section 5.1), and the P_r^3 stands for the standard cubic complexity of computing persistent homology. Note that the additional component $r \times (P_r - P_{\mathbb{F}})$ depends on the number of distinct bars in the persistence diagram when changing coefficient fields which, in light of Section 3.1, is directly related to torsion. In that sense, the algorithm is output-sensitive.

For clarity, we focus in this section on the persistent homology algorithm as presented in [12][Chapter VII], which consists of a reduction to column echelon form (defined later) of a matrix. All practical persistent homology algorithms rely on atomic matrix column operations. Our approach to multi-field persistence is described in terms of these column operations, and can consequently be adapted to other practical persistent homology implementations. In the following, $\mathbb{Z}/n\mathbb{Z}$ denotes the ring $(\mathbb{Z}/n\mathbb{Z}, +, \times)$ for any integer $n \geq 1$ and $(\mathbb{Z}/n\mathbb{Z})^\times$ the subset of invertible elements for $\times$. If it exists, we denote the inverse of $x \in \mathbb{Z}/n\mathbb{Z}$ by x^{-1}.

4.1. Persistent Homology Algorithm.

In this section we recall the standard matrix algorithm to compute persistent homology with coefficient in a field [12][Chapter VII].

For an $m \times m$ matrix $\mathbf{M}$, denote by col_j the j^{th} column of $\mathbf{M}$, $1 \leq j \leq m$, and denote by $\mathsf{col}_j[k]$ the k^{th} entry of the column. Let $\text{low}(j)$ denote the row index of the lowest non-zero entry of col_j. If the column j is entirely zero, $\text{low}(j)$ is undefined. We say that $\mathbf{M}$ is in *reduced*

Data: Boundary matrix $\mathbf{R} \leftarrow \mathbf{M}_\partial$, persistence diagram $\mathcal{P}_\mathbb{F} \leftarrow \emptyset$
Output: Persistence diagram $\mathcal{P}_\mathbb{F} = \{(i,j)\}$

```
for j = 1, ··· , m do
    while there exists j' < j with low(j') = low(j) do
        k ← low(j);
        col_j ← col_j − (col_j[k] × col_{j'}[k]^{-1}) · col_{j'};
    end
    if col_j ≠ 0 then P_F ← P_F ∪ {(low(j), j)};
end
```

Algorithm 1: Persistent homology algorithm.

column echelon form if, for any two non-zero columns col_j and $\mathsf{col}_{j'}$, $j \neq j'$, the columns satisfy $\mathrm{low}(j) \neq \mathrm{low}(j')$.

Let $\mathbf{K} = (\sigma_i)_{i=1\cdots m}$ be a filtered complex. For a fixed coefficient field $\mathbb{F}$, its boundary matrix $\mathbf{M}_\partial$ is the $m \times m$ matrix, with $\mathbb{F}$ entries, of the endomorphism ∂_* in the basis $\{\sigma_1, \cdots, \sigma_m\}$ of $\bigoplus_d \mathbf{C}_d(\mathbf{K}, \mathbb{F})$. The basis is ordered according to the filtration. It is a matrix with $\{-1, 0, 1\}$ entries, where 0 and 1 denote the identity $0_\mathbb{F}$ for $+$ and the identity $1_\mathbb{F}$ for $\times$ in $\mathbb{F}$ respectively, and -1 is the inverse of $1_\mathbb{F}$ in $\mathbb{F}$. The persistent homology algorithm consists of a left-to-right reduction to column echelon form of $\mathbf{M}_\partial$, presented in Algorithm 1. We denote by $\mathbf{R}$ the matrix we reduce, with columns col_j, and which is initially equal to $\mathbf{M}_\partial$. The algorithm returns the *(indexed) persistence diagram*, which is the set of pairs $\{(\mathrm{low}(j), j)\}$ in the reduced column echelon form of the matrix. Note that the "infinite intervals" of the diagram can be infered by reading the null columns of the reduced matrix, and for simplicity we do not include this computation in the pseudo-code.

The reduced form of the matrix is not unique, but the pairs (i, j) such that $i = \mathrm{low}(j)$ in the column echelon form are [12]. The algorithm requires $O(m^3)$ arithmetic operations in $\mathbb{F}$.

4.2. **Modular Reconstruction for Elementary Matrix Operations.** Denote by $[r]$ the set of integers $\{1, \cdots, r\}$. For a family of r distinct prime numbers $\{q_1, \cdots, q_r\}$, and a subset of indices $S \subseteq [r]$, Q_S refers to the product $\prod_{s \in S} q_s$, and we write simply $Q := Q_{[r]}$. For any integer $z \in \mathbb{Z}$ and positive integer $n > 0$, $z \bmod n$ refers to the equivalence class of z in $\mathbb{Z}/n\mathbb{Z}$. For simplicity, any element $x \in \mathbb{Z}/n\mathbb{Z}$ is identified with the smallest positive integer belonging to the class x in $\mathbb{Z}/n\mathbb{Z}$. We also denote this integer by $x \in \mathbb{Z}$, $0 \leq x < m$. Consequently, for $x \in \mathbb{Z}/n\mathbb{Z}$, $x \bmod n'$ refers to the class of $\mathbb{Z}/n'\mathbb{Z}$ to which belongs the integer $x \in \mathbb{Z}$, and $(\bmod n')$ can be seen as a ring homomorphism $\mathbb{Z}/n\mathbb{Z} \to \mathbb{Z}/n'\mathbb{Z}$.

We present a particular case of the *Chinese Remainder Theorem*, and recall a simple constructive proof.

Theorem 4.1 (Chinese Remainder Theorem [15])**.** *For a family $\{q_1, \cdots, q_r\}$ of r distinct prime numbers, there exists a ring isomorphism*

$$\psi : \mathbb{Z}/q_1\mathbb{Z} \times \cdots \times \mathbb{Z}/q_r\mathbb{Z} \to \mathbb{Z}/(q_1 \cdots q_r)\mathbb{Z}$$

The isomorphisms ψ and ψ^{-1} can be computed in $O(r)$ arithmetic operations in $\mathbb{Z}/(q_1 \cdots q_r)\mathbb{Z}$.

Proof. Euler's theorem states that for two coprime integers a and n, $a^{\varphi(n)} \bmod n = 1$, where φ is Euler's totient function, which is equal to $\varphi(q) = q - 1$ on a prime integer q. For $1 \leq s \leq r$, define $\nu_s = (Q_{[r]\setminus\{s\}})^{q_s - 1} \bmod Q$.

For all $1 \leq s \leq r$, there consequently exist integers ν_s such that $\nu_s \bmod q_t = 1$ if $s = t$ and 0 otherwise. The following expressions of ψ and ψ^{-1} realize the isomorphism of the theorem:

$$\begin{array}{rccc} & \mathbb{Z}/q_1\mathbb{Z} \times \cdots \times \mathbb{Z}/q_r\mathbb{Z} & \leftrightarrow & \mathbb{Z}/(q_1 \cdots q_r)\mathbb{Z} \\ \psi: & (u_1, \cdots, u_r) & \mapsto & (u_1\nu_1 + \cdots + u_r\nu_r) \bmod Q \\ \psi^{-1}: & (x \bmod q_1, \cdots, x \bmod q_r) & \leftarrow & x \end{array}$$

□

In the following, we consider the isomorphism of the former proof when referring to the isomorphism given by the Chinese Remainder Theorem. We denote by ψ_S the function $\psi_S : \prod_{s\in S} \mathbb{Z}/q_s\mathbb{Z} \to \mathbb{Z}/Q_S\mathbb{Z}$ realizing the isomorphism of the Chinese Remainder Theorem for the subset $\{q_s\}_{s\in S}$, $S \subset [r]$, of prime integers, and we write simply ψ for $\psi_{[r]}$. For a family of elements $u_s \in \mathbb{Z}/q_s\mathbb{Z}, s \in S$, we denote the corresponding $|S|$-tuple $(u_s)_{s\in S} \in \prod_{s\in S} \mathbb{Z}/q_s\mathbb{Z}$. Finally, we recall *Bezout's lemma* [15].

Lemma 4.2 (Bezout)**.** *For two integers a and b, not both 0, there exist integers v and w such that $va + wb = \gcd(a, b)$, the greatest common divisor of a and b, with $|v| < |b/\gcd(a, b)|$ and $|w| < |a/\gcd(a, b)|$.*

The Bezout's coefficients (v, w) can be computed with the extended Euclidean algorithm [15].

Elementary Column Operations. We are given a family of distinct prime numbers $\{q_1, \cdots, q_r\}$, and their product $Q = q_1 \cdots q_r$. Let $\mathbf{M}_Q$ be a matrix with entries in the ring $\mathbb{Z}/Q\mathbb{Z}$. Denoting by $\psi^{-1} : \mathbb{Z}/Q\mathbb{Z} \to \mathbb{Z}/q_1\mathbb{Z} \times \cdots \times \mathbb{Z}/q_r\mathbb{Z}$ the isomorphism of the Chinese Remainder Theorem, and $\pi_s : \mathbb{Z}/q_1\mathbb{Z} \times \cdots \times \mathbb{Z}/q_r\mathbb{Z} \to \mathbb{Z}/q_s\mathbb{Z}$ the projection on the s^{th} coordinate, we call *projection of $\mathbf{M}_Q$ onto $\mathbb{Z}/q_s\mathbb{Z}$*, denoted $\mathbf{M}_Q(\mathbb{Z}/q_s\mathbb{Z})$, the matrix with entries in $\mathbb{Z}/q_s\mathbb{Z}$, obtained by applying $\pi_s \circ \psi^{-1}$ pointwise to each entry of $\mathbf{M}_Q$.

Conversely, given a number r of $(m \times m)$-matrices $\mathbf{M}_{q_1}, \cdots, \mathbf{M}_{q_r}$ with coefficients in $\mathbb{Z}/q_1\mathbb{Z}, \cdots, \mathbb{Z}/q_r\mathbb{Z}$ respectively, there exists a unique matrix $\mathbf{M}_Q$ with $\mathbb{Z}/Q\mathbb{Z}$ entries such that, for every index s in a prime number q_s, $\mathbf{M}_Q$ satisfies $\mathbf{M}_Q(\mathbb{Z}/q_s\mathbb{Z}) = \mathbf{M}_{q_s}$. This is simply a matrix version of the Chinese Remainder Theorem.

Elementary column operations on a matrix $\mathbf{M}$ with entries in a ring $\mathcal{R}$ are of three kinds:

(i) exchange col_k and col_ℓ,
(ii) multiply col_k by $-1 \in \mathcal{R}$,
(iii) replace col_k by $(\mathsf{col}_k + \alpha \times \mathsf{col}_\ell)$, for a $\alpha \in \mathcal{R}$.

For an elementary column operation $(*)$ (i.e., an operation of type (i), (ii) or (iii) applied to some columns of the matrix), we denote by $(*) \circ \mathbf{M}$ the result of applying $(*)$ to $\mathbf{M}$. Any reduction algorithm relies on these three operations. A key feature of the persistent homology reduction is the ability to inverse elements when reducing a matrix with *field* coefficients (applying column operation (iii) in line 4 of the Algorithm 1).

In the following we introduce algorithms to run elementary column operations simultaneously on matrices $\mathbf{M}_{q_1}, \cdots, \mathbf{M}_{q_r}$ with coefficients in the fields $\mathbb{Z}/q_1\mathbb{Z}, \cdots, \mathbb{Z}/q_r\mathbb{Z}$ respectively, by performing *partial column operations* on matrix $\mathbf{M}_Q$ with coefficient in the ring $\mathbb{Z}/Q\mathbb{Z}$, such that the $\mathbf{M}_{q_j}$ and $\mathbf{M}_Q$ are related by the Chinese Remainder Theorem as above.

Specifically, for an elementary column operation $(*)$, on column k, and ℓ, and with scalar $\alpha \in \mathbb{Z}/Q\mathbb{Z}$, and a subset of indices $S \subseteq [r]$, we call *partial column operation*, denoted by $(*)_S$, on $\mathbf{M}_Q$ the operation transforming $\mathbf{M}_Q$ into $\mathbf{M}'_Q = (*)_S \circ \mathbf{M}_Q$ satisfying:

$$\mathbf{M}'_Q \text{ satisfies } \begin{cases} \mathbf{M}'_Q(\mathbb{Z}/q_s\mathbb{Z}) = (*) \circ \mathbf{M}_{q_s} & \text{if } s \in S, \\ \mathbf{M}'_Q(\mathbb{Z}/q_s\mathbb{Z}) = \mathbf{M}_{q_s} & \text{otherwise.} \end{cases}$$

where $(*)$ on $\mathbf{M}_{q_s}$ is on column k, and ℓ, and with scalar $\pi_s \circ \psi^{-1}(\alpha)$ in $\mathbb{Z}/q_s\mathbb{Z}$. The correspondence $\psi : \mathbb{Z}/q_1\mathbb{Z} \times \cdots \times \mathbb{Z}/q_r\mathbb{Z} \to \mathbb{Z}/Q\mathbb{Z}$ is a ring homomorphism, i.e., it satisfies:

$$\psi(u_1, \cdots, u_r) + \psi(v_1, \cdots, v_r) \times \psi(w_1, \cdots, w_r) = \psi(u_1 + v_1 \times w_1, \cdots, u_r + v_r \times w_r)$$

Consequently, we can compute additions and multiplications componentwise in $\mathbb{Z}/q_1\mathbb{Z} \times \cdots \times \mathbb{Z}/q_r\mathbb{Z}$ using addition and multiplication in $\mathbb{Z}/Q\mathbb{Z}$.

In order to compute partial column operations, we first introduce the set of *partial identities*, which are coefficients that allow us to proceed to the partial column operations of type (i) and (ii). Secondly, as the rings $\mathbb{Z}/q_s\mathbb{Z}$ are fields, we need to compute the multiplicative inverse of an element, that is used as multiplicative coefficient α in elementary column operation (iii). As $\mathbb{Z}/Q\mathbb{Z}$ is not a field, inversion is not possible, and we introduce the concept of *partial inverse* to overcome this difficulty. In the following, the term "arithmetic operation" refers to any operation $+$, $-$, $\times$, $\gcd(\cdot,\cdot)$, $(\cdot \mod Q_S)$, and Extended Euclidean algorithm on integer smaller than Q. Note they do not have constant time complexity for large Q. We discuss arithmetic complexity in Section 5.1.

Partial Identity and Partial Inverse. Given a subset of indices $S \subseteq [r]$, we define the *partial identities w.r.t.* S, denoted by L_S and equal to

$$L_S = \psi(\delta_{1,S}, \cdots, \delta_{r,S}), \text{ where } \delta_{s,S} \in \mathbb{Z}/q_s\mathbb{Z} \text{ is equal to } \begin{cases} 1 & \text{if } s \in S, \\ 0 & \text{otherwise.} \end{cases}$$

For any $S \subseteq [r]$, the partial identity L_S can be constructed in $O(r)$ arithmetic operations in $\mathbb{Z}/Q\mathbb{Z}$ by evaluating ψ on $(\delta_{1,S}, \cdots, \delta_{r,S})$. However, it is important to notice that if $S = [r]$, $L_{[r]} = \psi(1, \cdots, 1) = 1$, because ψ is a ring isomorphism, and L_r is computed in time $O(1)$.

Knowing the partial identities, we can implement the partial column operations (i) and (ii) for a set of indices S. Specifically,

(i) replace column col_k by $(\mathsf{col}_k \times L_{[r]\setminus S} + \mathsf{col}_\ell \times L_S)$, and
replace column col_ℓ by $(\mathsf{col}_\ell \times L_{[r]\setminus S} + \mathsf{col}_k \times L_S)$,
(ii) multiply column col_k by $L_{[r]} - 2 \times L_S$.

As mentioned earlier, we need a notion of "partial multiplicative inverse" in $\mathbb{Z}/Q\mathbb{Z}$ in order to pick the appropriate scalar α when defining a partial version of elementary column operation (iii). We define the *partial inverse* of an element of the ring $\mathbb{Z}/Q\mathbb{Z}$ to be:

Definition 4.3 (Partial Inverse)**.** *Given a set $S \subseteq [r]$ of indices, and an element $x = \psi(u_1, \cdots, u_r)$ in $\mathbb{Z}/Q\mathbb{Z}$, the* partial inverse of x with regard to S *is the element $\overline{x}^S \in \mathbb{Z}/Q\mathbb{Z}$ equal to*

$$\overline{x}^S = \psi(\overline{u_1}^S, \cdots, \overline{u_r}^S), \quad \textit{with} \quad \overline{u_s}^S = \begin{cases} u_s^{-1} & \textit{if } s \in S \textit{ and } u_s \in \mathbb{Z}/q_s\mathbb{Z}^\times, \\ 0 & \textit{otherwise.} \end{cases}$$

We prove elementary arithmetic and computational properties of partial inverses.

Proposition 4.4 (Partial Inverse Construction)**.** *For a set $S \subseteq [r]$ of indices and an element $x = \psi(u_1, \cdots, u_r)$ in $\mathbb{Z}/Q\mathbb{Z}$, the following is true:*

(1) $\gcd(x, Q_S) = Q_R$ for some $R \subseteq S$. Additionally, for all $s \in S$, u_s is invertible in $\mathbb{Z}/q_s\mathbb{Z}$ iff $s \notin R$. We denote by T the set $T := S \setminus R$.
(2) The Bezout's identity for x and Q_T gives $v \cdot x + w \cdot Q_T = 1$, where v satisfies $v \bmod Q_T = \psi_T((u_s^{-1})_{s \in T})$

(3) Finally,

$$\overline{x}^S = \left[\psi_T((u_s^{-1})_{s\in T}) \times L_T \bmod Q\right] \in \mathbb{Z}/Q\mathbb{Z},$$

where L_T is the partial identity with regard to T.

Proof. (1): The gcd of x and Q_S divides Q_S so $\gcd(x, Q_S) = Q_R$ for some $R \subseteq S$, and for every index $s \in S$, q_s divides x iff $s \in R$. Denote $T := S \setminus R$. According to the Chinese Remainder Theorem, for any $s \in T$, $u_s = x \bmod q_s \neq 0$ because q_s does not divide x. Because $\mathbb{Z}/q_s\mathbb{Z}$ is a field, its unique non invertible element is 0 and consequently u_s is invertible. Conversely, because q_t divides x for $t \in R$, $x \bmod q_t = u_t = 0$ is non invertible.

(2): First note that $x \bmod Q_T = \psi_T((u_t)_{t\in T}) \in \mathbb{Z}/Q_T\mathbb{Z}$. Indeed, because q_t divides Q_T for all $t \in T$, we have

$$(x \bmod Q_T) \bmod q_t = x \bmod q_t = u_t$$

By definition of T, $\gcd(x, Q_T) = 1$ and so the Bezout's lemma gives

$$v \cdot x + w \cdot Q_T = 1$$

Applying $(\cdot \bmod Q_T)$ to both sides of the equality gives

$$(v \bmod Q_T)\psi_T((u_t)_{t\in T}) = 1, \text{and consequently } ((v \bmod Q_T) \bmod q_t)u_t = 1$$

for every q_t such that $t \in T$. The result follows.

(3): Let L_T be the partial identity with regard to T. We form the product

$$\widetilde{x} = \left[\psi_T((u_t^{-1})_{t\in T}) \times L_T \bmod Q\right]$$

and evaluate it modulo q_s. For any index $s \in [r]$,

$$\widetilde{x} \bmod q_s = \left[(\psi_T((u_t^{-1})_{t\in T}) \bmod q_s) \times (L_T \bmod q_s)\right] \bmod q_s$$

If $s \notin T$, then

$$L_T \bmod q_s = 0, \quad \text{and} \quad \widetilde{x} \bmod q_s = 0$$

If $s \in T$, then

$$L_T \bmod q_s = 1, \quad \text{and} \quad \psi_T((u_t^{-1})_{t\in T}) \bmod q_s = u_s^{-1}$$

and consequently $\widetilde{x} \bmod q_s = u_s^{-1}$. Thus, $\widetilde{x}$ satisfies the definition of $\overline{x}^S$, the partial inverse of x with regard to S. □

We directly deduce an algorithm to compute the partial inverse of x w.r.t S if Q_S is given: compute $Q_R = \gcd(x, Q_S)$ and $Q_T = Q_S/Q_R$, then v using the extended Euclidean algorithm and finally $\overline{x}^S = (v \bmod Q_T) \times L_T \bmod Q$. Computing the partial identity L_T requires $O(r)$ arithmetic operations in $\mathbb{Z}/Q\mathbb{Z}$, but is constant if $T = [r]$, which happens iff $S = [r]$ and x is invertible in $\mathbb{Z}/Q\mathbb{Z}$. Consequently, computing $\overline{x}^S$ requires $O(r)$ arithmetic operations in general, but only $O(1)$ arithmetic operations in the latter case.

4.3. **Modular Reconstruction for Multi-Field Persistent Homology.** Let $\mathbf{K}$ be a filtered complex of size m. Define $\mathbf{M}_\partial(\mathbb{Z}/q_s\mathbb{Z})$ to be the $(m \times m)$ boundary matrix of $\mathbf{K}$ with $\mathbb{Z}/q_s\mathbb{Z}$ coefficients. Define $\mathbf{M}$ to be the $(m \times m)$ matrix with $\mathbb{Z}/Q\mathbb{Z}$ coefficients such that the projection of $\mathbf{M}$ onto $\mathbb{Z}/q_s\mathbb{Z}$ is equal to $\mathbf{M}_\partial(\mathbb{Z}/q_s\mathbb{Z})$, for all $s \in [r]$. Note that the matrices $\mathbf{M}$ and $\mathbf{M}_\partial(\mathbb{Z}/q_s\mathbb{Z})$, for any s, are "identical" matrices in the sense that they contain 0, 1 and -1 coefficients at the same positions, where 0, 1 and -1 refer respectively to elements of $\mathbb{Z}/Q\mathbb{Z}$ and $\mathbb{Z}/q_s\mathbb{Z}$.

We reduce a matrix $\mathbf{R}$ which is initially equal to $\mathbf{M}$. Denote by col_j the j^{th} column of $\mathbf{R}$. Define the *extended* low *function* $\text{low}(j, Q_S)$ to be the index of the lowest element of col_j such

Data: Matrix $\mathbf{R} = \mathbf{M}$, diagram $\mathcal{P}_r \leftarrow \emptyset$
Output: Multi-field persistence diagram $\mathcal{P}_r = \{(i, j, Q_S)\}$

1 **for** $j = 1, \cdots, m$ **do**
2 $\quad Q_S \leftarrow Q_{[r]}$;
3 $\quad$ **while** $\mathrm{low}(j, Q_S)$ *is defined* **do**
4 $\quad\quad k \leftarrow \mathrm{low}(j, Q_S)$; $\quad Q_T \leftarrow Q_S / \gcd(\mathsf{col}_j[k], Q_S)$;
5 $\quad\quad$ **while** *there exists* $j' < j$ *with* $(i, Q_{T'}) \in \mathcal{L}(j')$
6 $\quad\quad\quad$ *satisfying* $[i = \mathrm{low}(j, Q_S)$ *and* $\gcd(Q_{T'}, Q_T) > 1]$ **do**
7 $\quad\quad\quad Q_T \leftarrow Q_T / \gcd(Q_{T'}, Q_T)$;
8 $\quad\quad\quad \mathsf{col}_j \leftarrow \mathsf{col}_j - \left(\mathsf{col}_j[k] \cdot \overline{\mathsf{col}_{j'}[k]}^{T}\right) \times \mathsf{col}_{j'}$;
9 $\quad\quad$ **end**
10 $\quad\quad$ **if** $Q_T \neq 1$ **then** $\mathcal{P}_r \leftarrow \mathcal{P}_r \cup \{(k, j, Q_T)\}$; $\quad Q_S \leftarrow Q_S / Q_T$;
11 $\quad\quad$;
12 $\quad$ **end**
13 **end**

Algorithm 2: Simultaneous persistent homology algorithm for $\mathbb{Z}/q_1\mathbb{Z}, \dots \mathbb{Z}/q_r\mathbb{Z}$

that $\mathsf{col}_j[\mathrm{low}(j, Q_S)] \mod Q_S \neq 0$. In particular, $\mathrm{low}(j, q_s)$ is equal to the index of the lowest non-zero element of column j in the projection $\mathbf{R}(\mathbb{Z}/q_s\mathbb{Z})$, and $\mathrm{low}(j, Q_S)$ is equal to

$$\mathrm{low}(j, Q_S) = \max_{s \in S} \mathrm{low}(j, q_s)$$

After iteration j, we say that the columns $\mathsf{col}_1, \cdots, \mathsf{col}_j$ are *reduced*. We maintain, for every reduced column col_j, the collection of "lowest indices" i as a set $\mathcal{L}(j) = \{(i, Q_S)\}$ satisfying three conditions ensuring that low values for all indices $s \in [r]$ are represented, without redundancy. Specifically, the set $\mathcal{L}(j)$ satisfies:

- For every $(i, Q_S) \in \mathcal{L}(j)$, $i = \mathrm{low}(j)$ in matrix $\mathbb{R}(\mathbb{Z}/q_s\mathbb{Z})$ for every $s \in S$.
- Every two distinct pairs $(i, Q_S), (i', Q_{S'}) \in \mathcal{L}(j)$ satisfy both $i \neq i'$ and $S \cap S' = \emptyset$.
- The union $\cup_{(i,Q_S) \in \mathcal{L}(j)} S = [r]$.

The algorithm is presented in Algorithm 2. It returns the set of triplets $\mathcal{P}_r = \{(i, j, Q_S)\}$ such that $i = \mathrm{low}(j)$ in the column echelon form of the matrix $\mathbf{M}_\partial(\mathbb{Z}/q_s\mathbb{Z})$ iff $s \in S$, or, equivalently, $(i, Q_S) \in \mathcal{L}(j)$ once col_j has been reduced. This is a compact encoding of the *multi-field persistence diagram*. Note that it contains exactly P_r elements.

The $\{\mathcal{L}(j)\}_j$ form an index table that we maintain implicitly. At iteration j of the `for` loop, we use Q_S for the product of all prime numbers $\prod_{s \in S} q_s$ for which the column j in $\mathbf{R}(\mathbb{Z}/q_s\mathbb{Z})$ has not yet been reduced.

Analysis. We give details on the line-by-line computation of Algorithm 2 in terms of operations induced in the matrices $\mathbf{R}(\mathbb{Z}/q_s\mathbb{Z})$ for $s \in [r]$. A set of indices $S \subset [r]$ is maintained by storing the product Q_S, and set operations, such as set difference and set intersection, are implemented using respectively arithmetic division and greatest common divisor. Specifically, for $T \subset S \subset [r]$, and $T' \subset [r]$,

$$Q_S / Q_T = Q_{S \setminus T} \quad \text{and} \quad \gcd(Q_T, Q_{T'}) = Q_{T \cap T'}$$

The set S in the **while** loop line 3 contains exactly the set of indices $s \in S$ such that the column col_j of matrix $\mathbf{R}(\mathbb{Z}/q_s\mathbb{Z})$ is not yet reduced. In line 4, k is the lowest row index of a col_j in any of the matrices $\mathbf{R}(\mathbb{Z}/q_t\mathbb{Z})$ such that $t \in S$ (i.e., a matrix in which col_j is still

unreduced). The matrices $\mathbf{R}(\mathbb{Z}/q_t\mathbb{Z})$ where $\text{low}(j)$ is exactly k are the ones for which $t \in T$ (line 4). This property of T is maintained over all of the **while** loop line 3.

The set T' defined on line 5 contains some indices t such that col_j and $\mathsf{col}_{j'}$ have same lower index i in $\mathbf{R}(\mathbb{Z}/q_t\mathbb{Z})$. By definition of the partial inverse and the sets T and T', the column operation line 8 modifies only the matrix $\mathbf{R}(\mathbb{Z}/q_t\mathbb{Z})$ for $t \in T \cap T'$ and reduces strictly their $\text{low}(j)$ values. In line 7, the set T is updated to contain exactly the indices t such that $\text{low}(j) = k$ in $\mathbf{R}(\mathbb{Z}/q_t\mathbb{Z})$.

At line 10, all columns col_j in $\mathbf{R}(\mathbb{Z}/q_t\mathbb{Z}), t \in T$, are reduced and non-zero, we update the multi-field persistence diagram and maintain the property that S contains exactly the indices s for which col_j is still unreduced in $\mathbf{R}(\mathbb{Z}/q_s\mathbb{Z})$.

Correctness. First, note that all operations processed on $\mathbf{R}$ correspond to left-to-right elementary column operations in the matrices $\mathbf{R}(\mathbb{Z}/q_s\mathbb{Z})$ for all $s \in [r]$. One iteration of the `while` loop in line 3 either strictly reduces Q_S by dividing it by Q_T (when $T \neq \emptyset$ in line 10) or sets $(\mathsf{col}_j[k] \mod Q_S)$ to zero thus reducing strictly $\text{low}(j, Q_S)$ (when $T = \emptyset$ and $Q_T = 1$). Consequently, the algorithm terminates.

We prove recursively, on the number of columns, that each of the matrices $\mathbf{R}(\mathbb{Z}/q_s\mathbb{Z})$ gets reduced to column echelon form. We fix an arbitrary field $\mathbb{Z}/q_s\mathbb{Z}$: suppose that the $j-1$ first columns of $\mathbf{R}(\mathbb{Z}/q_s\mathbb{Z})$ have been reduced at the end of iteration $j-1$ of the `for` loop in line 1. We prove that at the end of the j^{th} iteration of the `for` loop in line 1, the j first columns of the matrix $\mathbf{R}(\mathbb{Z}/q_s\mathbb{Z})$ are reduced. Consider two cases.

1. First suppose that there is a triplet (i, j, Q_T) in the multi-field persistence diagram $\mathcal{P}_r$, for some $i < j$ and Q_T satisfying q_s divides Q_T. This implies that the algorithm exits the `while` loop line 5 with q_s dividing Q_T, and Q_T dividing Q_S (because by definition of Q_T, in line 4, Q_T divides Q_S) and there is no $j' < j$ such that $[\text{low}(j', Q_{T'}) = \text{low}(j, Q_S)$ and $[\gcd(Q_{T'}, Q_T) > 1]$. This in particular implies that there is no $j' < j$ such that $\text{low}(j', q_s) = \text{low}(j, q_s)$ and column j is reduced in $\mathbf{R}(\mathbb{Z}/q_s\mathbb{Z})$.

2. Secondly, suppose that there is no such pair (i, j, Q_T) in $\mathcal{P}_r$, with q_s dividing Q_T. Consequently, during all the computation of the `while` loop in line 3, q_s divides Q_S. When exiting this `while` loop, $\text{low}(j, Q_S)$ is undefined, implying in particular that $\text{low}(j, q_s)$ is undefined and column j of $\mathbf{R}(\mathbb{Z}/q_s\mathbb{Z})$ is zero, and hence reduced.

Reconstruction of Cycles and Pairs. Denote by $\mathbf{R}(\mathbb{Z}/q_s\mathbb{Z})$ the matrix maintained at iteration i of the standard persistent homology Algorithm 1 with coefficient in the field $\mathbb{Z}/q_s\mathbb{Z}$. Note that, at iteration i of the modular reconstruction Algorithm 2, we maintain a matrix $\mathbf{R}$ that is a compact representation of all matrices $\mathbf{R}(\mathbb{Z}/q_s\mathbb{Z})$, for $s = 1 \dots r$. Indeed, applying $(\cdot \mod q_s)$ to all coefficients of $\mathbf{R}$ leads to a matrix $\mathbf{R}(C)$. Consequently, we can reconstruct the cycles and the persistent pairs for standard persistent homology with $\mathbb{Z}/q_s\mathbb{Z}$ coefficients, for any $s = 1 \dots r$, with the modular reconstruction algorithm.

5. Output-Sensitive Complexity Analysis

We start by describing a complexity model for the arithmetic operation on large integers.

5.1. Arithmetic Complexity Model for Large Integers.

During the reduction algorithm we perform arithmetic operations on big integers, for which we describe a complexity model [15]. Suppose that on our architecture, a memory word is encoded on $\mathtt{w}$ bits (on modern architectures, $\mathtt{w}$ is usually 64). Computer chips contain Arithmetic Logic Units that allow arithmetic operations on a 1-memory word integer in $O(1)$ machine cycles. Let the *length* of an integer n be defined by: $\lambda(n) = \lfloor \log_2 n/\mathtt{w} \rfloor + 1$, i.e., by the number of memory

words necessary to encode n. We express the arithmetic complexity as a function of the length. For any positive integer n of length $\lambda(n) = \mathsf{B}$, operations in $\mathbb{Z}/n\mathbb{Z}$ cost $\mathsf{A}_+(n) = O(\mathsf{B})$ for additions, $\mathsf{A}_\times(n) = O(M(\mathsf{B}))$ for multiplications, and $\mathsf{A}_\div(n) = O(M(\mathsf{B})\log\mathsf{B})$ for the (extended) Euclidean algorithm, inversions and divisions, where $M(\mathsf{B})$ is a monotonic upper bound on the number of word operations necessary to multiply two integers of length B [15]. The best known upper bound [14] is $M(\mathsf{B}) = O(\mathsf{B}\log\mathsf{B}\, 2^{O(\log^*\mathsf{B})})$, where $\log^*\mathsf{B}$ is the iterated logarithm of B.

In the case of multi-field persistent homology, we are interested in the value of λ for an element in $\mathbb{Z}/Q\mathbb{Z}$, $Q = q_1 \cdots q_r$, in the case where $\{q_1, \cdots, q_r\}$ are the first r prime numbers. By virtue of the inequalities [23] $\ln Q < 1.01624 q_r$, and $q_r < r\ln(r\ln r)$ for $r \geq 6$, the number of bits to encode a scalar in $\mathbb{Z}/Q\mathbb{Z}$ (as an integer between 0 and $Q-1$) is $\lambda(Q) < \lfloor 1.46613\ r\ln(r\ln r)/\mathsf{w}\rfloor + 1$.

Notation. We denote by A_r an upper bound on the time complexities $\mathsf{A}_+(Q)$, $\mathsf{A}_\times(Q)$, and $\mathsf{A}_\div(Q)$, for performing arithmetic operations on integers smaller than Q, where $Q = q_1 \times \ldots \times q_r$ is the product of the r smallest prime numbers $q_1, \ldots, q_r$.

5.2. **Complexity of the Modular Reconstruction Algorithm.** Let $\mathbf{K}$ be a filtered complex of size m. We describe computational complexities in terms of the size of the persistence diagram $P_{\mathbb{F}}$, which is a $\Theta(m)$, and the size of the multi-field persistence diagram P_r. The persistent homology algorithm described in Section 4, applied on $\mathbf{K}$ with coefficients in a field $\mathbb{F}$, requires $O(P_{\mathbb{F}}^3)$ operations in $\mathbb{F}$. For a field $\mathbb{Z}/q\mathbb{Z}$ these operations take constant time and the algorithm has complexity $O(P_{\mathbb{F}}^3)$. The output of the algorithm is the persistence diagram.

For a set of prime numbers $\{q_1, \cdots, q_r\}$, let P_r be the total number of distinct pairs in all persistence diagrams for the persistent homology of $\mathbf{K}$ with coefficient fields $\mathbb{Z}/q_1\mathbb{Z}, \cdots, \mathbb{Z}/q_r\mathbb{Z}$. We express the complexity of the modular reconstruction algorithm in terms of the size of its output P_r, the number of fields r and the arithmetic complexity A_r.

First, note that, for a column j' in the reduced form of $\mathbf{R}$, the size of $\mathcal{L}(j')$ is equal to the number of triplets of the multi-field persistence diagram with death index j'. We denote this quantity by $|\mathcal{L}(j')|$. Hence, when reducing column col_j with $j > j'$, the column $\mathsf{col}_{j'}$ is involved in a column operation $\mathsf{col}_j \leftarrow \mathsf{col}_j + \alpha \cdot \mathsf{col}_{j'}$ at most $|\mathcal{L}(j')|$ times. Consequently, reducing col_j requires $O(\sum_{j'<j} |\mathcal{L}(j')|) = O(P_r)$ column operations. There is a total number of $O(m \times P_r)$ column operations to reduce the matrix, each of them being computed in time $O(m \times \mathsf{A}_r)$.

Computing the partial inverse of an element $x \in \mathbb{Z}/Q\mathbb{Z}$ takes time $O(r \times \mathsf{A}_r)$ in the general case, and only $O(\mathsf{A}_r)$ if x is invertible in $\mathbb{Z}/Q\mathbb{Z}$. The partial inverse of an element $x = \mathsf{col}_j[k]$ is computed only if there is a pair $(k, Q_T) \in \mathcal{L}(j)$. This element is not invertible in $\mathbb{Z}/Q\mathbb{Z}$ iff $|\mathcal{L}(j)| > 1$. There are consequently $O(|P_r - m|)$ non-invertible elements x that are at index $\mathrm{low}(j, Q_T)$ in some column j, for some Q_T. If we store the partial inverses when we compute them, the total complexity for computing all partial inverses in the modular reconstruction algorithm is $O(m + r \times (P_r - m) \times \mathsf{A}_r)$. We conclude that the total cost of the modular reconstruction algorithm for multi-field persistent homology is

$$O\left(\left[r \times (P_r - m) + m^2 P_r\right] \times \mathsf{A}_r\right) = O(\left[r \times (P_r - P_{\mathbb{F}}) + P_r^3\right] \times \mathsf{A}_r)$$

while the brute-force algorithm, consisting in computing persistence separately for every field $\mathbb{Z}/q_1\mathbb{Z}, \cdots, \mathbb{Z}/q_r\mathbb{Z}$ has time complexity

$$O(r \times P_{\mathbb{F}}^3)$$

5.3. **Discussion and Limitations.** Comparing the time complexity of the modular reconstruction algorithm and the brute-force approach, we notice that the former in particularly more efficient than the latter when A_r is not too large, and the difference between persistence diagrams for different coefficient fields are few. In that case, assuming $r \times (P_r - P_{\mathbb{F}}) \ll P_r^3$, the trade-off of time complexities is

$$\frac{r}{\mathsf{A}_r}$$

In light of Section 5.1, the complexity A_r in practice is a near-linear function in the number of memory word $\lambda(Q)$ necessary to store the integer $Q = q_1 \cdots q_r$, for the first r prime numbers. In particular, we note that $\lambda(Q) \ll r$ for $r \ln r \ll e^{\mathsf{w}}$.

We note two limitations to the modular reconstruction algorithm. First, for large numbers r of primes, one arithmetic operation in $\mathbb{Z}/Q_{[r]}\mathbb{Z}$ becomes more costly than r distinct arithmetic operations in $\mathbb{Z}/q_1\mathbb{Z}, \cdots, \mathbb{Z}/q_r\mathbb{Z}$, in which case the modular reconstruction approach developed in this article becomes worse than brute-force (even when P_r and $P_{\mathbb{F}}$ remain close).

Second, for complexes with torsion subgroups of very high order in their homology, the number of distinct pairs P_r in all the persistence diagram may become large.

We study these cases in practice in the experimental Section 7.

6. Complexity Analysis in Terms of Index Persistence

The cubic dependence in the size of the persistence diagram is, in practice, pessimistic. In this section we refine the complexity analysis in terms of the length of persistence intervals, in the spirit of the sparse complexity analysis of the standard persistence algorithm [12]. First, we recall the sparse complexity analysis of the persistent homology algorithm.

Theorem 6.1 (Sparse Complexity Analysis PH [12][Chapter VII)**.** *] With a sparse matrix implementation, where only non-zero matrix coefficients are represented, the algorithm reduces the boundary matrix of a filtered simplicial complex of dimension d in*

$$O\left(d \times \left[\sum_{(i,j)\in\mathcal{P}_{\mathbb{F}}, j\neq\infty} |j-i|^2 + \sum_{(i,\infty)\in\mathcal{P}_{\mathbb{F}}} i^2\right]\right)$$

arithmetic operations in $\mathbb{F}$.

Proof. The proof is identical to the one in [12], except that the "clear" optimization of [8] (see also [2]) allows us to improve the bound. The argument of the proof relies on the fact that:

1. to reduce a column col_j, eventually leading to an interval (i, j) in the diagram, only columns $\mathsf{col}_{j'}$ with $i < j' < j$ are used for the reduction.
2. to reduce a column col_i, eventually leading to an interval (i, ∞) in the diagram, only columns $\mathsf{col}_{j'}$ with $j' < i$ are used for the reduction.
3. any column col_i such that i is the birth index of a finite interval in the diagram can be reduced in $O(1)$ operations using the clear optimization.

The complexity bound can be read directly from this analysis. □

We can deduce almost directly from the proof of Theorem 6.1 the following:

Corollary 6.2. *With a sparse matrix implementation, where only non-zero matrix coefficients are represented, the modular reconstruction algorithm for multi-field persistent homology, with coefficient fields $\mathbb{Z}/q_1\mathbb{Z}, \cdots, \mathbb{Z}/q_r\mathbb{Z}$, applied on a filtered simplicial complex of dimension d,*

has complexity:

$$O\left(\begin{array}{l}\mathsf{A}_r \times r \times (P_r - P_{\mathbb{F}}) + \\ \mathsf{A}_r \times d \times \left[\sum_{(i,j,Q_S)\in\mathcal{P}_r, j\neq\infty} |j-i|^2 \times |\mathcal{L}(j)| + \sum_{(i,\infty,Q_S)\in\mathcal{P}_r} i^2 \times |\mathcal{L}(i)|\right]\end{array}\right)$$

where $|\mathcal{L}(j)|$ is the number of triplets (i, j, Q_S) of the multi-field persistence diagram $\mathcal{P}_r$ dying at index j.

Proof. We note that, when reducing a column col_j in the modular reconstruction algorithm, a column $\mathsf{col}_{j'}$ is added at most $|\mathcal{L}(j)|$ times to col_j, with different multiplicative weights. The rest of the proof is identical to the proof of Theorem 6.1. □

7. Experiments and Applications

In this section we report the performance of the modular reconstruction algorithm for multi-field persistent homology against the brute-force approach consisting in computing persistent homology separately for every field of coefficients. Our implementation is in `C++` and is available within the `Gudhi` software library [21] for topological data analysis. We use the `GMP` library [1] for storing large integers. All timings are measured on a 64 bits Linux machine with 3.00 GHz processor and 32 GB RAM., and are averaged over 10 independent runs.

We compute the persistent homology of Rips complexes [12], which are one of the most popular constructions in topological data analysis, built on a variety of both real and synthetic geometric data, and we compute the persistent homology of a variety of random simplicial complexes. We use the *compressed annotation matrix* implementation of persistence [4] for its efficiency and stability over various datasets. Additionally, the compressed annotation matrix is one of the fast implementations of persistent homology that use few arithmetic operations.

7.1. Description of the Data.

Datasets and running times are presented in Figure 2.

Topological Data Analysis. We use a variety of natural and synthetic geometric data for the running times: **Bud** is a set of points sampled from the surface of the *Stanford Buddha* in $\mathbb{R}^3$. **Bro** is a set of 5×5 *high-contrast patches* derived from natural images, interpreted as vectors in $\mathbb{R}^{25}$, from the Brown database (with parameter $k = 300$ and cut 30%) [7]. **Cy8** is a set of points in $\mathbb{R}^{24}$, sampled from the space of conformations of the cyclo-octane molecule [22], which is the union of two intersecting surfaces. **Kl** is a set of points sampled from the surface of the figure eight Klein Bottle embedded in $\mathbb{R}^5$. Finally **S3** is a set of points distributed on the unit 3-sphere in $\mathbb{R}^4$. Datasets are listed in Figure 2 with the size of point sets $|P|$, the ambient dimensions D and intrinsic dimensions d of the sample points (if known), the thresholds ρ for the Rips complex and the size of the complexes constructed $|\mathcal{K}|$.

In topological data analysis, data points are generally geometric samples of low-dimensional spaces—such as manifolds—embedded in high-dimensions. Their persistence usually show few (or none) long living torsion, of low order.

Random Complexes. We use three distinct models of random simplicial complexes ; see for example [3] for a survey on random complexes. The complex $\mathcal{R}(\mathbf{10000}, \mathbf{0.25})$ is the 5-skeleton of a Rips complex on 10000 uniform random points in the unit cube in $\mathbb{R}^5$, with threshold 0.25, where the filtration in the standard (geometric) Rips filtration. $\mathbf{X}(\mathbf{200}, \mathbf{5000})$ is the 5-skeleton of a random flag complex on 200 vertices with 5000 random edges, where the filtration is induced by an ordering of the edges. $\mathbf{Y_2}(\mathbf{50}, \mathbf{3000})$ is a Linial-Meshulam random 2-complex on 50 vertices with 3000 random triangles, where the complex is filtered by an ordering of the triangles.

Data	$\lvert P\rvert$	D	d	ρ	$\lvert\mathcal{K}\rvert$	T_1	R_1	T_{50}	R_{50}	T_{100}	R_{100}	T_{200}	R_{200}
Bud	49,990	3	2	0.09	$127 \cdot 10^6$	96.3	0.51	110.3	22.2	115.9	42.3	130.7	75.0
Bro	15,000	25	?	0.04	$142 \cdot 10^6$	123.8	0.41	143.5	17.8	150.2	34.0	174.5	58.5
Cy8	6,040	24	2	0.8	$193 \cdot 10^6$	121.2	0.63	134.6	28.2	139.2	54.6	148.8	102.2
Kl	90,000	5	2	0.25	$114 \cdot 10^6$	78.6	0.52	89.3	23.0	93.0	44.1	105.2	78.0
S3	50,000	4	3	0.65	$134 \cdot 10^6$	125.9	0.40	145.7	17.2	152.6	32.8	177.6	50.3

Data	D	T_1	R_1	T_{50}	R_{50}	T_{100}	R_{100}	T_{200}	R_{200}
$\mathcal{R}(\mathbf{10000}, \mathbf{0.25})$	5	11.6	0.49	14.9	20.0	15.7	37.1	19.4	61.1
$\mathbf{X}(\mathbf{200}, \mathbf{5000})$	5	10.55	0.52	34.9	21.6	47.5	29.0	67.6	41.1
$\mathbf{Y_2}(\mathbf{50}, \mathbf{3000})$	2	0.22	0.42	1.55	4.69	3.36	4.8	6.9	3.8

FIGURE 2. Timings T_r of the modular reconstruction algorithm for the first r prime numbers, and ratio R_r with the brute-force algorithm. Top: Rips complexes on geometric data from topological data analysis. Bottom: Diverse models of random complexes.

These complexes usually show a lot of torsion in their persistence, that may be of high order. In particular, Linial-Meshulam random 2-complexes on n points are known to show experimentally a burst of torsion, of potentially super-exponential order in n.

7.2. **Time Performance of the Algorithm.** In Figure 2, the values T_r for $r \in \{1, 50, 100, 200\}$ refers to the running time of the modular reconstruction algorithm for the r first prime numbers, and R_r refers to the ratio between the timings of the brute-force approach (cumulating timings for persistence in every coefficient field), and the timings of the modular reconstruction algorithm. Timings are average over 10 independent running times, picking up new instances of complexes for the random complexes.

Topological Data Analysis. Interestingly, we observe that on all experiments the number of differences between persistence diagrams with various coefficient fields is small. Following Section 5.3, the quantity $P_r - P_{\mathbb{F}}$ can be considered to be a small constant in our experiments. We have also observed that these differences appeared for small prime numbers q_s. Consequently, the linear dependence in r from component $r \times (P_r - P_{\mathbb{F}})$ of the complexity analysis in Section 5 is negligible experimentally. We can consider that, experimentally, the ratio between the brute-force timings and the modular reconstruction timings is at most

$$\frac{r}{\mathsf{A}_r}$$

where, in light of the discussion of Section 5.1, A_r is a small constant for small to medium values of r (here, $r \leq 200$). Specifically, for $q_1, \ldots, q_r$ the r first prime numbers and on a 64 bits machine, the number of memory words necessary to represent the product $Q = q_1 \times \ldots \times q_r$ is $\lambda(Q) = 7, 15,$ and 32 for $r = 50, 100,$ and 200 respectively. Additionally, the optimized implementation of persistent homology using fewer arithmetic operations, the trade-off r/A_r is pessimistic. These considerations are confirmed by the experiments.

Figure 2 presents the timings of the modular reconstruction approach for a variety of filtered simplicial complexes ranging between 114 and 193 million simplices. We note that from $r = 1$ to $r = 200$ prime numbers, the time for computing multi-field persistence using the modular reconstruction approach only increases by 23 to 41%, when the brute-force approach requires about 200 times more time, as expected. This difference appears in the speed-up expressed by the ratio R_r. For $r = 1$, the modular reconstruction approach is about twice slower than the

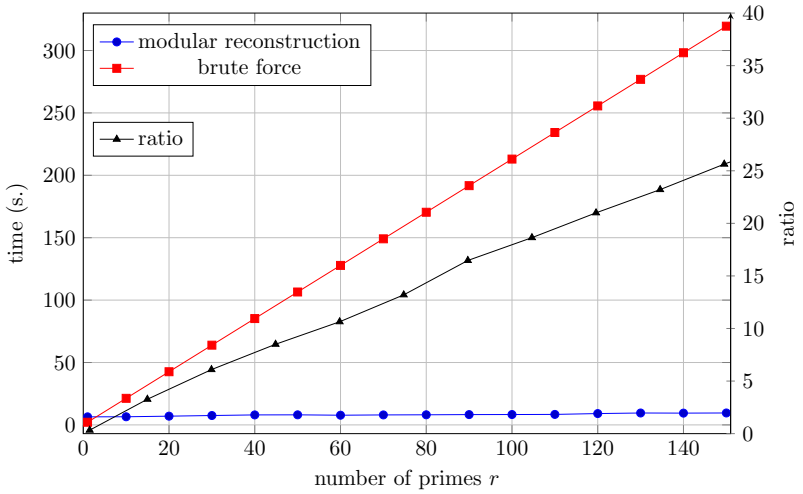

FIGURE 3. Timings for the modular reconstruction algorithm and brute force.

standard persistent homology algorithm in one field, because modular reconstruction is a more complex procedure and deals, in our implementation, with `GMP` integers that are slower than the classic `int` used in the standard persistent homology algorithm. However, this difference fades away as soon as $r > 1$ and the modular reconstruction is significantly more efficient than brute-force: it is, in particular, between 50.3 and 102.2 times faster for $r = 200$. We study the asymptotic behaviour of the running times for large values of r in Section 7.3.

Random Complexes. Figure 2 presents timings for the modular reconstruction on random complexes $\mathcal{R}(\mathbf{10000}, \mathbf{0.25})$ and $\mathbf{X}(\mathbf{200}, \mathbf{5000})$, despite the appearance of more torsion in their persistent homology. Indeed, for $r = 1$ we observe that the modular reconstruction algorithm is about twice slower due to the manipulation of `GMP` integers, but for increasing values of r the modular reconstruction approach gets faster, and is in particular between 41 and 61 times faster for $r = 200$.

The case of the random Linial-Meshulam complex $\mathbf{Y_2}(\mathbf{50}, \mathbf{3000})$ shows the limit of the approach, and the difference of running times is not as remarkable. These complexes show short torsion in their persistent homology (see Section 7.4 for an analysis) but the torsion subgroups of $H_1(Y_2)$ are of very high order. Following Section 5.3, the difference $P_r - P_{\mathbb{F}}$ in the complexity analysis increases for larger values of r, becoming non-negligible and hence slowing down the modular reconstruction algorithm.

7.3. **Asymptotic Behaviour in the Number of Primes for Geometric Data.** A limit of the modular reconstruction algorithm is the arithmetic complexity A_r for large r ; see Section 5.3. Additionally, in the case of topological data analysis, where the underlying space of the sample is unknown, the number r of primes used for multi-field persistence is "an exploratory parameter", attempting to find an upper bound q_r on the prime divisor of the torsion coefficients.

Figures 3 and 4 present the evolution of the running time of the modular reconstruction approach and the brute-force approach for an increasing number of fields r (using the first r prime numbers). Persistence is computed for a Rips complex built on a set of 10 000 points sampling a Klein bottle, which contains torsion in its integral homology, resulting in a

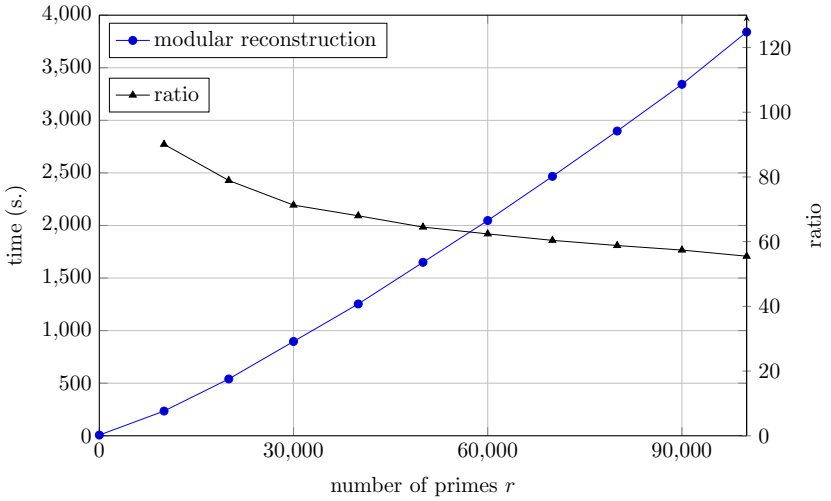

FIGURE 4. Asymptotic behaviour of modular reconstruction and brute force.

simplicial complex of 6.14 million simplices. We analyse the result in terms of the complexity analysis of Section 5. Here again, $P_{\mathbb{F}}$ and P_r remain close during the experiment, even when r grows. The complexity of the brute-force algorithm is $O(r \times P_{\mathbb{F}}^3)$ and we indeed observe a linear behaviour when r increases. The complexity of the modular reconstruction approach is $O(\left[r \times (P_r - P_{\mathbb{F}}) + P_r^3\right] \mathsf{A}_r)$. The part $r \times (P_r - P_{\mathbb{F}})$ of the complexity is negligible because $(P_r - P_{\mathbb{F}})$ is small. For medium values of r (≤ 150), like in Figure 3, the arithmetic complexity $O(\mathsf{A}_r)$ increases slowly because $\lambda(Q_{[r]}) = \left\lfloor \log_2 Q_{[r]}/\mathsf{w} \right\rfloor + 1$ increases slowly. Together with the little use of arithmetic operations, we consequently observe a very slow increase of the time complexity, compare to the one of brute-force.

Figure 4 describes the asymptotic behaviour of the modular approach, where the arithmetic operations become costly. We observe that the timings for the modular reconstruction approach follow a convex curve. The convexity comes from the growth of $\lambda(Q_{[r]})$, which is asymptotically $\Theta(r \log r))$ [23]. However, the increasing of the slope is very slow: all along this experiment, we have been unable to reach a value of r for which the modular approach is worse than the brute-force approach. For readability, the timings for the brute-force approach are implicitly represented through their ratio with the modular approach: all along the experiment presented in Figure 4, for $10\,000 \leq r \leq 100\,000$, the modular approach is between 55 and 90 times faster. Based on a linear interpolation of the timings for the brute-force approach, and a polynomial interpolation of the modular reconstruction timings, we expect the modular reconstruction to become worse than brute-force for a number of primes r bigger than 4.9 million. This is due to both the proximity between persistence diagram and multi-field persistence diagram, and the use of only few arithmetic operations by the persistence implementation.

As a consequence, the modular reconstruction algorithm remains substantially faster than brute force in topological data analysis, for medium to large r.

7.4. **Persistence of Torsion in Random Complexes.** In this section we study the persistence of torsion of Linial-Meshulam random complexes [18]. A Linial-Meshulam 2-complex $Y_2(n, p)$, for an integer n and a probability $0 \leq p \leq 1$, is a random abstract simplicial complex on n vertices made of a complete 1-skeleton, and where every triangle has been added

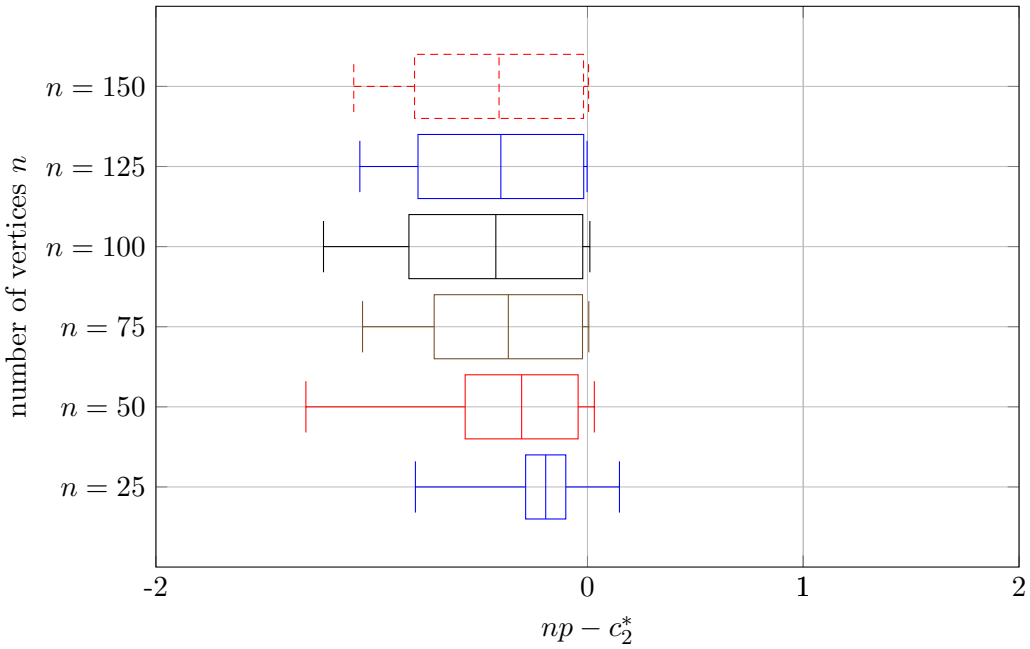

FIGURE 5. Range $np - c$ for which $H_1(Y_2(n,p),\mathbb{Z})$ admits torsion summands $\mathbb{Z}/q^k\mathbb{Z}$, for q one of the first 200 prime numbers.

to $Y(n,p)$ independently with probability p. The homology of these complexes have been extensively studied, and they are known to show a short "burst of torsion" for certain values of the parameter p, with the appearance of a torsion subgroup in homology of experimental super-exponential order [17, 20].

However, the complete understanding of torsion in the homology of these complexes remains a difficult problem. Łuczak and Peled conjecture the following:

Conjecture 7.1 (Łuczak, Peled [20])**.** *For $p = p(n)$ such that $|np - c|$ is bounded away from 0, $H_1(Y_2(n,m),\mathbb{Z})$ is torsion-free asymptotically almost surely, where the constant c is the phase transition constant $c = c_2^*$ of random 2-complexes (see* [19]*[Theorem 1.1]).*

In particular, the burst of torsion happens around $p = c_2^*/n$.

For our experiments, we study the closely related random 2-complex $Y(n,m)$, where m triangles are randomly picked and added to the complex. We use the persistent homology algorithm with torsion to study experimentally the size of the range around $m = c_2^*/n \cdot \binom{n}{3}$ for which $H_1(Y_2(n,m),\mathbb{Z})$ has torsion, for an increasing number of vertices n. Our index-valued filtration on $Y(n,m)$ is induced by the random order with which the m triangles are inserted in the complex (all vertices and edges have filtration value 0). We compute the persistent homology using the modular reconstruction approach for the $r = 200$ first prime numbers.

Figure 5 illustrates the intervals of values of $m \in [0, m_{\max}]$ for which the homology of an instance $Y(n, m_{\max})$ contains torsion summands $\mathbb{Z}/q^k\mathbb{Z}$, for q one of the first 200 prime numbers, and for $n \in \{25, 50, 75, 100, 125, 150\}$ and 25 independent runs for each value of n. The boxes represent the normalized quantity

$$n \cdot m \cdot \binom{n}{3}^{-1} - c_2^*$$

to correspond to Conjecture 7.1. The boxes represent the average lower bound and upper bound (and centre) of the intervals, and the whiskers stand for the extremal values observed in the samples.

Similarly to the study of homology with field coefficients [19], we observe a one-sided sharp transition at $p = c_2^*/n$ for the disappearance of torsion. The plot seems to corroborate the convergence of a lower bound for the interval at a constant $k_2 \approx -0.8$, which suggests that, following Conjecture 7.1, the homology group $H_1(Y_2(n,m),\mathbb{Z})$ is torsion-free a.a.s. when $|np - c| > k_2 \approx -0.8$.

Acknowledgement

The research leading to these results has received funding from the European Research Council (ERC) under the European Union's Seventh Framework Programme (FP/2007-2013) / ERC Grant Agreement No. 339025 GUDHI (Algorithmic Foundations of Geometry Understanding in Higher Dimensions).

Conflict of interest. On behalf of all authors, the corresponding author states that there is no conflict of interest.

References

[1] GMP, the GNU Multiple Precision arithmetic library. http://gmplib.org/. 14
[2] Ulrich Bauer, Michael Kerber, and Jan Reininghaus. Clear and compress: Computing persistent homology in chunks. In *Topological Methods in Data Analysis and Visualization III*, pages 103–117. 2014. 13
[3] Omer Bobrowski and Matthew Kahle. Topology of random geometric complexes: a survey. *Journal of Applied and Computational Topology*, Jan 2018. 14
[4] Jean-Daniel Boissonnat, Tamal K. Dey, and Clément Maria. The compressed annotation matrix: An efficient data structure for computing persistent cohomology. *Algorithmica*, 73(3):607–619, 2015. 2, 14
[5] Jean-Daniel Boissonnat and Clément Maria. Computing persistent homology with various coefficient fields in a single pass. In *Algorithms - ESA 2014 - 22th Annual European Symposium, Wroclaw, Poland, September 8-10, 2014. Proceedings*, pages 185–196, 2014. 1
[6] Jean-Daniel Boissonnat and Clément Maria. Computing persistent homology with various coefficient fields in a single pass. *J. Appl. Comput. Topol.*, 3(1-2):59–84, 2019. 1
[7] Gunnar E. Carlsson, Tigran Ishkhanov, Vin de Silva, and Afra Zomorodian. On the local behavior of spaces of natural images. *International Journal of Computer Vision*, 76(1):1–12, 2008. 1, 3, 14
[8] Chao Chen and Michael Kerber. Persistent homology computation with a twist. In *Proceedings 27th European Workshop on Computational Geometry*, 2011. 13
[9] David Cohen-Steiner, Herbert Edelsbrunner, and John Harer. Stability of persistence diagrams. *Discrete & Computational Geometry*, 37(1):103–120, 2007. 1, 4
[10] Vin de Silva, Dmitriy Morozov, and Mikael Vejdemo-Johansson. Persistent cohomology and circular coordinates. *Discrete & Computational Geometry*, 45(4):737–759, 2011. 2
[11] Tamal K. Dey, Fengtao Fan, and Yusu Wang. Computing topological persistence for simplicial maps. In *Symposium on Computational Geometry*, page 345, 2014. 2
[12] Herbert Edelsbrunner and John Harer. *Computational Topology - an Introduction*. American Mathematical Society, 2010. 1, 2, 5, 6, 13, 14
[13] Herbert Edelsbrunner, David Letscher, and Afra Zomorodian. Topological persistence and simplification. *Discrete & Computational Geometry*, 28(4):511–533, 2002. 2
[14] Martin Fürer. Faster integer multiplication. *SIAM J. Comput.*, 39(3):979–1005, 2009. 12
[15] Joachim Von Zur Gathen and Jurgen Gerhard. *Modern Computer Algebra*. Cambridge University Press, New York, NY, USA, 2 edition, 2003. 6, 7, 11, 12
[16] Allen Hatcher. *Algebraic Topology*. Cambridge University Press, 1 edition, December 2001. 2, 4
[17] M. Kahle, F. Lutz, A. Newman, and K. Parsons. Cohen–Lenstra heuristics for torsion in homology of random complexes. *ArXiv e-prints*, October 2017. 1, 18
[18] Nathan Linial and Roy Meshulam. Homological connectivity of random 2-complexes. *Combinatorica*, 26(4):475–487, Aug 2006. 17

[19] Nathan Linial and Yuval Peled. On the phase transition in random simplicial complexes. *Annals of mathematics*, 184(3):745–773, 2016. 18, 19

[20] Tomasz Łuczak and Yuval Peled. Integral homology of random simplicial complexes. *Discrete & Computational Geometry*, 59(1):131–142, Jan 2018. 3, 18

[21] Clément Maria. Persistent cohomology. In *GUDHI User and Reference Manual*. GUDHI Editorial Board, 2015. 1, 14

[22] S. Martin, A. Thompson, E.A. Coutsias, and J. Watson. Topology of cyclo-octane energy landscape. *J Chem Phys*, 132(23):234115, 2010. 1, 3, 14

[23] J. B. Rosser and L. Schoenfeld. Approximate formulas for some functions of prime numbers. *ijm*, 6:64–94, 1962. 12, 17

[24] Afra Zomorodian and Gunnar E. Carlsson. Computing persistent homology. *Discrete & Computational Geometry*, 33(2):249–274, 2005. 1, 2

Appendix A. Arithmetic Notations

- $\mathbb{Z}$ ring of integers,
- $\mathbb{Z}/n\mathbb{Z}$ ring of integers modulo $n \geq 2$,
- $\mathbb{Q}$ field of rationals,
- $q_1, \ldots, q_r$ the r first prime numbers, for $r \geq 1$,
- $[r]$ the set $\{1, \ldots, r\}$,
- $Q := q_1 \times \ldots \times q_r$, product of first r prime numbers,
- $Q_S := \prod_{s \in S} q_s$, for a subset of indices $S \subset [r]$,
- indices s, t, r, and set of indices S and T, are reserved to the indexing of prime numbers $\{q_1, \ldots, q_r\}$,
- indices i, j, k and m refer to indices in the filtration of a complex, and hence indices for matrix columns and matrix reduction algorithms.

INRIA Sophia Antipolis-Méditerranée
E-mail address: jean-daniel.boisonnat@inria.fr, clement.maria@inria.fr