Average Complexity of Matrix Reduction for Clique Filtrations

Barbara Giunti
Graz University of Technology
bgiunti@tugraz.at

Guillaume Houry
École Polytechnique Palaiseau
guillaume.houry@live.fr

Michael Kerber
Graz University of Technology
kerber@tugraz.at

Abstract

We study the algorithmic complexity of computing persistent homology of a randomly chosen filtration. Specifically, we prove upper bounds for the average fill-up (number of non-zero entries) of the boundary matrix on Erdős–Rényi and Vietoris–Rips filtrations after matrix reduction. Our bounds show that, in both cases, the reduced matrix is expected to be significantly sparser than what the general worst-case predicts. Our method is based on a link between the fill-up of the boundary matrix and expected Betti numbers of random filtrations. Our bound for Vietoris–Rips complexes is asymptotically tight up to logarithmic factors. We also provide an Erdős–Rényi filtration realising the worst-case.

CCS Concepts

• **Mathematics of computing → Computations on matrices**; *Algebraic topology*; • **Theory of computation → Computational complexity and cryptography**.

Keywords

Matrix reduction, Average complexity, Persistence algorithm

ACM Reference Format:
Barbara Giunti, Guillaume Houry, and Michael Kerber. 2022. Average Complexity of Matrix Reduction for Clique Filtrations. In *Proceedings of the 2022 Int'l Symposium on Symbolic and Algebraic Computation (ISSAC '22), July 4–7, 2022, Villeneuve-d'Ascq, France.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3476446.3535474

1 Introduction

Motivation and results. The standard algorithm used to compute persistent homology, first introduced in [14], is based on the Gaussian reduction of the boundary matrix. It performs left-to-right column additions until the lowest elements of non-zero columns in the matrix are pairwise distinct; the matrix is called **reduced** in this case. For a $(r \times c)$-boundary matrix with $r \leq c$, this reduction process runs in $O(r^2 c)$ time, and this complexity can be achieved by a concrete family of examples [25]. Designing such worst-case examples requires some care – for instance, the boundary matrix necessarily has to become dense (i.e., has $\Omega(r^2)$ non-zero entries) during the reduction. On the other hand, such dense reduced matrices are hardly formed in realistic data sets, and the reduction algorithm scales closer to linear in practice [3, 27]. This leads to the hypothesis that the worst-case examples are somewhat pathological, and the "typical" performance of the algorithm is better than what the worst-case predicts. The motivation of this paper is to provide formal evidence for this hypothesis.

To formalize the notion of a typical example, we fix a random variable that captures a boundary matrix based on a random model and analyze the expected runtime of the reduction algorithm over this random variable. We study two instances of random models:

Erdős–Rényi model. Given n vertices, apply a random permutation on the $\binom{n}{2}$ edges, and build the clique filtration over this edge order.

Vietoris–Rips model. In the d-dimensional unit cube (with d arbitrary), place n points uniformly, sort the edges by length, and build the clique filtration over the edge order.

In both cases, the resulting boundary matrices in degree 1 consist of $\binom{n}{2} = \Theta(n^2)$ rows and $\binom{n}{3} = \Theta(n^3)$ columns, and the best known worst-case bound for the number of non-zero entries of the reduced matrix is $O(n^4)$.

We refer to the number of non-zero entries of the reduced matrix as the **fill-up**. Our main results are that fill-up is only $O(n^3 \log n)$ in expectation in the Erdős–Rényi case and $O(n^2 \log^2 n)$ in the Vietoris–Rips case. This implies an expected runtime of $O(n^6 \log n)$ and $O(n^5 \log^2 n)$, respectively, in contrast to the general worst-case bound of $O(n^7)$ for matrices of this degree. In the Vietoris–Rips case, our bound on the non-zero entries is asymptotically optimal up to logarithmic factors. We also provide some experiments that suggest that neither our fill-up bound for the Erdős–Rényi case nor our time bound for the Vietoris–Rips case are tight. Finally, we construct a concrete family of clique filtrations for which the reduction algorithm yields a matrix with $\Theta(n^4)$ fill-up and runs in $\Theta(n^7)$ time. This shows that the general worst-case bounds on fill-up and runtime are tight for the Erdős–Rényi model.

Proof outline. The proof is based on a property that we first illustrate for the Vietoris–Rips model. The (unreduced) boundary matrix encodes a filtration $X_1 \subset X_2 \subset \ldots \subset X_m$ with $m = \binom{n}{2}$, where X_i is the clique complex arising from the i shortest edges. Now, consider a non-zero column in the reduced matrix to which at least one column was added, and let k be the index of its lowest non-zero entry. The key observation is that, in this case, the first Betti number of the complex X_k is strictly bigger than the one of X_{k-1}. It is well known that the Betti numbers of Vietoris–Rips complexes undergo a phase transition, and the probability of X_k having a positive first Betti number goes to zero very quickly when k gets large [20]. Hence, it is very unlikely for the reduced boundary matrix to contain a column with many non-zero entries and a large lowest entry. This allows us to upper-bound the number of non-zero entries in expectation.

The same approach also works for Erdős–Rényi filtrations, adapting a probability bound for Betti numbers from [9] to our situation. In fact, we prove the aforementioned connection of reduced column and increasing Betti numbers in any degree and for a more general

ISSAC '22, July 4–7, 2022, Villeneuve-d'Ascq, France.

ISBN 978-1-4503-8688-3/22/07.
https://doi.org/10.1145/3476446.3535474

class of filtrations of clique complexes, of which both Erdős–Rényi and Vietoris–Rips filtrations are special cases.

Related work. There are many variants of the standard reduction algorithm with the goal to improve its practical performance, partially with tremendous speed-ups, e.g. [1–3, 16, 26, 29, 32]. All these approaches are based on matrix reduction and do not overcome the cubic worst-case complexity of Gaussian elimination. We consider only the standard reduction algorithm in our analysis although we suspect that our techniques apply to most of these variants as well. An asymptotically faster algorithm in matrix-multiplication time is known [24], as well as a randomized output-sensitive algorithm that computes only the most persistent features [8]. However, these approaches are not based on elementary column operations and slower in practice.

In the persistence computation, the order of the simplices (and thus of the columns and rows in the boundary matrix) is crucial and can be altered only in specific cases [3, 7]. This order also determines which elements can be used as pivots. For that reason, we did not see how to transfer analyses of related problems, such as the expected complexity of computing the Smith normal form [11] or the study of fill-in for linear algebraic algorithms [12, 17]. These methods require either to interleave row and column operations and swap of rows and columns, or to reorder the rows and columns and cherry-pick the pivots.

The only previous work on the average complexity of persistence computation is by Kerber and Schreiber [23, 31]. They show that, for the so-called shuffled random model, the average complexity is better than what the worst-case predicts. However, the shuffled model is further away from realistic (simplicial) inputs than the two models studied in this paper. Moreover, their analysis requires a special variant of the reduction algorithm while our analysis applies to the original reduction algorithm with no changes. Moreover, the PhD thesis of Schreiber [31] contains extensive experimental evaluations of several random models (including Vietoris–Rips and Erdős–Rényi); our experiments partially redo and confirm these evaluations.

While the computational complexity for persistence has hardly been studied in terms of expectation, extensive efforts have gone into expected topological properties of random simplicial complexes. We refer to the surveys by Kahle [22] and Bobrowski and Kahle [5] for an overview for the general and the geometric case, respectively. From this body of literature, we use the works by Demarco, Hamm, and Kahn [9] and Kahle [20] in our work. There are also recent efforts to study expected properties of persistent homology over random filtrations, for instance the expected length of the maximally persistent cycles in a uniform Poisson process [6], or properties of the expected persistence diagram over random point clouds [10].

Finally, at the best of our knowledge, the only construction to achieve the worst-case running time for matrix reduction is by Morozov [25], which however involves only a linear number of edges and triangles with respect to the number of vertices and it is not a full (clique) filtration, as required by our models.

Outline. In Section 2, we introduce basic notions on (boundary) matrices and their reduction and simplicial homology. In Section 3, we focus on clique filtrations and prove the connection between Betti numbers and certain columns of the reduced matrix, which leads to a generic bound for the fill-up. We then apply the general bound for the Erdős–Rényi (Section 4) and Vietoris–Rips (Section 5) filtrations. In Section 6, we compare our bounds with experimental evaluation. In Section 7, we construct a clique filtration realising the worst-case fill-up and runtime. We conclude in Section 8.

2 Basic notions

Matrix reduction. In the following, fix an $(r \times c)$-matrix M over $\mathbb{Z}_2$, the field with two elements, and let its columns be denoted by $M_1, ..., M_c$. For a non-zero column M_i, we let its **pivot** be the row index of its lowest non-zero entry, denoted by $low(M_i)$. We write $\#M_i$ for the number of non-zero entries in the column, and $\#M := \sum_{i=1}^{c} \#M_i$ for the number of non-zero entries in the matrix. Clearly, $\#M \leq rc$; if $\#M$ is significantly smaller than that value (e.g., only linear in c), the matrix is usually called "sparse".

A **left-to-right column addition** is the operation of replacing M_i with $M_i + M_j$ for $j < i$. If M_i and M_j have the same pivot before the column addition, the pivot of M_i decreases under the column addition (or the column M_i becomes zero, if $M_i = M_j$).

Matrix reduction is the process of repeatedly performing left-to-right column additions until no two columns have the same pivot. We fix the following version: we traverse the columns from 1 to c in order. At column i, as long as it is non-zero and has a pivot that appears as a pivot in some column $j < i$, we add column j to column i. The resulting matrix is called **reduced**.

We define the **cost of a column addition** of the form $M_i \leftarrow M_i + M_j$ as $\#M_j$, i.e., the number of non-zero entries in the matrix that is added to M_i. The **cost of a matrix reduction** for a matrix M is then the added cost of all column additions performed during the reduction, and it is denoted by $cost(M)$. The **fill-up** of a reduced matrix M' is $\#M'$, the number of non-zero entries of M'. We can relate the cost of reducing a matrix M to the fill-up of the reduced matrix as follows.

Lemma 2.1. *For a matrix M with c columns, let M' denote its reduced matrix. Then $cost(M) \leq c\#M'$.*

Proof. Let $M'_{\leq i}$ denote the matrix formed by the first i columns of M'. Then, after the matrix reduction algorithm has traversed the first i columns, the partially reduced matrix agrees with M' on the first i columns. In order to reduce column $i+1$, the algorithm adds some subset of columns of $M'_{\leq i}$ to M_{i+1}, and each column at most once. Hence, the cost of reducing column M_{i+1} is bounded by $\#M'_{\leq i}$. Hence, we can bound

$$cost(M) \leq \sum_{i=1}^{c} \#M'_{\leq i} \leq \sum_{i=1}^{c} \#M' = c\#M'. \qquad \square$$

We interpret the cost of M as a model of the (bit) complexity for performing matrix reduction. Indeed, in practice, we will apply matrix reduction on (initially) sparse matrices whose columns are usually represented to contain only the indices of non-zero entries to reduce memory consumption. If we arrange these indices in a balanced binary search tree structure, for instance, performing a column operation $M_i \leftarrow M_i + M_j$ can be realized in $O(\#M_j \log \#M_i)$ time, which matches our cost up to a logarithmic factor. Alternatively, we can store columns as linked lists of non-zero indices and then, reducing column i, we can transform the column in a $\{0,1\}$-vector of length r, and perform all additions in time proportional to $\#M_j$, resulting in a total complexity of $O(c(r+\#M'))$. This complexity matches $cost(M)$ if the reduced matrix has at least $\Omega(r)$ non-zero entries (which will be the case for the cases studied in this paper). We refer to [3] for a more thorough discussion on the possible choices of data structures for (sparse) matrices.

Staircase matrices. We say that a matrix is **in staircase shape** (or **staircase shaped**) if for any $j < i$ such that M_j and M_i are non-empty, we have that the pivot of M_j is not larger than the pivot of M_i. In other words, the pivots of M, read from left to right and ignoring empty columns, form a non-decreasing sequence. We call a non-zero column M_i with pivot p a **step column** if it is the column with smallest index that has p as pivot (step columns and their pivots are also called **apparent pairs** [2, 4]).

Let M' be the reduction of the staircase shaped matrix M. Let $1 \le p \le r$ be a row index that appears as a pivot in some column of M'. We call p a **step index** if p is also a pivot of some column in M. Otherwise, p is called a **critical index**. See Figure 1 for an illustration of these concepts.

$$
\begin{array}{c|cccc|}
 & \text{abc} & \text{acd} & \text{abd} & \text{bcd} \\
\hline
\text{bc} & 1 & & & 1 \\
\text{ad} & & 1 & 1 & \\
\text{ab} & 1 & & 1 & \\
\text{cd} & & 1 & & 1 \\
\text{ac} & 1 & 1 & & \\
\text{bd} & & & 1 & 1
\end{array}
\xrightarrow[\text{reduction}]{\text{left-to-right column}}
\begin{array}{c|cccc|}
 & \text{abc} & \text{acd} & \text{abd} & \text{bcd} \\
\hline
\text{bc} & 1 & 1 & & \\
\text{ad} & & 1 & 1 & \\
\text{ab} & 1 & 1 & 1 & \\
\text{cd} & & 1 & & \\
\text{ac} & 1 & & & \\
\text{bd} & & & 1 &
\end{array}
$$

Figure 1: Example of a (1-boundary) matrix in staircase shape (on the left) and its reduced matrix (on the right), from the complex in Figure 2. The blue column is a step column, the row index of the red elements is a step index, and the row index of the green element is a critical index. The zeros are not displayed.

The next lemma is simple, yet crucial for our approach:

Lemma 2.2. Let M be staircase shaped. If M'_i is a column whose pivot is a step index, we have that $M'_i = M_i$. Moreover, writing $\mathcal{S}$ for the set of all step columns of M, we have that

$$\#M' \le \sum_{S \in \mathcal{S}} \#S + \sum_{p \text{ is critical index of } M'} p$$

Proof. Let M_i be the step column of M with pivot p. Because M is staircase shaped, there cannot be any column with index $j < i$ that has i as a pivot. Hence, $M_i = M'_i$. Moreover, every column of M' that has a step index as pivot necessarily is a step column of M, so the number of non-zero entries over all columns with a step index as pivot equals the first term. The columns with critical index as pivot are bounded by the second term because a column with pivot p can have at most p non-zero entries. □

(Filtered) simplicial complexes and boundary matrices. Given a finite set V, a **simplicial complex** K over V is a collection of (non-empty) subsets of V, called **simplices**, with the property that if $\sigma \in K$ and $\tau \subset \sigma$, also $\tau \in K$. A **subcomplex** L of a simplicial complex K is a subset of K which is itself a simplicial complex. A simplex with $(k+1)$ elements is called k-**simplex**. 0-, 1-, and 2-simplices are also called **vertices**, **edges**, and **triangles**, respectively. For a k-simplex $\sigma \in K$, we call a $(k-1)$-simplex τ with $\tau \subset \sigma$ a **facet** of σ. The set of facets is called the **boundary** of σ.

We call the basic notions of simplicial homology (with coefficients over the field $\mathbb{Z}_2$): for a simplicial complex K, the k**-th chain group** C_k is the vector space over $\mathbb{Z}_2$ that has the k-simplices of K as basis elements. Let $\partial_k : C_k \to C_{k-1}$ denote the unique homomorphism that maps every k-simplex σ to the sum of its facets. We call the kernel $Z_k(K)$ of ∂_k the k**-th cycle group** and the image $B_k(K)$ of ∂_{k+1} the k**-th boundary group**; note $B_k(K) \subseteq Z_k(K)$ because $\partial_k \circ \partial_{k+1} = 0$. The k**-th homology group** $H_k(K)$ of K is then defined as $Z_k(K)/B_k(K)$. Note that despite the name "group" for chains, cycles, boundaries, and homologies, all these objects are vector spaces (because we take coefficients over $\mathbb{Z}_2$). The k**-th Betti number** of K, denoted by $\beta_k(K)$, is the dimension of $H_k(K)$.

A **filtered simplicial complex** is a sequence of nested simplicial complexes. Given a filtered simplicial complex, we see its simplices as totally ordered first by the entrance (i.e. the step in the sequence when they first appear) and then by degree, with the ties broken arbitrarily. The **boundary matrix** D of a filtered simplicial complex in dimension k is a $(r \times c)$-matrix with r the number of k-simplices, c the number of $(k+1)$-simplices, and the entry (i,j) in the matrix to be 1 if the i-th k-simplex (with respect to the fixed order) is a facet of the j-th $(k+1)$-simplex of K, and 0 otherwise. We interpret boundary matrices as matrices over $\mathbb{Z}_2$. For a boundary matrix D in dimension k, we have that $\#D = (k+2)c$ because every $(k+1)$-simplex has $k+2$ facets. Hence, boundary matrices are sparse, but this sparsity is not necessarily preserved by matrix reduction [25].

The matrix reduction gives a wealth of information when applied on a filtered boundary matrix D. For once, it yields the rank of D, which can be used, for instance, to compute the Betti numbers of the simplicial complex: Writing D and $\overline{D}$ for the boundary matrix in dimension k and $k-1$, respectively, and n_k for the number of k-simplices, we have that $\beta_k(K) = n_k - \text{rank}(D) - \text{rank}(\overline{D})$. Moreover, since the matrix reduction respects the order of the simplices, the pivots of the reduced matrix yield the so-called **barcode** of the filtered simplicial complex, which reveals the existence and lifetime of topological features. We refer to [13, 28] for further details about persistent homology, which are not needed for the rest of the paper.

3 Fill-up for clique filtrations

Clique filtrations. Fix a filtered simplicial complex K with (ordered) k-simplices $\sigma_1, \ldots, \sigma_r$. We define the k**-clique complex** K_i with $0 \le i \le r$ as the largest subcomplex of K that contains exactly $\sigma_1, \ldots, \sigma_i$ as k-simplices. Note that each K_i necessarily contains all ℓ-simplices of K with $\ell < k$, but that is not the case for simplices of dimension $\ge k$. Also, note that $K_i \subset K_j$ for $i < j$, and the inclusions $K_0 \subset K_1 \subset \ldots \subset K_r = K$ induce an order on the ℓ-simplices of K for every $\ell > k$: for every ℓ-simplex τ, we define its **entry time** as the smallest index i such that $\tau \in K_i$, and we order the ℓ-simplices such that the order respects the entry times, with ties broken arbitrarily. We are especially interested in the case $\ell = k+1$: the simplices with entry time i correspond exactly to the columns of the boundary matrix in dimension k with pivot i. This implies that the boundary matrix D of dimension k is in staircase shape, if the k-simplices and $(k+1)$-simplices are sorted in the given order.

The most famous example for the above construction is the case of a **Vietoris–Rips** filtration for $k = 1$, where for n points in a metric space, the $\binom{n}{2}$ edges are sorted by length, defining an order among them, and triangles are added to the filtration as soon as all boundary edges are present. Indeed, the entry time of a triangle is the index of the longest boundary edge in this case.

Fill-up analysis. From now on, we fix K to be the complete complex over n vertices, yielding a boundary matrix D with $r = \binom{n}{k+1} = \Theta(n^{k+1})$ rows and $c = \binom{n}{k+2} = \Theta(n^{k+2})$ columns, in which the order of k-simplices is arbitrary and the order of $(k+1)$-simplices respects the k-clique filtration. Let D' be the reduction of D. We have a simple lower bound on the fill-up of D':

Lemma 3.1. $\#D' \geq \binom{n}{k+1} - \binom{n}{k} = \Omega(n^{k+1})$.

Proof. Let $\overline{D}$ denote the boundary matrix in dimension $k-1$. Observe that $\#D' \geq \operatorname{rank}(D)$ because there will be at least one non-zero entry at the pivot entries. By the aforementioned formula of Betti numbers, we have $\operatorname{rank}(D) = n_k - \beta_k(K) - \operatorname{rank}(\overline{D})$, where K is the complete complex over n vertices. The statement follows because the Betti number of K is 0 in all dimensions $k \geq 1$ and $\overline{D}$ is a $(\binom{n}{k} \times \binom{n}{k+1})$-matrix whose rank is at most $\binom{n}{k}$. □

We now turn to an upper bound for $\#D'$. By Lemma 2.2, we get

$$\#D' \leq (k+2)\binom{n}{k+1} + \sum_{p \text{ is critical index of } D'} p \tag{1}$$

since every column of D has exactly $k+2$ non-zero entries. Assuming k to be a constant, the first term further simplifies to $\Theta(n^{k+1})$.

For the second term, we link the presence of a critical index p with the homology of the k-clique complex K_p (see Figure 2):

Lemma 3.2. If p is a critical index of D', then $\beta_k(K_p) > \beta_k(K_{p-1})$.

Proof. Let σ_p denote the k-simplex indexed by p. In K_p, there is no $(k+1)$-simplex with facet σ_p: else p would be a step index, in contradiction with Lemma 2.2. Hence, by definition of k-clique filtration, $K_p \setminus \sigma_p = K_{p-1}$.

Since p is a pivot in D', the non-zero entries in its column encode a k-cycle z. All k-simplices that belong to z are in K_p because p is the pivot, but not in K_{p-1} as σ_p is not. Thus, $z \in Z_k(K_p)$ and $z \notin Z_k(K_{p-1})$. However, $B_k(K_p)$ is generated by the boundaries of $(k+1)$-simplices in K_p. Written as columns in D, each of them has a pivot $< p$, so no linear combination of them yields z. Thus, z is a non-trivial k-cycle in K_p but not in K_{p-1} and the claim follows. □

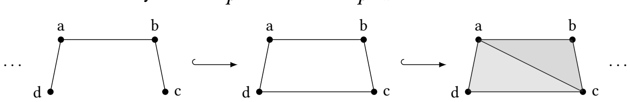

Figure 2: The insertion of the edge cd - that has critical index as Figure 1 shows - create a 1-cycle, i.e., increases β_1.

In combination with (1), the lemma gives

$$\#D' \leq \Theta(n^{k+1}) + \sum_{i=0}^{r} i \mathbb{1}_{(\beta_k(K_i) > \beta_k(K_{p-1}))}, \tag{2}$$

with $\mathbb{1}_{(\beta_k(K_i) > \beta_k(K_{p-1}))}$ the indicator function that is 0 if a new k-th homology group is created, and 1 otherwise. In the worst-case, β_k increases in $\Theta(r)$ step, and the bound yields $O(r^2)$, which can also be derived directly as an upper bound on $\#D'$.

Average complexity. We fix n, the number of vertices, and the dimension k, and we consider a random variable X that yields an order of the k-simplices. Now, the induced k-clique filtration, the induced order on the $(k+1)$-simplices, the boundary matrix D, its reduction D', and the fill-up $\#D'$ are random variables as well depending on X. Reviewing (2), we see a better bound is possible for the fill-up in expectation if the creation of new k-homology becomes unlikely for large row indices. More formally, we have the following conditional bound.

Lemma 3.3. For $T \geq 1$ an integer and $A > 0$ a real value, assume that $\mathbb{P}(\beta_k(K_i) > \beta_k(K_{i-1})) < A/r$ for all $i > T$ (with $r = \binom{n}{k+1}$ as before). Then we have that $\mathbb{E}[\#D'] = O((1+A)n^{k+1} + T^2)$.

Proof. We split the second term of (2) in two parts:

$$\sum_{i=0}^{r} i \mathbb{1}_{(\beta_k(K_i) > \beta_k(K_{i-1}))} = \sum_{i=0}^{T-1} i \underbrace{\mathbb{1}_{(\beta_k(K_i) > \beta_k(K_{i-1}))}}_{\leq 1} + \sum_{i=T}^{r} i \underbrace{\mathbb{1}_{(\beta_k(K_i) > \beta_k(K_{i-1}))}}_{\leq A/r} \leq T^2 + Ar,$$

and the result follows by adding the first term of (2). □

In the next two sections, we will show that the assumption of the lemma holds for two basic cases of the random variable X. In both cases, we assume $k = 1$, so that the random variable picks an order on the $\binom{n}{2}$ edges of the simplicial complex with n vertices.

4 Complexity for the Erdős–Rényi filtration

Our first choice of a random variable is simply to pick one of the $\binom{n}{2}!$ possible edge orders uniformly at random. This process can be described as follows: Take the Erdős–Rényi model $G(n, Z)$, for Z natural number, of a graph with n vertices and Z edges. Then consider K_i as the clique complex given by the first i edges. Hence, the filtration of clique-complexes can be seen as a filtration of Erdős–Rényi graphs (and their clique complexes) with increasing integer threshold. We thus call this random variable the **Erdős–Rényi filtration model**. We shortcut "Erdős–Rényi" with ER from now.

Our goal is to show that the boundary matrix of an ER filtration (in dimension 1) does not fill up too much under matrix reduction using Lemma 3.3. For that, we show that for i large enough, the probability of $\beta_1(K_i) > 0$ is small, which bounds the probability that $\beta_k(K_i) > \beta_k(K_{i-1})$.

We proceed in two steps. First, we consider the probability of $\beta_i(X) > 0$, where X is the clique complex of an ER graph. For that, let $G(n, p)$ be the ER model on n vertices with edge probability p.

Lemma 4.1. There are constants $\kappa > 0$ and $\rho > 0$ such that if $p > \rho\sqrt{\frac{\log n}{n}}$, $G \sim G(n, p)$ (i.e., G is a randomly chosen ER graph) and X is the clique complex of G, then $\mathbb{P}(\beta_1(X) > 0) < \kappa/n^4$.

This result is almost given in [9, Theorem 1.2] : they show that for $\rho = \frac{3}{2}$, $\beta_1(X) = 0$ with high probability (i.e., the probability goes to 1 when n goes to ∞). We will need the stated speed of convergence for our proof, and the proof of [9] yields this guarantee (with a constant larger than 3/2). However, proving this requires us to go through a large part of the technical details of that paper. We defer to [15] for these details.

Lemma 4.1 is not sufficient to bound the probability of $\beta_1(K_i)$ being non-zero. The reason is that K_i is not picked using the model $G(n,p)$ with a fixed p, but $G(n,i)$ with a fixed number of edges i. However, we can derive a (crude) bound to relate the two:

Lemma 4.2. Let $p = \frac{i}{r}$, $G \sim G(n,p)$, and X denote the clique complex of G. Then $n^2\mathbb{P}(\beta_1(X) > 0) \geq \mathbb{P}(\beta_1(K_i) > 0)$.

Proof. We have that

$$\begin{aligned}\mathbb{P}(\beta_1(X) > 0) &\geq \mathbb{P}((\beta_1(X) > 0) \wedge |G| = i)\\ &= \mathbb{P}((\beta_1(X) > 0) \,|\, |G| = i)\,\mathbb{P}(|G| = i)\ .\end{aligned}$$

We claim that the first factor is equal to $\mathbb{P}(\beta_1(K_i) > 0)$. Indeed, under the condition of having exactly i edges, G is just drawn uniformly at random among all i-edge graphs with n vertices (because of the symmetry of the ER model), just as in the ER filtration model at position i.

For the second factor, we observe that the number of edges is a binomial distribution whose expected value is equal to the integer $rp = i$. It is known that, in this case, the probability is maximized at the expected value (see for instance [19]). Hence, since there are $r + 1 \leq n^2$ possible values for the distribution, we have that $\mathbb{P}(G \text{ has exactly } i \text{ edges}) \geq 1/n^2$. ◻

Main Theorem 1. Let D' be the reduced matrix of the 1-boundary matrix of an ER filtration. Then $\mathbb{E}[\#D'] = O(n^3 \log n)$ and the cost of the matrix reduction is bounded by $O(n^6 \log n)$.

Proof. Let $K_0 \subset K_1 \subset \ldots \subset K_r$ denote the ER filtration. Choose κ and ρ as in Lemma 4.1. Set $T := \rho r\sqrt{\frac{\log n}{n}}$. For every $i > T$, we have that $p := \frac{i}{r} > \rho\sqrt{\frac{\log n}{n}}$. Choosing G and X as in Lemma 4.1 and Lemma 4.2, we can thus bound

$$\mathbb{P}(\beta_1(K_i) > 0) \leq n^2\mathbb{P}(\beta_1(X) > 0) \leq \kappa/n^2 < \kappa/r\,.$$

Hence, the hypothesis of Lemma 3.3 is satisfied using the chosen T and $A := \kappa$. It follows

$$\mathbb{E}[\#D'] = O((1+\kappa)n^2 + \rho^2 r^2 \frac{\log n}{n}) = O(n^3 \log n)$$

proving the first part of the statement. The second part follows by Lemma 2.1 since the number of columns is $O(n^3)$. ◻

5 Complexity for the Vietoris–Rips filtration

Our second random variable is $V = \{v_1, \ldots, v_n\}$, a set of points randomly independently sampled from the uniform distribution in the d-dimensional unit cube $[0,1]^d$ (with d a constant), and the order of edges is determined by their Euclidean length. The **Vietoris–Rips complex at scale** α over V is given by all the simplices whose diameter is at most α, and it is denoted by $R(V,\alpha)$. Thus, fixing $\alpha > 0$, we see that the clique complex of the graph with all edges of length at most α forms the Vietoris–Rips complex at scale α of the chosen point set. Hence, we call this random variable the **Vietoris–Rips filtration model**. We shortcut "Vietoris–Rips" with VR from now.

We proceed similarly to the previous section. First, we use a result that states that the first Betti number of VR complex is unlikely to be non-trivial for large values of α.

Lemma 5.1. There is a constant $\rho > 0$ such that:

$$\mathbb{P}\left(\exists \alpha \geq \rho\left(\frac{\log n}{n}\right)^{1/d} \,:\, \beta_1(R(V,\alpha)) \neq 0\right) < 1/n^2\,.$$

Again, this result is almost available in prior work. Kahle [20, Theorem 6.5] shows that for a smoothly bounded convex shape and a point set sampled uniformly from that set, the VR complex at scale α has trivial first homology asymptotically almost surely, if $\alpha \geq \rho\left(\frac{\log n}{n}\right)^{1/d}$ (Kahle's statement is in fact stronger and contains this statement as special case). Our statement is slightly different because we require that property to hold for the (non-smooth) cube, we want to bound the speed of convergence, and we want the bound to hold uniformly over *all* scales greater than the critical value instead of a single scale. However, Kahle's proof strategy also works for our adapted statement (to the expense of a slightly larger constant ρ). The proof, however, requires us to redo most of the arguments of [20], and we defer the derivation to [15].

We set $\alpha^* := \rho\left(\frac{\log n}{n}\right)^{1/d}$. We let α_i denote the length of the i-th edge in the VR filtration, so that K_i is the VR complex at scale α_i. We show that, for i sufficiently large, it is unlikely that α_i is smaller than α^*. $B(v,r)$ denotes the ball with center v and radius r.

Lemma 5.2. There is a constant $\lambda > 0$ such that, for all $i \geq \lambda n \log n$,

$$\mathbb{P}\left(\alpha_i \leq \alpha^*\right) \leq 1/n^2\,.$$

Proof. Let λ be a fixed constant, to be defined later. Let Y denote the number of edges of length at most α^*. Then $\alpha_i \leq \alpha^*$ if and only if $Y \geq i$. It follows that

$$\mathbb{P}\left(\alpha_i \leq \alpha^*\right) = \mathbb{P}(Y \geq i) \leq \mathbb{P}(Y \geq \lambda n \log n)\ .$$

Hence, it suffices to prove that $\mathbb{P}(Y \geq \lambda n \log n) \leq 1/n^2$ for an appropriate choice of λ.

Write Y_k, for $k = 1, \ldots, n$, for the number of edges of length at most α^* with endpoint k. Then $Y = \frac{1}{2}\sum_{k=1}^{n} Y_k$. If $Y \geq \lambda n \log n$, at least one Y_k has to be at least $2\lambda \log n$. With the union bound this implies

$$\begin{aligned}\mathbb{P}(Y \geq \lambda n \log n) &\leq \mathbb{P}(\exists\, k\colon Y_k \geq 2\lambda \log n)\\ &\leq \sum_{i=1}^{n} \mathbb{P}(Y_i \geq 2\lambda \log n) \leq n\mathbb{P}(Y_1 \geq 2\lambda \log n)\ .\end{aligned}$$

Therefore, to prove the claim, it suffices to show that

$$\mathbb{P}(Y_1 \geq 2\lambda \log n) \leq 1/n^3\,.$$

Fix $v \in [0,1]^d$ arbitrarily and pick $n-1$ further points in $[0,1]^d$ uniformly at random. Let W_v denote the number of points chosen at distance at most α^* to v. We will show that

$$\mathbb{P}(W_v \geq 2\lambda \log n) \leq 1/n^3\,,$$

which proves the previous claim on Y_1. Indeed, if we fix the position of the first vertex of our filtration to be equal to v, then Y_1 and W_v are identically distributed (as $v_2, \ldots, v_n$ are $n-1$ points picked uniformly). So, if the probability of W_v being large is bounded independently of v, the same bound holds for Y_1.

Finally, to prove the bound on W_v, we use Chernoff's bound in the form given in [18, Theorem 2.1]: For a binomial distribution $\xi \sim Binom(N, p)$, and any $\kappa > 0$,

$$\mathbb{P}\left(\xi \geq Np + \kappa\right) \leq \exp\left(-\frac{\kappa^2}{2(Np + \kappa/3)}\right) \leq \exp\left(-\frac{\kappa^2}{2(Np + \kappa)}\right).$$

We apply this bound on W_v, which is a binomial distribution with $N = n - 1$. For p, we use the following upper bound

$$p = \frac{\mathrm{vol}\Big(B(v, \alpha^*) \cap [0,1]^d\Big)}{\mathrm{vol}[0,1]^d} \leq \frac{\mathrm{vol} B(v, \alpha^*)}{\mathrm{vol}[0,1]^d} \leq \nu_d (\alpha^*)^d = \nu_d \rho^d \frac{\log n}{n}$$

where ν_d is the volume of the d-dimensional unit ball. Hence, for $Binom(n-1, p)$, we have that the expected value $(n-1)p$ is at most $\nu_d \rho^d \log n$. We apply Chernoff's bound with $\kappa = 2\lambda \log n - (n-1)p$ and obtain

$$\mathbb{P}\left(W_v \geq 2\lambda \log n\right) \leq \exp\left(-\frac{\kappa^2}{4\lambda \log n}\right).$$

Now we fix $\lambda := \max\{\nu_d \rho^d, 8\}$. With this choice, we have that $\lambda \log n \geq (n-1)p$ and therefore $\kappa \geq \lambda \log n$, so we can further bound:

$$\exp\left(-\frac{\kappa^2}{4\lambda \log n}\right) \leq \exp\left(-\frac{1}{4}\lambda \log n\right) \leq 1/n^2. \qquad \square$$

Main Theorem 2. Let D' be the matrix reduction of the boundary matrix of a Vietoris–Rips filtration. Then $\mathbb{E}[\#D'] = O(n^2 \log^2 n)$ and the cost of the matrix reduction is bounded by $O(n^5 \log^2 n)$.

Proof. We let $K_0 \subset K_1 \subset \ldots \subset K_r$ denote the VR filtration. Choose ρ and λ as in Lemmas 5.1 and 5.2. Set $T := \lambda n \log n$. For every $i > T$, we have that

$$\begin{aligned}\mathbb{P}\left(\beta_1(K_i) > 0\right) &= \mathbb{P}\left(\beta_1(K_i) > 0 \wedge \alpha_i > \alpha^*\right) + \mathbb{P}\left(\beta_1(K_i) > 0 \wedge \alpha_i \leq \alpha^*\right)\\ &\leq \mathbb{P}\left(\exists \alpha > \alpha^*:\ \beta_1(R(V, \alpha)) \neq 0\right) + \mathbb{P}\left(\alpha_i \leq \alpha^*\right)\\ &\leq 1/n^2 + 1/n^2\end{aligned}$$

where the last inequality follows from Lemmas 5.1 and 5.2. Hence, the hypothesis of Lemma 3.3 is satisfied using the chosen T and $A := 1$. It follows $\mathbb{E}[\#D'] = O(2n^2 + (\lambda n \log n)^2) = O(n^2 \log^2 n)$ proving the first part of the statement. The second part follows by Lemma 2.1 since the number of columns is $O(n^3)$. $\square$

Unsurprisingly, our bound implies that the reduced matrix has fewer entries in expectation than the unreduced boundary matrix which has precisely $3\binom{n}{3} = \Theta(n^3)$ non-zero entries. Moreover, Lemma 3.1 implies that the expected fill-up cannot be smaller than quadratic in n, hence our bound is tight up to a factor of $\log^2 n$.

6 Comparison with experimental results

We ran experiments to compare our bounds for fill-up and cost of our two filtration types with the empirical outcome. For that, we used [30] to generate 50 random filtrations for each considered value of n. Then we used [3] to compute the the average fill-up and cost of the obtained reduced boundary matrices. Moreover, in all experiments, we use linear regression on a log-log-scale to calculate the real values λ and e such that the plot is best approximated by the curve λn^e. Similar experiments have been performed in [31].

Figures 5 and 6 show the results for Vietoris–Rips filtrations. For the fill-up (upper plots), we observe an empirical fill-up of $\Theta(n^{2.01})$ which is quite expected because of our upper bound of $O(n^2 \log^2 n)$ and a matching lower bound of $O(n^2)$. The cost (bottom plots) follows a curve of around $\Theta(n^{3.8})$ which is far from our upper bound of $O(n^5 \log^2 n)$, suggesting that our bound on the cost is not tight. This is perhaps not surprising because our bound on the cost is based on the (pessimistic) assumption that the reduction of a column needs to add all previously reduced columns of the matrix to it (see the proof of Lemma 2.1). A tighter upper bound for the cost would have to improve on this part of the argument.

Figures 7 show the results for Erdős–Rényi filtrations. The regression yields an observed complexity of around $\Theta(n^{2.08})$ for the fill-up and $\Theta(n^{5.06})$ for the cost, which are quite far from our upper bounds of $O(n^3 \log n)$ and $O(n^6 \log n)$, respectively. Note that in the proof of Lemma 3.3, we assume that all columns with a pivot smaller than the threshold T are dense, and we use the rather large value of $T = \Theta(\sqrt{n^3 \log n})$ in the proof of Main Theorem 1. We speculate that a tighter bound has to analyze the behavior in this "subcritical regime" more carefully (a possible approach for that might be to use techniques from [17] to find a probabilistic bound on the density of the columns). On the other hand, it is perhaps surprising that the empirical cost seems bigger than the empirical fill-up by a factor very close to n^3. That suggests that, unlike in the Vietoris–Rips case, Lemma 2.1 is not too pessimistic in bounding the cost of the reduction in this case.

7 Worst-case fill-up and complexity

Our upper bounds on average fill-up and cost are smaller than the respective worst-case estimates. However, these estimates are based on the assumption that the reduction algorithm produces dense columns (since the fill-up of a column with pivot i is upper bounded with i). Since the boundary matrix initially has only a constant number of non-zero entries per column, the question is whether such a bound is really achieved in an example, or whether the upper bound is not tight.

Even for general boundary matrices of simplicial complexes, it requires some care to generate just one dense column in the reduced matrix. For the worst-case, however, one has to generate many such columns (to achieve the worst-case fill-up), and ensure that these columns get used in the reduction of subsequent columns (to achieve the worst-case cost). This has been done by Morozov [25] for general simplicial complexes. However, restricting to clique complexes puts additional constraints and invalidates his example. In this section we show the following result:

Theorem 7.1. *For every $n \in \mathbb{N}$, there is a clique filtration over n vertices, for which the left-to-right reduction of the 1-boundary matrix has a fill-up of $\Theta(n^4)$ and a cost of $\Theta(n^7)$.*

This result complements Main Theorem 1 because the clique filtration we construct is a possible instance of the ER filtration model, hence the expected fill-up and cost for this model are indeed smaller than the worst-case by a factor of roughly n.

Idea of the construction. Recall that a clique filtration is not completely fixed by the order of edges: many triangles can be created by the insertion of an edge, forming columns in the boundary matrix with the same pivot, and the order of these columns influences the resulting matrix (even if their order was irrelevant for the expected bounds). Our construction for Theorem 7.1 carefully chooses an edge order and an order of the columns with the same pivot. The details are rather technical, so we start with a more high-level description of the major gadgets of our construction.

The main idea is to define two groups of $\Theta(n^2)$ edges (or equivalently, rows of the boundary matrix) that we call group II rows and group III rows, with group II rows having smaller index than group III rows (the notation is chosen to fit the technical description that follows). We first make sure to produce $\Theta(n^2)$ columns in the reduced matrix such that each column has exactly one non-zero group III element that is its pivot, and $\Theta(n^2)$ non-zero group II elements. We call such columns **fat** for now. Achieving this already yields a fill-up of $\Theta(n^4)$. To get the cost bound, we make sure to produce $\Theta(n^3)$ further columns (i.e., on the right of the fat columns) which we call **costly** columns. They have the property that during the reduction, they reach an intermediate state where they have gathered $\Theta(n^2)$ non-zero elements in group III and their pivot is in this group as well. To complete the reduction, the algorithm is then required to add $\Theta(n^2)$ fat columns to the current column. That means that the cost of reducing a costly column is $\Theta(n^4)$, and since there are $\Theta(n^3)$ costly columns, the bound on the cost follows.

The main question is: how do we produce fat and costly columns? Let us start with fat columns. The key notion is the one of the **cascade**; we refer to Figure 3 for an illustration of the following description. We introduce another set of $\Theta(n^2)$ edges that define group IV (which come after group III in the edge order). Let i denote the row index of some group IV row. Our construction ensures that there is a column with pivot i that has as further entries $i-1$ and one in group II. We select this column as step column for i, so it does not change in the reduction. The set of these step columns forms the cascade. Moreover, we ensure that all entries in group II over all cascade columns are at pairwise distinct indices to avoid unwanted cancellation in later steps.

After the construction the cascade, we include $\Theta(n)$ edges of group V. This generates $\Theta(n^2)$ columns that acquire a group IV pivot during the reduction. Moreover, we ensure that the (partially reduced) column has exactly one non-zero element of group III, and that all these group III indices are pairwise distinct for all columns in this group. To reduce this column further, we have to add the cascade columns, until the non-zero group III element becomes the pivot. While iterating through the cascade, the reduced column accumulates more and more non-zero elements in group II, resulting in a size of $\Theta(n^2)$. This creates the fat columns. Note that no two fat columns are added to each other because we ensure that they have pairwise distinct pivots from group III – again, this avoids unwanted cancellation.

For generating costly columns, the idea is the same: we construct another cascade (using rows of group VI) and then a group VII to ensure that the cascade will fill up columns in the row indices of group III. Afterwards, the reduction has to continue and adds columns with pivots at group III, which are precisely the fat columns from the previous step.

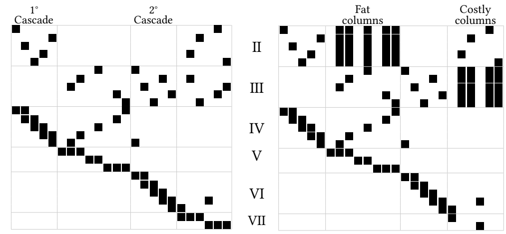

Figure 3: Example of a boundary matrix realising the worst-cases, unreduced (left) and halfway in the reduction (right), i.e. after running through both cascades.

7.2. *Vertices.* Consider five groups of p vertices, for p an odd number, called A, B, C, D, and E. The elements in these groups are enumerated from 0 to $p-1$. An additional vertex, outside these groups, is called $roof$. We construct our clique filtration on these $n = 5p + 1$ vertices. We divide the edges in eight groups I-VIII such that the edge order is given first by the group number and then by some order that we specify inside each group.

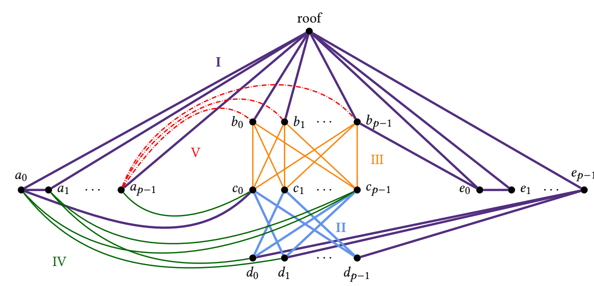

Figure 4: Points of the five groups, where the point x_i belong to the group X, and edges of the first five groups. The edges of group I and II are thicker, the ones of group V are dashed.

7.3. *Groups of edges.* Group I is given by the $3p$ edges between $roof$ and A, $roof$ and B, and $roof$ and E, by the $2(p-1)$ edges that form the path $(a_0, ..., a_{p-1})$ in A and the path $(e_0, ..., e_{p-1})$ in E, by the edges a_0c_0 and $b_{p-1}e_0$, and finally by the $\frac{1}{2}(p^2+p)$ edges between the last $\frac{p+1}{2}$ vertices in E and all the vertices in D. Group II has $p(p-1)$ edges, given by all the edges of the complete bipartite graph between C and D except of the edges c_id_i, for $i = 0, ..., p-1$. Group III is given by all the p^2 edges that form a complete bipartite graph between B and C. The order of the edges inside each of these groups is irrelevant and chosen randomly. The groups IV and VI are given by a subset of cardinality $p^2 - 1$ of the $2p^2$ edges between A and $C \cup D$, and between E and $B \cup C$, respectively. These edges and their order have to be chosen carefully and we describe them in Paragraphs 7.4 and 7.6, respectively. Group V is given by all the p edges from a_{p-1} to B. Group VII is made of the $\frac{(p+1)p}{2}$ edges between $\{b_{\frac{p-1}{2}}, ..., b_{p-1}\}$ and D, and are ordered firstly in decreasing order on the indices in B and then in decreasing order on the indices in D. Finally, the eighth group is given by all the remaining edges, whose order is irrelevant as long as they enter in the filtration after all the previous groups. We do not consider them further. The first five groups are depicted in Figure 4 and the sixth and seventh groups in the zoom-in of Figure 8.

7.4. *Group IV.* Since each vertex in the graph $G = (C \cup D, \text{group II})$ has even degree, there exists an Eulerian path on G, starting in c_0. The edges in group IV are given as follows: for $j = 1, ..., p-1$, starting from the $(jp-j)$-th vertex in the path, we connect $p-1$ consecutive vertices of the Eulerian path to the vertex a_{j-1}. The edges are ordered first by increasing j and then by the order of the Eulerian path. Note that, by construction, none of the columns with pivots in row group IV have a non-zero element in row group III.

We choose as step columns the triangles with two edges in group IV, i.e., the elements in the cascade. Of these triangles, all but $p-1$, i.e., the triangles constructed using the edges between elements in A, have as third edge an element of the Eulerian path. In particular, we have $(p-1)^2$ step columns that have all different elements in the row group II. These columns form the first cascade. The order of the non-step triangles is irrelevant.

7.5. *Step columns from group V.* We fix as step columns all the triangles that have an edge in group V and two edges in group I. The order of all the other $p-1$ triangles created by that edge in V is irrelevant, and we add them as required by the filtration. Note that none of the step columns has a non-zero element in row group III, but each of the non-step columns has one.

7.6. *Edges of group VI and triangles from group VI and VII.* Consider the subset S of edges in the bipartite graph $B \cup C$ given by all the edges but $b_i c_i$, for $i = 0, ..., p-1$. Now every vertex in $B \cup C$ has even degree in S, and thus there exists an Eulerian path on S, starting in b_{p-1}. We give the ordered edges of group VI analogously to how we gave the edges of group IV: for $j = 1, ..., p-1$, starting from the $(jp-j)$-th vertex in the path, we connect $p-1$ consecutive vertices of the Eulerian path to the vertex e_{j-1}. The edges are ordered first by increasing j and then by the order of the Eulerian path.

For the order of the triangles closed by an edge in group VI, we choose as step columns the ones given by triangles with two edges in group VI and one in group III, but for the first step column which is given by the points $b_{p-1}c_{p-2}e_0$. Therefore the step columns form a cascade with all different elements in group III. The step columns with pivots in group VII are given by the triangle with a vertex in B, one in D and one in E. The order of the remaining triangles of both groups is irrelevant and chosen randomly.

Proof of Theorem 7.1. We begin by proving the fill-up, and then we use it to prove the complexity. The edges of group I form some triangles whose reduction is not relevant to the worst-cases. The edges of group II and group III do not form any triangles. The edges of group IV close many triangles; the ones corresponding to the step columns form the cascade and are not reduced. The columns of the other triangles need to be reduced. At the end of their reduction, they will have a pivot in row group II which does not influence the rest of the construction. We now consider group V. By construction, in the reduction of a column t with pivot in row group V, we add the previous step columns with pivot in row group V, moving the pivot of t somewhere in the last $p-1$ rows of group IV. This triggers the cascade reduction, and we add all the $\Theta(p^2)$ columns of the cascade. All the cascade columns have different non-zero elements in row group II and none of them has a non-zero element in row group III. Thus, t accumulates $\Theta(p^2)$ non-zero entries in row group II before exiting the reduction with a pivot in row group III. This procedure has to be repeated for all the $\Theta(p^2)$ non-step columns that the edges of group V form, resulting in $\Theta(p^2)$ columns with $\Theta(p^2)$ elements, for a total fill-up of $\Theta(n^4)$.

We now discuss the complexity. We first note that, by construction, none of the columns with pivots in row group VI or VII has a non-zero element in group IV or V. Moreover, there are $\Theta(p^3)$ non-step columns with pivot in row group VII, given by one of the $\frac{p(p+1)}{2}$ edges in group VII and the $p-1$ points in C. Since the step columns with pivots in row group VII have each a non-zero element in the last half of row group VI, the reduction of those non-step columns pass through at least half of the cascade of group VI, thus accumulating $\Theta(p^2)$ elements in row group III. Now, from the fill-up discussion, the rows of group III are already pivots, specifically of the $\Theta(p^2)$-dense columns discussed above. Therefore, we have $\Theta(p^3)$ columns that accumulates $\Theta(p^2)$ elements, requiring thus an equal amount of operations, each of which flip $\Theta(p^2)$ elements, for a total of $\Theta(n^7)$ complexity. ◻

8 Conclusions and future work

We established upper bounds for fill-up and cost of matrix reduction for two filtration types commonly studied in topological data analysis. We focused on (homological) degree 1. Our approach extends in principle also to the higher-dimensional homology because the boundary matrix of a clique filtration is also in staircase shape for all $k > 1$ if ties are broken in an appropriate manner. In the Erdős–Rényi case, the major obstacle is the generalization of Lemma 4.1, for which we are not aware of a proof (such a generalization is proved for homology with rational coefficients [21]). While the result of [20, Theorem 6.5] also gives a threshold for all higher homology groups to be trivial, the adaption of our proof is not straight-forward because for $k > 1$, several k- and $(k+1)$- simplices can be added to the filtration simultaneously. To extend to other filtration types, it should be noted that our analysis crucially exploits the property of the filtrations to be a sequence of clique complexes; without this property, the connection of critical columns and Betti numbers (Lemma 3.2) fails and a modified approach is needed.

We assumed complete filtrations in this work. It is common in practice to consider truncated filtrations. For instance, for VR filtrations, one often removes all simplices with a diameter greater than a given threshold. In this case, when the number of columns is equal to $c_0 \leq c$, the cost bound of Lemma 2.1 reduces to $c_0 \# D'$.

Our bounds on average fill-up and cost for matrix reduction are better than the currently best known worst-case bounds of $O(n^4)$ and $O(n^7)$, respectively (for a $(\binom{n}{2} \times \binom{n}{3})$-boundary matrix). We showed that these worst-cases can be realised in the ER model, but it is unclear if they can be realised by a VR filtration. We leave the study of these worst-cases for future work.

Acknowledgments. We thank Mickaël Buchet, Matthew Kahle, and Jeff Kahn for useful discussions. This research was funded in whole, or in part, by the Austrian Science Fund (FWF) P 29984-N35 and P 33765-N. For the purpose of open access, the author has applied a CC BY public copyright licence to any Author Accepted Manuscript version arising from this submission. GH thanks TU Graz for the hospitality during his Master Internship.

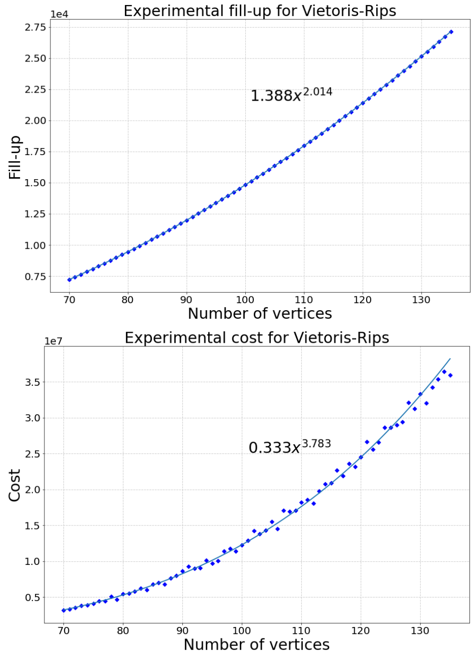

Figure 5: Average fill-up and cost for the reduction of the VR filtration of a random point set sampled uniformly in $[0, 1]^2$. The regression coefficients are shown in the figure.

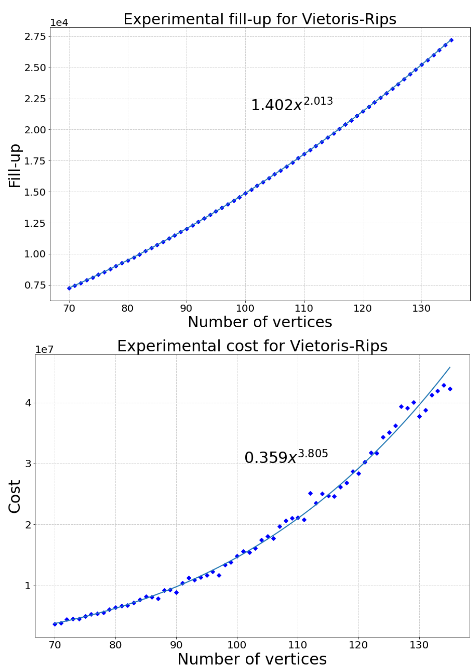

Figure 6: Average fill-up and cost for the reduction of the VR filtration of a random point set sampled uniformly in $[0, 1]^3$. The regression coefficients are shown in the figure.

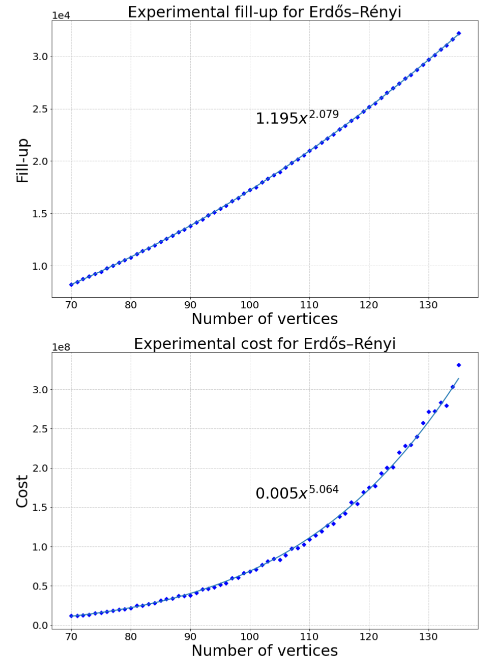

Figure 7: Average fill-up and cost for the reduction of the ER filtration. The regression coefficients are shown in the figure.

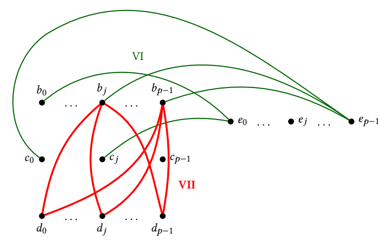

Figure 8: Zoom-in to the edges of groups VI and VII. For readability, $j = \frac{p-1}{2}$. The edges of group VII are thicker.

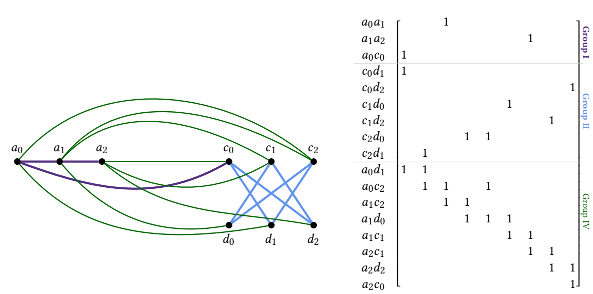

Figure 9: Example of the cascade construction for $p = 3$ given by the edges (left) and relative (sub)matrix (right). For clarity's sake, we depicted only a subset of the edges, namely group II, IV and some of group I. The edges of group I and II are thicker.

References

[1] Henry Adams, Andrew Tausz, and Mikael Vejdemo-Johansson. 2014. JavaPlex: A research software package for persistent (co)homology. In *Mathematical Software – ICMS 2014*, Vol. 8592. Springer Berlin Heidelberg, Lecture Notes in Computer Science, 129–136.

[2] Ulrich Bauer. 2021. Ripser: efficient computation of Vietoris–Rips persistence barcodes. *Journal of Applied and Computational Topology* 5, 3 (2021), 1–33.

[3] Ulrich Bauer, Michael Kerber, Jan Reininghaus, and Hubert Wagner. 2017. Phat–persistent homology algorithms toolbox. *Journal of Symbolic Computation* 78 (2017), 76–90.

[4] Ulrich Bauer and Fabian Roll. 2022. Gromov hyperbolicity, geodesic defect, and apparent pairs in Vietoris-Rips filtrations. In *Proceedings of the 38th Annual Symposium on Computational Geometry (SoCG '22)*. Association for Computing Machinery, New York, NY, USA.

[5] Omer Bobrowski and Matthew Kahle. 2018. Topology of random geometric complexes: a survey. *Journal of Applied and Computational Topology* 1, 3 (2018), 331–364.

[6] Omer Bobrowski, Matthew Kahle, and Primož Škraba. 2017. Maximally persistent cycles in random geometric complexes. *The Annals of Applied Probability* 27, 4 (2017), 2032–2060.

[7] Wojciech Chachólski, Barbara Giunti, Alvin Jin, and Claudia Landi. 2020. Decomposing filtered chain complexes: geometry behind barcoding algorithms. Preprint arXiv:2012.01033.

[8] Chao Chen and Michael Kerber. 2013. An output-sensitive algorithm for persistent homology. *Computational Geometry* 46, 4 (2013), 435–447.

[9] Bobby Demarco, Arran Hamm, and Jeff Kahn. 2013. On the triangle space of a random graph. *Journal of Combinatorics* 4, 2 (2013), 229–249.

[10] Vincent Divol and Frédéric Chazal. 2019. The density of expected persistence diagrams and its kernel based estimation. *Journal of Computational Geometry* 10, 2 (2019), 127–153.

[11] Bruce R. Donald and David R. Chang. 1991. On the complexity of computing the homology type of a triangulation. In *[1991] Proceedings 32nd Annual Symposium of Foundations of Computer Science*. IEEE, San Juan, PR, USA, 650–661.

[12] Iain S. Duff, Albert M. Erisman, and John K. Reid. 1986. *Direct Methods for Sparse Matrices*. Oxford University Press, Inc., USA.

[13] Herbert Edelsbrunner and John Harer. 2010. *Computational Topology - an Introduction*. American Mathematical Society, Providence, RI.

[14] Herbert Edelsbrunner, David Letscher, and Afra Zomorodian. 2000. Topological persistence and simplification. In *Proceedings 41st Annual Symposium on Foundations of Computer Science*. IEEE, Redondo Beach, CA, USA, 454–463.

[15] Barbara Giunti, Guillaume Houry, and Michael Kerber. 2021. Average complexity of matrix reduction for clique filtrations. Preprint arXiv:2111.02125.

[16] Gregory Henselman and Robert Ghrist. 2016. Matroid filtrations and computational persistent homology. Preprint arXiv:1606.00199.

[17] Hsueh Hsieh and Mohammed Ghausi. 1972. A probabilistic approach to optimal pivoting and prediction of fill-in for random sparse matrices. *IEEE Transactions on Circuit Theory* 19, 4 (1972), 329–336.

[18] Svante Janson, Tomasz Luczak, and Andrzej Ruciǹski. 2000. *Random Graphs*. John Wiley & Sons, Ltd, USA.

[19] Rob Kaas and Jan M. Buhrman. 1980. Mean, median and mode in binomial distributions. *Statistica Neerlandica* 34, 1 (1980), 13–18.

[20] Matthew Kahle. 2011. Random geometric complexes. *Discrete & Computational Geometry* 45, 3 (2011), 553–573.

[21] Matthew Kahle. 2014. Sharp vanishing thresholds for cohomology of random flag complexes. *Annals of Mathematics* 179, 3 (2014), 1085–1107.

[22] Matthew Kahle. 2014. Topology of random simplicial complexes: a survey. In *Algebraic topology: applications and new directions*. Contemp. Math., Vol. 620. American Mathematical Society, Providence, RI, 201–221.

[23] Michael Kerber and Hannah Schreiber. 2020. On the expected complexity of matrix reduction for random complexes. In *Computer Algebra in Scientific Computing (CASC)*. Extended abstract.

[24] Nikola Milosavljević, Dmitriy Morozov, and Primož Škraba. 2011. Zigzag Persistent Homology in Matrix Multiplication Time. In *Proceedings of the 27th Annual Symposium on Computational Geometry (SoCG '11)*. Association for Computing Machinery, New York, NY, USA, 216–225.

[25] Dmitriy Morozov. 2005. Persistence algorithm takes cubic time in worst case. *BioGeometry News, Dept. Comput. Sci., Duke Univ* 2 (2005).

[26] Dmitriy Morozov. 2010. Dionysus Library for Computing Persistent Homology. Retrieved from mrzv.org/software/dionysus.

[27] Nina Otter, Mason A. Porter, Ulrike Tillmann, Peter Grindrod, and Heather A Harrington. 2017. A roadmap for the computation of persistent homology. *EPJ Data Science* 6 (2017), 1–38.

[28] Steve Y. Oudot. 2015. *Persistence theory: from quiver representations to data analysis*. Vol. 209. American Mathematical Society, Providence, RI.

[29] Julián Burella Pérez, Sydney Hauke, Umberto Lupo, Matteo Caorsi, and Alberto Dassatti. 2021. Giotto-ph: A Python Library for High-Performance Computation of Persistent Homology of Vietoris–Rips Filtrations. Preprint arXiv:2107.05412.

[30] Dominik Schimd. 2020. Code for generating doundary matrix of random filtration up to degree 1. Available at https://github.com/Dominiksfp/Bac-Programming.

[31] Hannah Schreiber. 2019. *Algorithmic Aspects in standard and non-standard Persistent Homology*. Ph.D. Dissertation. Graz University of Technology.

[32] The GUDHI Project. 2015. *GUDHI User and Reference Manual*. GUDHI Editorial Board. http://gudhi.gforge.inria.fr/doc/latest/