

# Computing Topological Persistence for Simplicial Maps

[Extended Abstract] \*

Tamal K. Dey  
Department of Computer  
Science and Engineering  
The Ohio State University  
Columbus, OH 43210, USA  
tamaldey@cse.ohio-  
state.edu

Fengtao Fan  
Department of Computer  
Science and Engineering  
The Ohio State University  
Columbus, OH 43210, USA  
fanf@cse.ohio-state.edu

Yusu Wang  
Department of Computer  
Science and Engineering  
The Ohio State University  
Columbus, OH 43210, USA  
yusu@cse.ohio-state.edu

## ABSTRACT

Algorithms for persistent homology are well-studied where homomorphisms are induced by inclusion maps. In this paper, we propose a practical algorithm for computing persistence under  $\mathbb{Z}_2$  coefficients for a (monotone) sequence of general simplicial maps and show how these maps arise naturally in some applications of topological data analysis.

A simplicial map can be decomposed into a set of elementary inclusions and vertex collapses—two atomic operations that can be supported efficiently with the notion of simplex annotations for computing persistent homology. A consistent annotation through these atomic operations implies the maintenance of a consistent cohomology basis, hence a homology basis by duality. While the idea of maintaining a cohomology basis through an inclusion is not new, maintaining them through a vertex collapse is new, which constitutes an important atomic operation for simulating simplicial maps. Annotations support the vertex collapse in addition to the usual inclusion quite naturally.

Finally, we exhibit an application of this new tool in which we approximate the persistence diagram of a filtration of Rips complexes where vertex collapses are used to tame the blow-up in size.

## Categories and Subject Descriptors

F.2.2 [Analysis of Algorithms and Problem Complexity]: Nonnumerical Algorithms and Problems—*Geometrical problems and computations*

## General Terms

Algorithms, Theory

\*A full version is available from authors' webpages; this work is supported by NSF grants CCF 1116258, CCF 1318595, CCF 1319406

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from Permissions@acm.org.  
SoCG'14, June 8–11, 2014, Kyoto, Japan.

Copyright 2014 ACM 978-1-4503-2594-3/14/06 ...\$15.00.

## Keywords

Topological persistence, simplicial maps, homology, cohomology, topological data analysis

## 1. INTRODUCTION

Several applications in topological data analysis encounter the following problem: when a simplicial complex  $K_1$  is modified to another complex  $K_2$ , how do the topological features change. If the modification pertains only to inclusions, that is,  $K_1 \subseteq K_2$ , one can quantify the changes by the *persistent homology group*. This idea of topological persistence, originally introduced in [20], has been explored extensively both algebraically and algorithmically in the past decade, see e.g. [3, 5, 9, 7, 12, 19, 24, 26]. When the modification is more general than the inclusions, modeled by considering the map  $K_1 \rightarrow K_2$  to be a simplicial map instead of an inclusion map, the status is not the same. In this paper, we present an efficient algorithm for computing topological persistence for simplicial maps and show its application to a problem in topological data analysis.

Traditional persistent homology is defined for a *monotone* sequence of homomorphisms, where all the maps  $K_i \rightarrow K_{i+1}$  are along the same direction. In [6], Carlsson and de Silva introduced the *zigzag persistence* defined for a zigzagging sequence of homomorphisms containing maps both of the form  $K_i \rightarrow K_{i+1}$  and  $K_i \leftarrow K_{i+1}$ . They also presented a generic prototype algorithm for computing zigzag persistence induced by general homomorphisms. It requires an explicit representation of the homomorphisms between the homology groups of two consecutive complexes in a sequence. In particular, if the input is given in terms of maps between input spaces such as a continuous map  $f : K_i \rightarrow K_{i+1}$ , a representation of the induced homomorphism  $f_* : H_*(K_i) \rightarrow H_*(K_{i+1})$  between the homology groups needs to be computed. Often this step is costly and, in general, leads to  $O(n^4)$  algorithm where each input complex has  $O(n)$  simplices. In contrast, when the map  $f$  is an inclusion, the persistence algorithm computes the persistent homology in  $O(n^3)$  time where  $n$  is the total number of simplices inserted.

Using classical algebraic topological concepts such as mapping cylinders, it is not hard to simulate a simplicial map  $f : K_i \rightarrow K_{i+1}$  by zigzag inclusions through an intermediate complex  $\hat{K}$  built from  $K_i$ . However, the complex  $\hat{K}$ , if constructed naively, may have a huge size. As detailed in section 3, one can improve upon this naïve construction which converts the input zigzag filtration connected by sim-

plicial maps to another zigzag filtration connected only by inclusion maps. One can then take advantage of the efficient algorithms to compute the persistence diagram for an inclusion-induced zigzag filtration [7, 24].

Our main result detailed in Sections 4 and 5 is that when the input filtration is connected by a monotone (i.e., non-zigzag) sequence of simplicial maps, we can improve further upon the above construction by taking advantage of annotations introduced recently in [4]. One of the main advantages of this approach is that it avoids the detour through  $\hat{K}$ , and thus requires far fewer operations to move from  $K_i$  to  $K_{i+1}$ ; see Figure 2. Furthermore, the main auxiliary structure this new direct approach avails is a set of binary bits (elements of  $\mathbb{Z}_2$ ) attached to simplices which together can be viewed as a single binary matrix. This is in contrast to the simple-minded coning approach which uses the zigzag persistence algorithm [7] that requires multiple such matrices.

One key aspect of our annotation based approach is that it lets us simulate the simplicial maps by a sequence of *inclusions* and *vertex collapses* in monotone direction *without* zigzag. An annotation is linked with a cohomology basis which by duality corresponds to a homology basis. Thus, annotations over inclusions and vertex collapses allow us to maintain a consistent homology basis indirectly under simplicial maps and infer the persistent homology. Our handling of inclusions can be seen as an alternative formulation of the algorithm for computing persistent cohomology proposed in [14]. However, the handling of vertex collapses (which are neither inclusions nor deletions) in the context of persistence is new, and has not been addressed previously.

Finally, in Section 6, we show an application where the need for computing persistence under simplicial maps arises naturally. Our algorithm from Section 5 can be used for this application directly. It is known that the persistence diagram [11] of Vietoris-Rips (Rips in short) filtrations provides avenues for topological analysis of data [1, 18, 21]. However, the inclusive nature of Rips complexes makes its size too huge to be taken advantage of in practice. One can consider sparsified versions of Rips complexes [25] or graph induced complexes [17] by subsampling input points which can be achieved by vertex collapses. Our algorithm supports vertex collapses and thus naturally yields to maps arising out of such subsampling.

Throughout the paper, simplicial homology and cohomology groups are defined with coefficients in  $\mathbb{Z}_2$ .

## 2. PRELIMINARIES

**DEFINITION 2.1.** *Given a finite set  $V$ , a simplicial complex  $K = K(V)$  is defined as a collection of subsets  $\{\sigma \subseteq V\}$  so that  $\sigma \in K$  implies that any subset  $\sigma' \subseteq \sigma$  is in  $K$ . The vertex set  $V(K)$  of  $K$  is  $V$ . The elements of  $K$  are called its simplices. An element  $\sigma \in K$  is a  $p$ -simplex if its cardinality is  $p + 1$ . A simplex  $\sigma'$  is a face of  $\sigma$  and  $\sigma$  is a coface of  $\sigma'$  if  $\sigma' \subseteq \sigma$ .*

**DEFINITION 2.2.** *Let  $X$  be a subset of a simplicial complex  $K$ . The set  $\text{St } X := \{\sigma' \in K \mid \exists \sigma \in X \text{ and } \sigma \subseteq \sigma'\}$  is called the star of  $X$ . The closure of  $X$ , denoted  $\bar{X}$ , is the simplicial complex formed by simplices in  $X$  and all of their faces. The link of  $X$  is  $\text{Lk } X := \bar{\text{St } X} \setminus \text{St } X$ .*

The star of  $X$  consists of the simplices in  $K$  that are cofaces of simplices in  $X$ . The link of  $X$  consists of the faces

of the simplices in its star which contain no vertex of  $X$ .

### 2.1 (Co)homology groups

We briefly introduce the notion of homology and cohomology groups here which we use extensively; see e.g. Hatcher [23] for details. Both groups are defined under  $\mathbb{Z}_2$  coefficients. A  $p$ -chain  $c_p$  in a simplicial complex  $K$  is a formal sum of  $p$ -simplices, that is,  $c_p = \sum \alpha_i \sigma_i$ ,  $\alpha_i \in \{0, 1\}$  and  $\sigma_i \in K$ . The chains under  $\mathbb{Z}_2$ -additions form an abelian group called the  $p$ -chain group of  $K$  and is denoted  $C_p(K)$ . The boundary of a  $p$ -simplex  $\sigma$ , denoted  $\partial_p \sigma$ , is defined to be the formal sum of its boundary  $(p-1)$ -simplices. We obtain the boundary homomorphism  $\partial_p : C_p \rightarrow C_{p-1}$  given by  $\partial_p(\sum \alpha_i \sigma_i) = \sum \alpha_i (\partial_p \sigma_i)$ . The kernel of  $\partial_p$  is the cycle group  $Z_p \subseteq C_p$ . The image of  $\partial_p$  is the boundary group  $B_{p-1} \subseteq C_{p-1}$ . It can be easily verified that  $\partial_p \circ \partial_{p-1} = 0$  which makes the quotient group  $H_p(K) = Z_p(K)/B_p(K)$ , known as the  $p$ th homology group, well defined.

Cohomology groups are defined by cochains, cocycles, and coboundaries that are, in a sense, functional duals to the chains, cycles, and boundaries respectively. A  $p$ -cochain is a homomorphism  $c^p : C_p(K) \rightarrow \mathbb{Z}_2$  and thus can be completely specified by its value on each  $p$ -simplex. The  $p$ -cochain group  $C^p(K)$  is the group of all cochains under  $\mathbb{Z}_2$ -additions. The coboundary operator  $\delta_p : C^p \rightarrow C^{p+1}$  sends  $p$ -cochains to  $(p+1)$ -cochains by evaluating  $\delta_p c^p$  on each chain  $d_{p+1} \in C_{p+1}$  as  $c^p(\partial_{p+1} d_{p+1})$ . The kernel of  $\delta_p$  is the cocycle group  $Z^p(K)$  and its image is the coboundary group  $B^{p+1}(K)$ . Since  $\delta_p \circ \delta_{p+1} = 0$ , we have the quotient group  $Z^p(K)/B^p(K)$  well defined which is called the cohomology group  $H^p(K)$ .

### 2.2 Simplicial maps

**DEFINITION 2.3.** *A map  $f : K \rightarrow K'$  is simplicial if for every simplex  $\sigma = \{v_0, v_1, \dots, v_k\}$  in  $K$ ,  $f(\sigma) = \{f(v_0), f(v_1), \dots, f(v_k)\}$  is a simplex in  $K'$ . The restriction  $f_V$  of  $f$  to  $V(K)$  is a vertex map.*

A simplicial map  $f : K_1 \rightarrow K_2$  induces a homomorphism  $H_p(K_1) \xrightarrow{f_*} H_p(K_2)$  for the homology groups in the forward direction while a homomorphism  $H^p(K_1) \xleftarrow{f^*} H^p(K_2)$  for the cohomology groups in the backward direction. The latter sends a cohomology class  $[c]$  in  $H^p(K_2)$  to the cohomology class  $[c']$  in  $H^p(K_1)$  where  $c'(c_p) = c(f(c_p))$  for each  $c_p \in C_p(K_1)$ .

**DEFINITION 2.4.** *A simplicial map  $f : K \rightarrow K'$  is called elementary if it is of one of the following two types:*

- *$f$  is injective, and  $K'$  has at most one more simplex than  $K$ . In this case,  $f$  is called an elementary inclusion.*
- *$f$  is not injective but is surjective, and the vertex map  $f_V$  is injective everywhere except on a pair  $\{u, v\} \subseteq V(K)$ . In this case,  $f$  is called an elementary collapse. An elementary collapse maps a pair of vertices into a single vertex, and is injective on every other vertex.*

We observe that any simplicial map is a composition of elementary simplicial maps (see Appendix A).

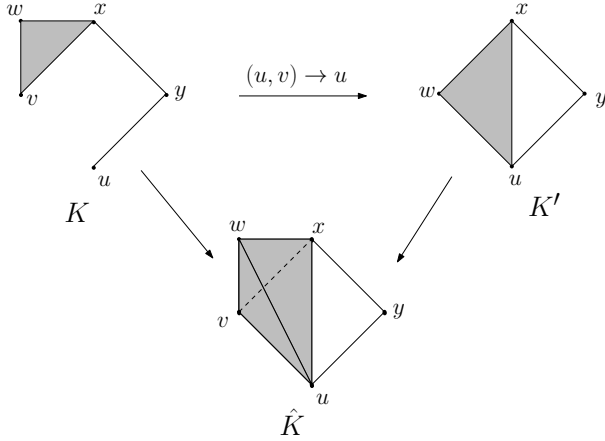
PROPOSITION 2.5. If  $f : K \rightarrow K'$  is a simplicial map, then there are elementary simplicial maps  $f_i$

$$K \xrightarrow{f_1} K_1 \xrightarrow{f_2} K_2 \cdots \xrightarrow{f_n} K_n = K' \text{ so that } f = f_n \circ f_{n-1} \circ \cdots \circ f_1.$$

In view of Proposition 2.5, it is sufficient to show how one can design the persistence algorithm for an elementary simplicial map. At this point, we make a change in the definition 2.4 of elementary simplicial maps that eases further discussions. We let  $f_V$  be identity (which is an injective map) everywhere except possibly on a pair of vertices  $\{u, v\} \subseteq V(K)$  for which  $f_V$  maps to a single vertex, say  $u$  in  $K'$ . This change can be implemented by renaming the vertices in  $K'$  that are mapped onto injectively. Since the standard persistence algorithm handles inclusions, we focus mainly on handling the elementary collapses.

### 3. SIMULATION WITH CONING

First, we propose a simulation of simplicial maps with a coning strategy that only requires additions of simplices. We focus on elementary collapses. Let  $f : K \rightarrow K'$  be an



**Figure 1: Elementary collapse  $(u, v) \rightarrow u$ : the cone  $u * \overline{\text{St } v}$  adds edges  $\{u, w\}, \{u, v\}, \{u, x\}$ , triangles  $\{u, w, x\}, \{u, v, x\}, \{u, v, w\}$ , and the tetrahedron  $\{u, v, w, x\}$ .**

elementary collapse. Assume that the induced vertex map collapses vertices  $u, v \in K$  to  $u \in K'$ , and is identity on other vertices. For a subcomplex  $X \subseteq K$ , define the cone  $u * X$  to be the complex  $\{\sigma \cup \{u\} \mid \sigma \in X\}$ . Consider the augmented complex

$$\hat{K} := K \cup (u * \overline{\text{St } v}).$$

In other words, for every simplex  $\{u_0, \dots, u_d\} \in \overline{\text{St } v}$  of  $K$ , we add the simplex  $\{u_0, \dots, u_d\} \cup \{u\}$  to  $\hat{K}$  if it is not already in. See Figure 1. Notice that  $K'$  is a subcomplex of  $\hat{K}$  in this example which we observe is true in general.

CLAIM 3.1.  $K' \subseteq \hat{K}$ .

PROOF. For a simplex  $\sigma \in K'$  that does not contain  $u$ ,  $f$  is identity on its unique pre-image; that is,  $\sigma \in K \subseteq \hat{K}$ . Now consider a  $d$ -simplex  $\sigma = \{u, u_1, \dots, u_d\} \in K'$ . Since

$f$  is surjective, there exists at least one pre-image of  $\sigma$  in  $K$  of the form  $\sigma' = \{u_0, u_1, \dots, u_d\}$ , where  $u_0$  is either  $u$  or  $v$ . If it is  $u_0 = u$ , we have  $f(\sigma') = \sigma' = \sigma$  and thus  $\sigma \in K \subseteq \hat{K}$ . So, assume that  $u_0 = v$ . This means that the simplex  $\{u_1, \dots, u_d\}$  is in  $\text{Lk } v$  (and thus in  $\overline{\text{St } v}$ ), implying that  $\sigma = \{u, u_1, \dots, u_d\} \in \hat{K}$ .  $\square$

Now consider the canonical inclusions  $i : K \hookrightarrow \hat{K}$  and  $i' : K' \hookrightarrow \hat{K}$ . These inclusions constitute the diagram on the righthand side which does not necessarily commute. Nevertheless, it commutes at the homology level which is precisely stated below.

$$\begin{array}{ccc} K & \xrightarrow{f} & K' \\ & \searrow i & \downarrow i' \\ & & \hat{K} \end{array}$$

PROPOSITION 3.2.  $f_* : H_*(K) \rightarrow H_*(K')$  is equal to  $(i'_*)^{-1} \circ i_*$  where  $i'_*$  is an isomorphism and  $H_*(K) \xrightarrow{i_*} H_*(\hat{K}) \xleftarrow{i'_*} H_*(K')$ .

PROOF. We use the notion of contiguous maps which induces equal maps at the homology level. Two maps  $f_1 : K_1 \rightarrow K_2$ ,  $f_2 : K_1 \rightarrow K_2$  are contiguous if for every simplex  $\sigma \in K_1$ ,  $f_1(\sigma) \cup f_2(\sigma)$  is a simplex in  $K_2$ . We observe that the simplicial maps  $i' \circ f$  and  $i$  are contiguous and  $i'$  induces an isomorphism at the homology level, that is,  $i'_* : H_*(K) \rightarrow H_*(\hat{K})$  is an isomorphism.

Since  $i$  is contiguous to  $i' \circ f$  (Proposition A.1 [16]), we have  $i_* = (i' \circ f)_* = i'_* \circ f_*$ . Since  $i'_*$  is an isomorphism (Proposition A.2 [16]),  $(i'_*)^{-1}$  exists and is an isomorphism. It then follows that  $f_* = (i'_*)^{-1} \circ i_*$ .  $\square$

Proposition 3.2 allows us to simulate the persistence of a sequence of simplicial maps with only inclusion-induced homomorphisms. Consider the following sequence of simplicial complexes connected with a zigzag sequence of simplicial maps (the arrows could be oriented differently in general):

$$K_1 \xrightarrow{f_1} K_2 \xleftarrow{f_2} K_3 \xrightarrow{f_3} \cdots \rightarrow K_n$$

which generates the module at the homology level by induced homomorphisms  $f_{i*}$

$$\mathcal{F} : H_*(K_1) \xrightarrow{f_{1*}} H_*(K_2) \xleftarrow{f_{2*}} H_*(K_3) \xrightarrow{f_{3*}} \cdots \rightarrow H_*(K_n)$$

When the maps  $f_i$  are all inclusions, it is known that the zigzag persistence induced by them can be computed in matrix multiplication time by a recent algorithm of [24]. This algorithm does not extend to simplicial maps as per se though we know that a persistence module induced by simplicial maps admits a decomposition [26] and hence a persistence diagram [11]. With our observation that every simplicial map can be simulated with inclusion maps, we can take advantage of the algorithm of [24] for computing zigzag persistence for simplicial maps. In view of Proposition 3.2, consider the following sequence connected only with inclusions:

$$K_1 \hookrightarrow \hat{K}_1 \hookleftarrow K_2 \hookrightarrow \hat{K}_3 \hookleftarrow K_3 \hookrightarrow \cdots \hookleftarrow K_n$$

At the homology level we have  $H_*(\hat{K}_i) \simeq H_*(K_{i+1})$  induced by the inclusion  $\hat{K}_i \hookleftarrow K_{i+1}$  and also  $H_*(K_i) \simeq H_*(\hat{K}_{i+1})$

$$\begin{array}{ccccccccccccccc}
H_*(K_1) & \xrightarrow{f_{1*}} & H_*(K_2) & \xrightarrow{=} & H_*(K_2) & \xrightarrow{=} & H_*(K_2) & \xleftarrow{f_{2*}} & H_*(K_3) & \xrightarrow{f_{3*}} & \dots & \longrightarrow & H_*(K_m) \\
\downarrow = & & \downarrow \simeq & & \downarrow = & & \downarrow \simeq & & \downarrow = & & & & \downarrow = \\
H_*(K_1) & \xrightarrow{i_{1*}} & H_*(\hat{K}_1) & \xrightarrow{\simeq} & H_*(K_2) & \xrightarrow{\simeq} & H_*(\hat{K}_3) & \xleftarrow{i_{2*}} & H_*(K_3) & \xrightarrow{i_{3*}} & \dots & \longrightarrow & H_*(K_m)
\end{array}$$

induced by the inclusion  $K_i \hookrightarrow \hat{K}_{i+1}$ . Thus, we have the following persistence module:

$$\begin{aligned}
\mathcal{M} : H_*(K_1) & \xrightarrow{i_*} H_*(\hat{K}_1) \\
& \simeq H_*(K_2) \\
& \simeq H_*(\hat{K}_3) \xleftarrow{i_*} H_*(K_3) \xrightarrow{i_*} \dots \xleftarrow{i_*} H_*(K_n)
\end{aligned}$$

**THEOREM 3.3.** *The persistence diagram of  $\mathcal{F}$  can be derived from the that of the module  $\mathcal{M}$ .*

**PROOF.** Consider the diagram between vector spaces as shown above. All isomorphisms are induced by inclusions, hence every square being supported only by isomorphisms commutes. The other squares supported by  $f_{i*}$  also commute because of Proposition 3.2. Hence every square in this diagram commutes, and the claim follows [6, 19].  $\square$

## 4. ANNOTATIONS

When we are given a *non-zigzag* sequence of simplicial maps  $K_1 \xrightarrow{f_1} K_2 \xrightarrow{f_2} K_3 \dots \xrightarrow{f_n} K_n$  we can improve upon the coning approach by reducing simplex insertions as illustrated in Figure 2. Consider the map  $f_{ij} : K_i \rightarrow K_j$  where  $f_{ij} = f_{j-1} \circ \dots \circ f_{i+1} \circ f_i$ . To compute the persistent homology, the persistence algorithm essentially maintains a consistent basis by computing the image  $f_{ij*}(B_i)$  of a basis  $B_i$  of  $H_*(K_i)$ . As one moves through a map in the filtration, the homology basis elements get created (birth) or can be interpreted to be destroyed (death). The notion of this birth and death of the homology basis elements can be formulated precisely with algebra [26] and can be summarized with persistence diagrams [11]. Here, instead of a consistent homology basis, we maintain a consistent cohomology basis, that is, if  $B^i$  is a cohomology basis of  $H^*(K_i)$  maintained by the algorithm, we compute the preimage  $f_{ij}^{*-1}(B^j)$

where  $H^*(K_i) \xleftarrow{f_{ij}^*} H^*(K_j)$  is the homomorphism induced in the cohomology groups by  $f_{ij}$ . By duality, this implicitly maintains a consistent homology basis and thus captures all information about persistent homology as well [14].

Our main tool to maintain a consistent cohomology basis is the notion of annotation [4] which are binary vectors assigned to simplices. We maintain the annotations as we go forward through the given sequence, and thus maintain a cohomology basis in the reverse direction whose birth and death coincide with the death and birth respectively of a consistent homology basis.

**DEFINITION 4.1.** *Given a simplicial complex  $K$ , Let  $K(p)$  denote the set of  $p$ -simplices in  $K$ . An annotation for  $K(p)$  is an assignment  $\mathbf{a} : K(p) \rightarrow \mathbb{Z}_2^g$  of a binary vector  $\mathbf{a}_\sigma = \mathbf{a}(\sigma)$  of same length  $g$  for each  $p$ -simplex  $\sigma \in K$ . Entries of  $\mathbf{a}_\sigma$  are called its elements. We also have an induced annotation for any  $p$ -chain  $c_p$  given by  $\mathbf{a}_{c_p} = \sum_{\sigma \in c_p} \mathbf{a}_\sigma$ .*

**DEFINITION 4.2.** *An annotation  $\mathbf{a} : K(p) \rightarrow \mathbb{Z}_2^g$  is valid if conditions 1 and 2 are satisfied:*

1.  $g = \text{rank } H_p(K)$ , and
2. two  $p$ -cycles  $z_1$  and  $z_2$  have  $\mathbf{a}_{z_1} = \mathbf{a}_{z_2}$  iff their homology classes are identical, i.e.  $[z_1] = [z_2]$ .

**PROPOSITION 4.3.** *Statements 1 and 2 are equivalent:*

1. An annotation  $\mathbf{a} : K(p) \rightarrow \mathbb{Z}_2^g$  is valid
2. The cochains  $\{\phi_i\}_{i=1, \dots, g}$  given by  $\phi_i(\sigma) = \mathbf{a}_\sigma[i]$  for all  $\sigma \in K(p)$  are cocycles whose cohomology classes  $\{[\phi_i]\}$ ,  $i = 1, \dots, g$  constitute a basis of  $H^p(K)$ .

**PROOF.**  $1 \rightarrow 2$ : The cochains  $\phi_i$  are cocycles since for any  $(p+1)$ -simplex  $\tau \in K(p+1)$  one has  $[\partial\tau] = [0]$  and hence  $\delta_p \phi_i(\tau) = \phi_i(\partial\tau) = \phi_i(0) = 0$ , where  $\delta_p$  is the co-boundary operator for  $p$ -dimensional co-chains. Let  $[z_1], [z_2], \dots, [z_g]$  be a basis of  $H_p(K)$ . Let  $V$  be the vector space generated by  $[\phi_i]$ ,  $i = 1, \dots, g$ . Define a bilinear form  $\alpha : V \times H_p(K) \rightarrow \mathbb{Z}_2$  by  $\alpha([\phi_i], [z_j]) = \phi_i(z_j)$ . The matrix  $[\phi_i(z_j)]_{ij}$  has full rank due to the condition 2 in the definition of annotation. This means the vector spaces  $V$  and  $H_p(K)$  have the same rank and hence are isomorphic. It follows that  $V \simeq H^p(K)$ .

$2 \rightarrow 1$ : For this direction, consider a basis  $[z_1], [z_2], \dots, [z_g]$  of  $H_p(K)$ . By universal coefficient theorem we have an isomorphism  $H^p(K) \simeq \text{Hom}(H_p(K), \mathbb{Z}_2)$  which sends a cocycle class  $[\phi_i]$  to the homomorphism  $[z_j] \mapsto \phi_i(z_j)$ . This means that the matrix  $[\phi_i(z_j)]_{ij}$  has full rank and hence the vectors  $[\phi_1(z_j), \dots, \phi_g(z_j)]$  and  $[\phi_1(z_k), \dots, \phi_g(z_k)]$  are identical if and only if  $[z_i] = [z_k]$ . The claim can be extended to any homology class since it can be expressed as a linear combination of the basis elements.  $\square$

In light of the above result, an annotation is simply one way to represent a cohomology basis. However, by representing the corresponding basis as an explicit vector associated with each simplex, it localizes the basis to each simplex. As a result, we can update the cohomology basis locally by changing the annotations locally (see Proposition 5.4). This point of view also helps to reveal how we can process elementary collapses, which are neither inclusions nor deletions, by transferring annotations (see Proposition 5.5).

## 5. ALGORITHM

Consider the persistence module  $\mathcal{M}$  induced by elementary simplicial maps  $f_i : K_i \rightarrow K_{i+1}$ .

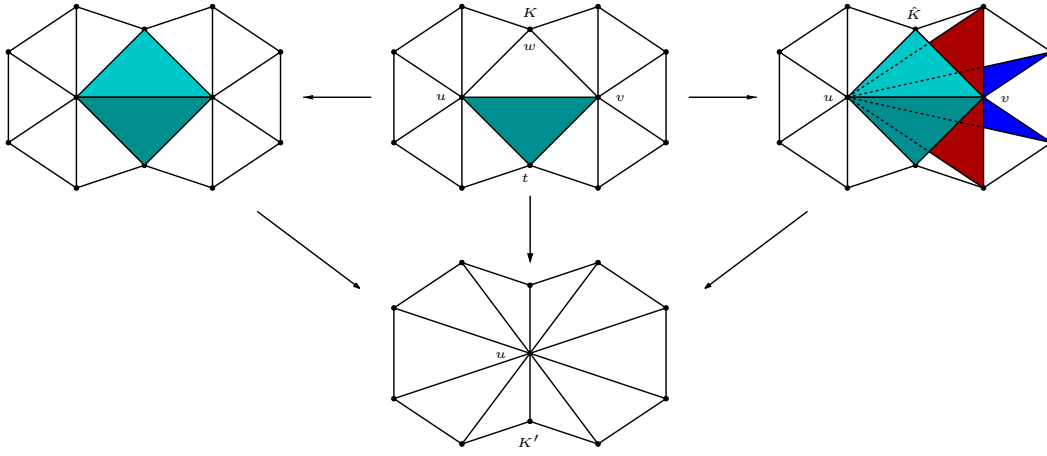
$$\mathcal{M} : H_*(K_1) \xrightarrow{f_{1*}} H_*(K_2) \xrightarrow{f_{2*}} H_*(K_3) \dots \xrightarrow{f_{n*}} H_*(K_n)$$

Instead of tracking a consistent homology basis for the module  $\mathcal{M}$ , we track a cohomology basis in the dual module  $\mathcal{M}^*$  where the homomorphisms are in reverse direction:

$$\mathcal{M}^* : H^*(K_1) \xleftarrow{f_1^*} H^*(K_2) \xleftarrow{f_2^*} H^*(K_3) \dots \xleftarrow{f_n^*} H^*(K_n)$$

As we move from left to right in the above sequence, the annotations implicitly maintain a cohomology basis whose





**Figure 2: Annotation vs. coning:** The pair  $(u, v)$  is collapsed to  $u$  in  $K$  to produce  $K'$  (middle column). The 2-simplices are the shaded triangles alone. Annotation requires inserting (implicitly) the single triangle as shown on the left whereas coning requires inserting many more simplices as shown by shaded triangles on the right. Specifically, the coning approach requires inserting all simplices in the cone  $u * \overline{\text{St } v}$  formed by  $u$  and all simplices in the closure of the star of  $v$ .

elements are also *time stamped* to signify when a basis element is born or dies. We should keep in mind that the *birth* and *death* of a cohomology basis element coincides with the *death* and *birth* of a homology basis element because the two modules run in opposite directions.

## 5.1 Elementary inclusion

The handling of elementary inclusions using annotations can be viewed as an alternative formulation of the algorithm proposed in [14]; see also [13]. We describe it in terms of the annotation here because it is also used in an elementary collapse, a new atomic operation that we need to address. Consider an elementary inclusion  $K_i \hookrightarrow K_{i+1}$ . Assume that  $K_i$  has a valid annotation. We describe how we obtain a valid annotation for  $K_{i+1}$  from that of  $K_i$  after inserting the  $p$ -simplex  $\sigma = K_{i+1} \setminus K_i$ . We compute the annotation  $\mathbf{a}_{\partial\sigma}$  for the boundary  $\partial\sigma$  in  $K_i$  and take actions as follows. A formal justification is provided in the extended version [16].

Case (i): If  $\mathbf{a}_{\partial\sigma}$  is a zero vector, the class  $[\partial\sigma]$  is trivial in  $H_{p-1}(K_i)$ . This means  $\sigma$  creates a  $p$ -cycle in  $K_{i+1}$  and by duality a  $p$ -cocycle is killed while going left from  $K_{i+1}$  to  $K_i$ . In this case we augment the annotations for all  $p$ -simplices by one element with a time stamp  $i + 1$ , that is, an annotation  $[b_1, b_2, \dots, b_g]$  for a  $p$ -simplex  $\tau$  is updated to  $[b_1, b_2, \dots, b_g, b_{g+1}]$  with the last element time stamped  $i + 1$  where  $b_{g+1} = 0$  for  $\tau \neq \sigma$  and  $b_{g+1} = 1$  for  $\tau = \sigma$ . The element  $b_i$  of  $\mathbf{a}_\sigma$  is set to zero for  $1 \leq i \leq g$ . Other annotations for other simplices remain unchanged. See Figure 3(a).

Case (ii): If  $\mathbf{a}_{\partial\sigma}$  is not a zero vector, the class of the  $(p - 1)$ -cycle  $\partial\sigma$  is nontrivial in  $H_{p-1}(K_i)$ . Therefore,  $\sigma$  kills the class of this cycle and a corresponding dual class of cocycles is born in the reverse direction. We simulate it by forcing  $\mathbf{a}_{\partial\sigma}$  to be zero which affects other annotations as well. Let  $i_1, i_2, \dots, i_k = u$  be the set of indices in non-decreasing order so that  $b_{i_1}, b_{i_2}, \dots, b_{i_k} = b_u$  are all of the nonzero elements in  $\mathbf{a}_{\partial\sigma} = [b_1, b_2, \dots, b_u, \dots, b_g]$ . The cocycle  $\phi = \phi_{i_1} + \phi_{i_2} + \dots + \phi_{i_k} = \phi_u$  should become a coboundary after the

addition of  $\sigma$ , which renders

$$\phi_u = \phi_{i_1} + \phi_{i_2} + \dots + \phi_{i_{k-1}}.$$

We make the latest cocycle  $\phi_u$  to be dependent on others. In other words, the cocycle class  $[\phi]$  which is born at the time  $i + 1$  is chosen to be killed at time when  $b_u$  was introduced. This pairing matches that of the standard persistence algorithm where the youngest basis element is always paired first. We add the vector  $\mathbf{a}_{\partial\sigma}$  to all annotations of  $(p - 1)$ -simplices whose  $u$ th element is nonzero. This zeroes out the  $u$ th element of all annotations of  $(p - 1)$ -simplices. We simply delete this element from all such annotations. See Figure 3(b).

Notice that determining if we have case (i) or (ii) can be done easily in  $O(pg)$  time by checking the annotation of  $\partial\sigma$ . Indeed, this is achieved because the annotation already localizes the co-homology basis to each individual simplex.

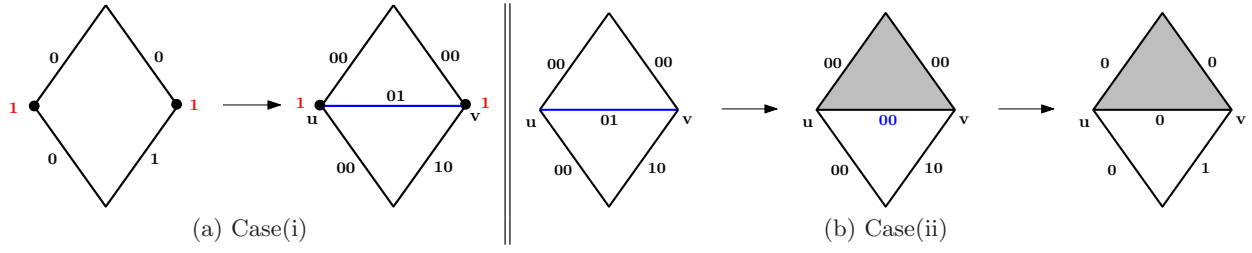
## 5.2 Elementary collapse

The case for handling collapse is more interesting. It has three distinct steps, (i) elementary inclusions to satisfy the so called link condition, (ii) local annotation transfer to prepare for the collapse, and (iii) collapse of the simplices with updated annotations. We explain each of these steps now.

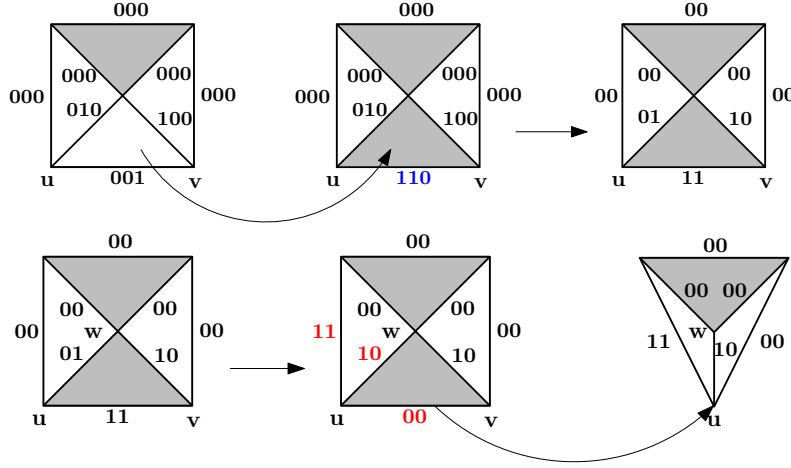
The elementary inclusions that may precede the final collapse are motivated by a result that connects collapses with the change in (co)homology. Consider an elementary collapse  $K_i \xrightarrow{f_i} K_{i+1}$  where the vertex pair  $(u, v)$  collapses to  $u$ . The following link condition, introduced in [15] and later used to preserve homotopy [1], becomes relevant.

**DEFINITION 5.1.** *A vertex pair  $(u, v)$  in a simplicial complex  $K_i$  satisfies the link condition if the edge  $\{u, v\} \in K_i$ , and  $\text{Lk } u \cap \text{Lk } v = \text{Lk } \{u, v\}$ . An elementary collapse  $f_i : K_i \rightarrow K_{i+1}$  satisfies the link condition if the vertex pair on which  $f_i$  is not injective satisfies the link condition.*

**PROPOSITION 5.2.** [1] *If an elementary collapse  $f_i : K_i \rightarrow K_{i+1}$  satisfies the link condition, then the underlying spaces  $|K_i|$  and  $|K_{i+1}|$  remain homotopy equivalent and hence the*

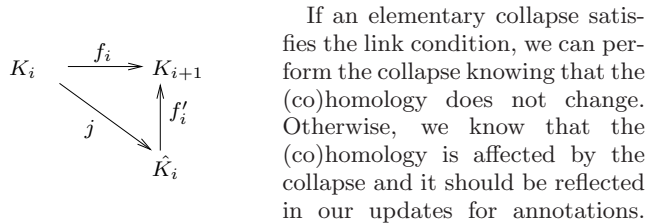


**Figure 3: Case(i) of inclusion:** the boundary  $\partial uv = u + v$  of the edge  $uv$  has annotation  $1 + 1 = 0$ . After its addition, every edge gains an element in its annotation which is 0 for all except the edge  $uv$ . **Case (ii) of inclusion:** the boundary of the top triangle has annotation 01. It is added to the annotation of  $uv$  which is the only edge having the second element 1. Consequently the second element is zeroed out for every edge, and is deleted.



**Figure 4: Annotation updates for elementary collapse:** inclusion of a triangle to satisfy the link condition (upper row), annotation transfer and actual collapse (lower row); annotation 11 of the vanishing edge  $uv$  is added to all edges (cofaces) adjoining  $u$ .

induced homomorphisms  $f_{i*} : H_*(K_i) \rightarrow H_*(K_{i+1})$  and  $f_i^* : H^*(K_i) \leftarrow H^*(K_{i+1})$  are isomorphisms.



The diagram at the left provides a precise means to carry out the change in (co)homology. Let  $S$  be the set of simplices in non-decreasing order of dimensions, whose absence from  $K_i$  makes  $(u, v)$  violate the link condition. For each such simplex  $\sigma \in S$ , we modify the annotations of every simplex which we would have done if  $\sigma$  were to be inserted. Thereafter, we carry out the rest of the elementary collapse. In essence, implicitly, we obtain an intermediate complex  $\hat{K}_i = K_i \cup S$  where the diagram on the left commutes. Here,  $f'_i$  is induced by the same vertex map that induces  $f_i$ , and  $j$  is an inclusion. This means that the persistence of  $f_i$  is identical to that of  $f'_i \circ j$  which justifies our action of elementary inclusions followed by the actual collapses.

We remark that this is the only place where we may insert implicitly a simplex  $\sigma$  in the current approach. The number of such  $\sigma$  is usually much smaller than the number of simplices in the cone  $u * \overline{\text{St } v}$  that we would need to insert for the algorithm using coning.

Next, we transfer annotations in  $\hat{K}_i$ . This step locally changes the annotations for simplices containing the vertices  $u$  and/or  $v$ . The following definition facilitates the description.

**DEFINITION 5.3.** For an elementary collapse  $f_i : K_i \rightarrow K_{i+1}$ , a simplex  $\sigma \in K_i$  is called vanishing if the cardinality of  $f_i(\sigma)$  is one less than that of  $\sigma$ . Two simplices  $\sigma$  and  $\sigma'$  are called mirror pairs if one contains  $u$  and the other  $v$ , and share rest of the vertices. In Figure 4(lower row), the vanishing simplices are  $\{\{u, v\}, \{u, v, w\}\}$  and the mirror pairs are  $\{\{u\}, \{v\}\}, \{\{u, w\}, \{v, w\}\}$ .

In an elementary collapse that sends  $(u, v)$  to  $u$ , all vanishing simplices need to be deleted, and all simplices containing  $v$  need to be pulled to the vertex  $u$  (which are their mirror partners). We update the annotations in such a way that the annotations of all vanishing simplices become zero, and those of each pair of mirror simplices become the same. Once this is achieved, the collapse is implemented by simply

deleting the vanishing simplices and replacing  $v$  with  $u$  in all simplices containing  $v$  without changing their annotations. The following proposition provides the justification behind the specific update operators that we perform.

**PROPOSITION 5.4.** *Let  $K$  be a simplicial complex and  $\mathbf{a} : K(p) \rightarrow \mathbb{Z}_2^g$  be a valid annotation. Let  $\sigma \in K(p)$  be any  $p$ -simplex and  $\tau$  any of its  $(p-1)$ -faces. Adding  $\mathbf{a}_\sigma$  to the annotation of all cofaces of  $\tau$  of codimension 1 produces a valid annotation for  $K(p)$ . Furthermore, the cohomology basis corresponding to the annotations remains unchanged by this modification.*

**PROOF.** Let  $\{[\phi_1], \dots, [\phi_g]\}$  be a cohomology basis of  $H^p(K)$  corresponding to  $\mathbf{a} : K(p) \rightarrow \mathbb{Z}_2^g$  as stated in Proposition 4.3. Let  $T$  be the set of cofaces of  $\tau$  of codimension 1 and

$$\phi'_i(\sigma') = \begin{cases} \phi_i(\sigma') & \text{if } \sigma' \in K(p) \setminus T \\ \phi_i(\sigma') + \phi_i(\sigma) & \text{if } \sigma' \in T \end{cases}$$

By construction,  $\phi'_i$  is the cochain that corresponds to the new annotation obtained by adding  $\mathbf{a}_\sigma$  to that of the simplices in  $T$ . We prove that  $\phi'_i$  is a cocycle in the class  $[\phi_i]$ . Therefore,  $\{[\phi'_1], \dots, [\phi'_g]\}$  is a cohomology basis of  $H^p(K)$ . The new annotation is valid by Proposition 4.3 and the cohomology bases remain unchanged.

If  $\phi_i(\sigma) = 0$ , we have  $\phi'_i = \phi_i$  and thus  $[\phi'_i] = [\phi_i]$  trivially. So, assume that  $\phi_i(\sigma) = 1$ . In this case  $\phi'_i = 1 + \phi_i$  on  $T$  and equals  $\phi_i$  everywhere else. Consider the  $(p-1)$ -cochain  $\phi$  defined by  $\phi(\tau) = 1$  and  $\phi(\tau') = 0$  for every  $\tau' \in K_{p-1} \setminus \tau$ . Then the coboundary  $\delta\phi$  is a  $p$ -cochain that is 1 for every simplex in  $T$  and 0 on other  $p$ -simplices. We can write  $\phi'_i = \phi_i + \delta\phi$ . It follows that  $[\phi'_i] = [\phi_i]$ .  $\square$

Consider an elementary collapse  $f_i : K_i \rightarrow K_{i+1}$  that sends  $(u, v)$  to  $u$ . We update the annotations in  $K_i$  as follows. First, note that the vanishing simplices are exactly those simplices containing the edge  $\{u, v\}$ . For every  $p$ -simplex containing  $\{u, v\}$ , i.e., a vanishing simplex, exactly two among its  $(p-1)$ -faces are mirror simplices, and all other remaining  $(p-1)$ -faces are vanishing simplices. Let  $\sigma$  be a vanishing  $p$ -simplex and  $\tau$  be its  $(p-1)$ -face that is a mirror simplex containing  $u$ . We add  $\mathbf{a}_\sigma$  to the annotations of all cofaces of  $\tau$  of codimension 1 including  $\sigma$ . We call this an *annotation transfer* for  $\sigma$ . By Proposition 5.4, the new annotation generated by this process corresponds to the old cohomology basis for  $K_i$ . This new annotation has  $\mathbf{a}_\sigma$  as zero since  $\mathbf{a}_\sigma + \mathbf{a}_\sigma = 0$ . See the lower row of Figure 4.

We perform the above operation for each vanishing simplex. It turns out that by using the relations of vanishing simplices and mirror simplices, each mirror simplex eventually acquires an identical annotation to that of its partner. Specifically, we have the following observation.

**PROPOSITION 5.5.** *After all possible annotation transfers involved in a collapse, (i) each vanishing simplex has a zero annotation; and (ii) each mirror simplex  $\tau$  has the same annotation as its mirror partner simplex  $\tau'$ .*

**PROOF.** Our algorithm performs an annotation transfer for every vanishing simplex. Furthermore, the annotation transfer for a vanishing simplex  $\sigma$  does not affect the annotation of any other vanishing simplex. Hence, the annotation of each vanishing simplex  $\sigma$  is updated exactly once after which it becomes zero and remains so throughout the rest of the annotation transfers for other vanishing simplices. This proves claim (i).

For claim (ii), consider a pair of  $(p-1)$ -dimensional mirror simplices  $\tau = \{u, u_2, \dots, u_p\}$  and  $\tau' = \{v, u_2, \dots, u_p\}$ . Since  $(u, v)$  satisfies the link condition, it is necessary that the  $p$ -simplex  $\alpha = \{u, v, u_2, \dots, u_p\}$  must exist in  $\hat{K}_i$ . Thus, we have that  $\mathbf{a}_{\partial\alpha} = 0$ . On the other hand, other than  $\tau$  and  $\tau'$ , any  $(p-1)$ -face of  $\alpha$  is a vanishing simplex, and by Claim (i), in the end, has zero annotation. Therefore, after all annotation transfers,  $\mathbf{a}_{\partial\alpha} = \mathbf{a}_\tau + \mathbf{a}_{\tau'} = 0$ , implying that  $\mathbf{a}_\tau = \mathbf{a}_{\tau'}$ .  $\square$

Subsequent to the annotation transfer, the annotation of  $\hat{K}_i$  fits for actual collapse since each pair of mirror simplices which are collapsed to a single simplex get the identical annotation and the vanishing simplex acquires the zero annotation. Furthermore, Proposition 5.4 tells us that the cohomology basis does not change by annotation transfer which aligns with the fact that  $f_i^* : H^*(\hat{K}_i) \leftarrow H^*(K_{i+1})$  is indeed an isomorphism. Accordingly, no time stamp changes after the annotation transfer and the actual collapse. Propositions 5.2 and 5.3 in [16] provide formal statements justifying the algorithm for annotation updates.

## 6. APPLICATION TO TOPOLOGICAL DATA ANALYSIS

In topological data analysis, several applications and approaches use Rips complex filtration [1, 18, 25]. The computation of the persistence diagram or its approximation for a Rips filtration appears to be a key step in these applications. However, the size of this filtration becomes a bottleneck because of the inclusive nature of Rips complexes. A natural way to handle this problem is to successively subsample the input data and build a filtration on top of them. We show how one can apply our results from previous sections to approximate the persistence diagrams of a Rips filtration from such a sparser filtration. Given a set of points  $V \subset \mathbb{R}^d$  (Similar to [25], results in this section can be extended to any metric space with doubling dimension  $d$ ), let  $\mathcal{R}^r(V)$  denote the Rips complex on the point set  $V$  with parameter  $r$ . That is, a  $k$ -simplex  $\sigma = \{u_0, \dots, u_k\} \subseteq V$  is in  $\mathcal{R}^r(V)$  if and only if  $\|u_i - u_j\| \leq r$  for any  $i, j \in [0, k]$ . We present an algorithm to approximate the persistence diagram for the following Rips filtration. The parameters  $\alpha > 0$  and  $0 \leq \varepsilon \leq 1$  are assumed to be preselected.

$$\mathcal{R}^\alpha(V) \hookrightarrow \mathcal{R}^{(1+\varepsilon)\alpha}(V) \dots \hookrightarrow \mathcal{R}^{(1+\varepsilon)^m\alpha}(V). \quad (1)$$

The number of  $k$ -simplices in a Rips complex with  $n$  vertices can be  $\Theta(n^{k+1})$ . This makes computing the persistent homology of the above filtration costly. In [25], Sheehy proposed to approximate the persistence diagram of the above filtration by another Rips filtration where each simplicial complex involved has size only linear in  $n$ . This approach allows vertices to be collapsed (deleted) with a weighting scheme when the parameter  $r$  for the Rips complex becomes large, which helps to keep the size of the simplicial complex at each stage small.

In this section, we provide an alternative approach to approximate the persistence diagram of the filtration given in (1). We achieve sparsification by subsampling as in [25], but our persistence algorithm for simplicial maps allows us to handle the sequence of complexes induced by the clustering / collapsing of vertices directly instead of an additional weighting scheme. We consider two sparsification schemes,

one produces a sequence of sparsified Rips complexes, and the other produces a sequence of graph induced complexes (GICs) which have been shown to be even sparser in practice [17]. Asymptotically, both sequences have sizes linear in number of vertices.

**Persistence diagram approximation by sparsified Rips complex.** Given a set of points  $V$ , we say that  $V' \subseteq V$  is a  $\delta$ -net of  $V$  if (i) for any point  $v \in V$ , there exists a point  $v' \in V'$  such that  $\|v - v'\| \leq \delta$ ; and (ii) no two points in  $V'$  are within  $\delta$  distance. A  $\delta$ -net  $V'$  can be easily constructed by a standard greedy approach by taking furthest points iteratively or by more sophisticated and efficient methods as in [10, 22].

Now set  $V_0 := V$ . We first construct a sequence of point sets  $V_k, k = 0, 1, \dots, m$ , such that  $V_{k+1}$  is a  $\frac{\alpha\varepsilon^2}{2}(1+\varepsilon)^{k-1}$ -net of  $V_k$ . Consider the following vertex maps  $\pi_k : V_k \rightarrow V_{k+1}$ , for  $k \in [0, m-1]$ , where for any  $v \in V_k$ ,  $\pi_k(v)$  is the vertex in  $V_{k+1}$  that is closest to  $v$ . Define  $\hat{\pi}_k : V_0 \rightarrow V_{k+1}$  as  $\hat{\pi}_k(v) = \pi_k \circ \dots \circ \pi_0(v)$ .

Each vertex map  $\pi_k$  induces a well-defined simplicial map  $h_k : \mathcal{R}^{\alpha(1+\varepsilon)^k}(V_k) \rightarrow \mathcal{R}^{\alpha(1+\varepsilon)^{k+1}}(V_{k+1})$ . Indeed, since  $V_{k+1}$  is a  $\frac{1}{2}\alpha\varepsilon^2(1+\varepsilon)^{k-1}$ -net of  $V_k$ , for each edge  $e = \{u, v\}$  from  $\mathcal{R}^{\alpha(1+\varepsilon)^k}(V_k)$ , we have

$$\begin{aligned} \|\pi_k(u) - \pi_k(v)\| &\leq \|u - v\| + \|u - \pi_k(u)\| + \|v - \pi_k(v)\| \\ &\leq \alpha(1+\varepsilon)^k + \alpha\varepsilon^2(1+\varepsilon)^{k-1} \\ &\leq \alpha(1+\varepsilon)^{k+1}. \end{aligned}$$

Hence  $\pi_k(u)\pi_k(v)$  is an edge in  $\mathcal{R}^{\alpha(1+\varepsilon)^{k+1}}(V_{k+1})$ . Since in a Rips complex, higher dimensional simplices are determined by the edges, every simplex  $\{u_0, \dots, u_d\}$  in  $\mathcal{R}^{\alpha(1+\varepsilon)^k}(V_k)$  has a well-defined image  $\{\pi_k(u_0), \dots, \pi_k(u_d)\}$  in  $\mathcal{R}^{\alpha(1+\varepsilon)^{k+1}}(V_{k+1})$ . Hence, each  $h_k$  is well-defined providing the filtration:

$$\mathcal{R}^\alpha(V_0) \xrightarrow{h_0} \mathcal{R}^{\alpha(1+\varepsilon)}(V_1) \dots \xrightarrow{h_{m-1}} \mathcal{R}^{\alpha(1+\varepsilon)^m}(V_m). \quad (2)$$

In other words, as the parameter  $r = \alpha(1+\varepsilon)^k$  increases, we can simply consider the Rips complex built upon the sparsified data points  $V_k$ . Note that the sequence above is not connected by inclusion maps and thus classical persistent algorithms cannot be applied directly; while our algorithm from Section 5 can be used here in a straightforward manner.

Our main observation is that the persistence diagram of the sequence of simplicial maps in (2) approximates that of the inclusion maps in (1). In particular, we show that the persistence modules induced by these two sequences interleave in the sense described in [8].

First, we need maps to connect these two sequences. For this, we observe that the vertex map  $\hat{\pi}_k : V_0 \rightarrow V_{k+1}$  also induces a simplicial map  $\hat{h}_k : \mathcal{R}^{\alpha(1+\varepsilon)^k}(V_0) \rightarrow \mathcal{R}^{\alpha(1+\varepsilon)^{k+1}}(V_{k+1})$ . To establish that this simplicial map is well-defined, it can be shown that if there is an edge  $\{u, v\}$  in  $\mathcal{R}^{\alpha(1+\varepsilon)^k}(V_0)$ , then there is an edge  $\hat{\pi}_k(u)\hat{\pi}_k(v)$  in  $\mathcal{R}^{\alpha(1+\varepsilon)^{k+1}}(V_{k+1})$ .

**CLAIM 6.1.** *Each triangle in the following diagram commutes at the homology level.*

$$\begin{array}{ccc} \mathcal{R}^{\alpha(1+\varepsilon)^k}(V_0) & \xrightarrow{i_k} & \mathcal{R}^{\alpha(1+\varepsilon)^{k+1}}(V_0) \\ \uparrow j_k & \searrow \hat{h}_k & \uparrow j_{k+1} \\ \mathcal{R}^{\alpha(1+\varepsilon)^k}(V_k) & \xrightarrow{h_k} & \mathcal{R}^{\alpha(1+\varepsilon)^{k+1}}(V_{k+1}) \end{array}$$

Here, the maps  $i_k$ s and  $j_k$ s are canonical inclusions. The simplicial maps  $\hat{h}_k$  and  $h_k$  are induced by the vertex maps  $\hat{\pi}_k : V_0 \rightarrow V_{k+1}$  and  $\pi_k : V_k \rightarrow V_{k+1}$ , respectively, as described before.

**PROOF.** First, we consider the bottom triangle. Note that the vertex map  $\hat{\pi}_k$  restricted on the set of vertices  $V_k$  is the same as the vertex map  $\pi_k$ . (That is, for a vertex  $u \in V_k \subseteq V_0$ ,  $h_k(u) = \hat{h}_k(u)$ .) Thus  $h_k = \hat{h}_k \circ j_k$ . Hence the bottom triangle commutes both at the simplicial complex level and at the homology level.

Consider the top triangle. We claim that the map  $j_{k+1} \circ \hat{h}_k$  is contiguous to the inclusion  $i_k$ . Since two contiguous maps induce the same homomorphisms at the homology level, the top triangle commutes at the homology level.

This claim can be verified by the definition of contiguous maps. Given a simplex  $\sigma \in \mathcal{R}^{\alpha(1+\varepsilon)^k}(V_0)$ , we wish to show that vertices from  $\sigma \cup \hat{h}_k(\sigma)$  span a simplex in  $\mathcal{R}^{\alpha(1+\varepsilon)^{k+1}}(V_0)$ . Since  $\mathcal{R}^{\alpha(1+\varepsilon)^{k+1}}(V_0)$  is a Rips complex, we only need to show that for any two vertices  $u$  and  $v$  from  $\sigma \cup \hat{h}_k(\sigma)$ , the edge  $uv$  has length less than  $\alpha(1+\varepsilon)^{k+1}$  (and thus in  $\mathcal{R}^{\alpha(1+\varepsilon)^{k+1}}(V_0)$ ). If  $u$  and  $v$  are both from  $\sigma$  or both from  $\hat{h}_k(\sigma)$ , then obviously  $\|u - v\| \leq \alpha(1+\varepsilon)^{k+1}$ . Otherwise, assume without loss of generality that  $v \in \sigma$  and  $u \in \hat{h}_k(\sigma)$  where  $u = \hat{\pi}_k(\bar{u})$  for some  $\bar{u} \in \sigma$ . It then follows that,

$$\begin{aligned} \|u - v\| &\leq \|u - \bar{u}\| + \|\bar{u} - v\| \\ &\leq \frac{\alpha\varepsilon^2}{2(1+\varepsilon)} \sum_{i=0}^k (1+\varepsilon)^i + \alpha(1+\varepsilon)^k < \alpha(1+\varepsilon)^{k+1}. \end{aligned}$$

Therefore, the vertices of  $\sigma \cup \hat{h}_k(\sigma)$  span a simplex in  $\mathcal{R}^{\alpha(1+\varepsilon)^{k+1}}(V_0)$ .  $\square$

The above claim implies that the persistence modules induced by sequences (1) and (2) are weakly  $\frac{\log(1+\varepsilon)}{2}$ -interleaved at the log-scale. By Theorem 4.3 of [8], we thus conclude with the following:

**PROPOSITION 6.2.** *The persistence diagram of the sequence (2) provides a  $\frac{3\log(1+\varepsilon)}{2}$ -approximation of the persistence diagram of the sequence (1) at the log-scale.*

Finally, since  $V_{k+1}$  is a  $\delta$ -net of  $V_k$  for  $\delta = \frac{\alpha\varepsilon^2}{2}(1+\varepsilon)^{k-1}$ , we can show by a standard packing argument that each  $\mathcal{R}^{\alpha(1+\varepsilon)^k}(V_k)$  is of size linear in  $n$ . See Proposition C.1 in [16]. Note that the persistence diagram of the simplicial maps in (2) can be computed by our algorithm in Section 5. Putting everything together, we have the following result.

**THEOREM 6.3.** *Given a set of  $n$  points  $V$  in a metric space with doubling-dimension  $d$  and  $0 \leq \varepsilon \leq 1$ , we can  $\frac{3\log(1+\varepsilon)}{2}$ -approximate the persistence diagram of the Rips complex filtration (1) by that of the filtration (2). The p-skeleton of each simplicial complex involved in (2) has size  $O((\frac{1}{\varepsilon})^{O(dp)}n)$ .*

**Persistence diagram approximation by graph induced complex.** We now present an alternative way to construct a sequence of complexes for gradually sparsified or subsampled points. The *graph induced complex* (GIC) proposed in [17] works on a subsample as the sparsified Rips complex does. However, it contains much fewer simplices



in practice. In [17], it was shown how GIC can be used to estimate the homology of compact sets by investigating the persistence of a single simplicial map. Here we show how one can build a sequence of GICs to approximate the persistence diagram of a Rips filtration. Similar to the case of a sequence of sparsified Rips complexes, simplicial maps occur naturally to connect these GICs in the sequence.

**DEFINITION 6.4.** Let  $G(V)$  be a graph with the vertex set  $V$  and let  $\nu : V \rightarrow V'$  be a vertex map where  $\nu(V) = V' \subseteq V$  is a subset of vertices. The graph induced complex  $\mathcal{G}(V, V', \nu) := \mathcal{G}(G(V), V', \nu)$  is defined as the simplicial complex where a  $k$ -simplex  $\sigma = \{v'_1, v'_2, \dots, v'_{k+1}\}$  is in  $\mathcal{G}(V, V', \nu)$  if and only if there exists a  $(k+1)$ -clique  $\{v_1, v_2, \dots, v_{k+1}\} \subseteq V$  so that  $\nu(v_i) = v'_i$  for each  $i \in \{1, 2, \dots, k+1\}$ . To see that it is indeed a simplicial complex, observe that a subset of a clique is also a clique. Let  $G(V)$  be called the base-graph for  $\mathcal{G}(V, V', \nu)$ .

Intuitively, the vertex map  $\nu$  maps a cluster of vertices from  $V$  to a single vertex  $v' \in V'$ , and these vertices constitute the “Voronoi cell” of the site  $v'$ . The GIC  $\mathcal{G}(V, V', \nu)$  is somewhat the combinatorial dual of such a Voronoi diagram. In our case the base graph  $G(V)$  is the 1-skeleton of the Rips complex  $\mathcal{R}^r(V_0)$  and the vertex map  $\nu$  is the map  $\hat{\pi}_k : V_0 \rightarrow V_{k+1}$  as defined in the last section. Denote  $\mathcal{G}^r(V_0, V_k) := \mathcal{G}(V_0, V_k, \hat{\pi}_{k-1})$  constructed using the 1-skeleton of  $\mathcal{R}^r(V_0)$  as the base-graph. It is easy to show that by the definition of  $\hat{\pi}_k$  and construction of  $V_k$ s, the vertex map  $\pi_k : V_k \rightarrow V_{k+1}$  induces a well-defined simplicial map  $f_k : \mathcal{G}^{\alpha(1+\varepsilon)^{k-1}}(V_0, V_k) \rightarrow \mathcal{G}^{\alpha(1+\varepsilon)^k}(V_0, V_{k+1})$ , giving rise to the following sequence:

$$\mathcal{G}^{\alpha}(V_0, V_1) \xrightarrow{f_1} \mathcal{G}^{\alpha(1+\varepsilon)}(V_0, V_2) \xrightarrow{f_2} \mathcal{G}^{\alpha(1+\varepsilon)^2}(V_0, V_3) \dots \xrightarrow{f_{m-1}} \mathcal{G}^{\alpha(1+\varepsilon)^{m-1}}(V_0, V_m). \quad (3)$$

We prove that the persistence diagram of the above filtration induced by simplicial maps  $f_k$ 's has the same approximation factor to the persistence of diagram of the filtration (1) as that of the filtration (2). Thus, we have:

**THEOREM 6.5.** Given a set of  $n$  points  $V$  in a metric space with doubling-dimension  $d$  and  $0 \leq \varepsilon \leq 1$ , we can  $\frac{3 \log(1+\varepsilon)}{2}$ -approximate the persistence diagram of the Rips complex filtration (1) by that of the filtration (3). The  $p$ -skeleton of each simplicial complex involved in (3) has size  $O((\frac{1}{\varepsilon})^{O(dp)} n)$ .

**PROOF.** In sequence (3),  $V_{k+1}$  is a  $\delta_{k+1}$ -net of  $V_k$  for  $\delta_{k+1} = \frac{\alpha\varepsilon^2}{2}(1+\varepsilon)^{k-1}$  ( $k = 0, 1, \dots, m-1$ ) as in the sequence of (2). Now consider  $\hat{\pi}_k : V_0 \rightarrow V_{k+1}$ . It is immediate that  $|p\hat{\pi}_k(p)| \leq \frac{\alpha\varepsilon^2}{2(1+\varepsilon)} \sum_{i=0}^k (1+\varepsilon)^i \leq \frac{\alpha\varepsilon}{2}(1+\varepsilon)^k$  for each  $p \in V_0$ . In other words,  $V_{k+1}$  is a  $\frac{\alpha\varepsilon}{2}(1+\varepsilon)^k$ -sample of  $V_0$ . Recall the GIC  $\mathcal{G}^{\alpha(1+\varepsilon)^k}(V_0, V_{k+1})$  is constructed based on the 1-skeleton of  $\mathcal{R}^{\alpha(1+\varepsilon)^k}(V_0)$  (used as the base-graph). It is easy to show that  $\hat{\pi}_k(p)$  induces a simplicial map  $\hat{f}_k : \mathcal{R}^{\alpha(1+\varepsilon)^k}(V_0) \rightarrow \mathcal{G}^{\alpha(1+\varepsilon)^k}(V_0, V_{k+1})$ . To prove that the persistence diagram of the sequence (1) is approximated by that of the sequence (3), it is sufficient to show that the sequence (1) interleaves with the sequence (3). The following claim reveals the desired interleaving property. Its proof is similar to that of the Claim 6.1.

**CLAIM 6.6.** Each triangle in the following diagram commutes at the homology level.

$$\begin{array}{ccc} \mathcal{R}^{\alpha(1+\varepsilon)^k}(V_0) & \xhookrightarrow{i_k} & \mathcal{R}^{\alpha(1+\varepsilon)^{k+1}}(V_0) \\ & \searrow \hat{f}_k & \uparrow j_{k+1} \\ \mathcal{G}^{\alpha(1+\varepsilon)^{k-1}}(V_0, V_k) & \xrightarrow{f_k} & \mathcal{G}^{\alpha(1+\varepsilon)^k}(V_0, V_{k+1}) \end{array}$$

Here, the maps  $i_k$ s and  $j_k$ s are canonical inclusions. The simplicial map  $\hat{f}_k$  is induced by the vertex map  $\hat{\pi}_k : V_0 \rightarrow V_{k+1}$ , and the simplicial map  $f_k = \hat{f}_k \circ j_k$ .

□

Note that for every edge  $uv$  in  $\mathcal{G}^{\alpha(1+\varepsilon)^k}(V_0, V_{k+1})$ , there is an edge  $ab$  in  $\mathcal{R}^{\alpha(1+\varepsilon)^k}(V_0)$  such that  $\hat{\pi}_k(a) = u$  and  $\hat{\pi}_k(b) = v$ . Since  $V_{k+1}$  is a  $\frac{\alpha\varepsilon}{2}(1+\varepsilon)^k$ -sample of  $V_0$ , one has that

$$|uv| \leq |ua| + |vb| + |ab| \leq \alpha\varepsilon(1+\varepsilon)^k + \alpha(1+\varepsilon)^k = \alpha(1+\varepsilon)^{k+1}.$$

Therefore, the 1-skeleton of the graph induced complex  $\mathcal{G}^{\alpha(1+\varepsilon)^k}(V_0, V_{k+1})$  is a subcomplex of the 1-skeleton of  $\mathcal{R}^{\alpha(1+\varepsilon)^{k+1}}(V_{k+1})$ . Consequently,  $\mathcal{G}^{\alpha(1+\varepsilon)^k}(V_0, V_{k+1})$  is a subcomplex of  $\mathcal{R}^{\alpha(1+\varepsilon)^{k+1}}(V_{k+1})$  which is the maximal simplicial complex containing its 1-skeleton. This observation implies that the sequence (3) has smaller size compared to the sequence (2). Furthermore, although the asymptotic space complexity of each GIC is the same as that of the sparsified Rips complex, in practice, the size of GICs can be far smaller; see [17]. However, the construction of each GIC is more expensive, as one needs to compute each  $\mathcal{G}^{\alpha(1+\varepsilon)^k}(V_0, V_{k+1})$  from the Rips complex  $\mathcal{R}^{\alpha(1+\varepsilon)^k}(V_0)$  built on the original vertex set  $V_0$ , instead of the vertex set from the previous complex  $\mathcal{G}^{\alpha(1+\varepsilon)^{k-1}}(V_0, V_k)$ . Hence there is a trade-off of space versus time for the approaches given in Section 6.

## 7. CONCLUSIONS

In this paper, we studied algorithms to compute the persistence diagram of a (monotone) filtration connected by simplicial maps efficiently. As discussed in [13], the algorithm based on the cohomology view in [14] has a good practical performance for the case of computing inclusion-induced non-zigzag persistence. Our annotation-based algorithm extends such a view of maintaining an appropriate cohomology basis to the case of vertex collapses. This allows us to compute the persistence diagram for a filtration connected by simplicial maps directly and efficiently. The coning approach in Section 3 works for any finite fields though the collapse based algorithm in Section 5 currently works with  $\mathbb{Z}_2$  coefficients only. Although inclusions can be handled under other finite field coefficients, it is not clear how to handle collapses efficiently.

We believe that, as the scope of topological data analysis continues to broaden, further applications based on simplicial maps will arise. Currently, an efficient implementation of the persistence algorithm taking advantage of the compressed representation of annotations has been reported in [2]. We have also developed an efficient implementation of the persistence algorithms for simplicial maps in the same vein. The software named **SimpPers** is available from authors' web-pages.

## 8. REFERENCES

- [1] D. Attali, A. Lieutier, and D. Salinas. Efficient data structure for representing and simplifying simplicial complexes in high dimensions. In *Proceedings of the Twenty-seventh Annual Symposium on Computational Geometry*, SoCG '11, pages 501–509, 2011.
- [2] J.-D. Boissonnat, T. K. Dey, and C. Maria. The compressed annotation matrix: An efficient data structure for computing persistent cohomology. In *European Symposium on Algorithms 2013*, volume 8125 of *Lecture Notes in Computer Science*, pages 695–706, 2013.
- [3] D. Burghlelea and T. K. Dey. Topological persistence for circle-valued maps. *Discrete Comput. Geom.*, 50(1):69–98, 2013.
- [4] O. Busaryev, S. Cabello, C. Chen, T. K. Dey, and Y. Wang. Annotating simplices with a homology basis and its applications. In *Proceedings of the 13th Scandinavian Conference on Algorithm Theory*, SWAT'12, pages 189–200, 2012.
- [5] G. Carlsson. Topology and data. *Bull. Amer. Math. Soc.*, 46:255–308, 2009.
- [6] G. Carlsson and V. de Silva. Zigzag persistence. *Foundations of Computational Mathematics*, 10(4):367–405, 2010.
- [7] G. Carlsson, V. de Silva, and D. Morozov. Zigzag persistent homology and real-valued functions. In *Proceedings of the Twenty-fifth Annual Symposium on Computational Geometry*, SCG '09, pages 247–256, 2009.
- [8] F. Chazal, D. Cohen-Steiner, M. Glisse, L. J. Guibas, and S. Y. Oudot. Proximity of persistence modules and their diagrams. In *Proceedings of the Twenty-fifth Annual Symposium on Computational Geometry*, SCG '09, pages 237–246, 2009.
- [9] C. Chen and M. Kerber. An output-sensitive algorithm for persistent homology. *Comput. Geom. Theory Appl.*, 46(4):435–447, 2013.
- [10] K. L. Clarkson. Nearest-neighbor searching and metric space dimensions. In *In Nearest-Neighbor Methods for Learning and Vision: Theory and Practice*, pages 15–59, 2006.
- [11] D. Cohen-Steiner, H. Edelsbrunner, and J. Harer. Stability of persistence diagrams. *Discrete & Computational Geometry*, 37(1):103–120, 2007.
- [12] D. Cohen-Steiner, H. Edelsbrunner, and D. Morozov. Vines and vineyards by updating persistence in linear time. In *Proceedings of the Twenty-second Annual Symposium on Computational Geometry*, SCG '06, pages 119–126, 2006.
- [13] V. de Silva, D. Morozov, and M. Vejdemo-Johansson. Dualities in persistent (co)homology. *Inverse Problems*, 27(12):124003, 2011.
- [14] V. de Silva, D. Morozov, and M. Vejdemo-Johansson. Persistent cohomology and circular coordinates. *Discrete & Computational Geometry*, 45(4):737–759, 2011.
- [15] T. K. Dey, H. Edelsbrunner, S. Guha, and D. V. Nekhayev. Topology preserving edge contraction. *Publ. Inst. Math. (Beograd) (N.S.)*, 66:23–45, 1999.
- [16] T. K. Dey, F. Fan, and Y. Wang. Computing topological persistence for simplicial maps, 2012. arXiv:1208.5018v3.
- [17] T. K. Dey, F. Fan, and Y. Wang. Graph induced complex on point data. In *Proceedings of the Twenty-ninth Annual Symposium on Computational Geometry*, SoCG '13, pages 107–116, 2013.
- [18] T. K. Dey, J. Sun, and Y. Wang. Approximating cycles in a shortest basis of the first homology group from point data. *Inverse Problems*, 27(12):124004, 2011.
- [19] H. Edelsbrunner and J. Harer. *Computational Topology*. American Mathematical Society, 2009.
- [20] H. Edelsbrunner, D. Letscher, and A. Zomorodian. Topological persistence and simplification. *Discrete & Computational Geometry*, 28(4):511–533, 2002.
- [21] R. Ghrist. Barcodes: The persistent topology of data. *Bull. Amer. Math. Soc.*, 45:61–75, 2008.
- [22] S. Har-Peled and M. Mendel. Fast construction of nets in low dimensional metrics, and their applications. In *Proceedings of the Twenty-first Annual Symposium on Computational Geometry*, SCG '05, pages 150–158, 2005.
- [23] A. Hatcher. *Algebraic Topology*. Cambridge U. Press, New York, 2002.
- [24] N. Milosavljević, D. Morozov, and P. Skraba. Zigzag persistent homology in matrix multiplication time. In *Proceedings of the Twenty-seventh Annual Symposium on Computational Geometry*, SoCG '11, pages 216–225, 2011.
- [25] D. R. Sheehy. Linear-size approximations to the vietoris-rips filtration. *Discrete & Computational Geometry*, 49(4):778–796, 2013.
- [26] A. Zomorodian and G. Carlsson. Computing persistent homology. *Discrete Comput. Geom.*, 33:249–274, 2005.

## APPENDIX

### A. MISSING PROOF FROM SECTION 2

**Proof of Proposition 2.5.** Consider the surjective simplicial map  $f' : K \rightarrow f(K)$  defined as  $f'(\sigma) = f(\sigma)$  for any  $\sigma \in K$ . Writing  $V = V(K)$  and  $V' = V(K')$ , we have  $f'_V = f_V$ . The simplicial map  $f : K \rightarrow K'$  is a composition  $i \circ f'$ , where  $i : f(K) \hookrightarrow K'$  is the canonical inclusion  $f(K) \subseteq K'$ . Obviously, the inclusion  $i$  can be easily decomposed into a sequence of elementary inclusions. We now show that  $f'$  can be decomposed into a sequence of elementary collapses.

Let  $A := \{v \in V' \mid |f_V^{-1}(v)| > 1\}$ . Hence  $f_V$  maps injectively onto  $V' \setminus A$ . Order vertices in  $A$  arbitrarily as  $\{v_1, \dots, v_k\}$ ,  $k = |A|$ , and let  $A_i$  denote  $f_V^{-1}(v_i)$ . We now define  $f_i$  and  $K_i$  in increasing order of  $i$ . For the base case, set  $K_0 = K$ . For any  $i > 0$ , consider the vertex map  $f_{A_i}$  which is the injective map on  $V(K_{i-1}) \setminus A_i$ , but maps  $A_i$  to  $v_i$ . We set  $f_i$  to be the simplicial map induced by this vertex map  $f_{A_i}$ , and set  $K_i := f_i(K_{i-1})$ . By construction,  $f_i$  is a surjective simplicial map.

It is easy to see that the vertex map  $f_{A_k} \circ \dots \circ f_{A_1}$  equals  $f_V$ . Hence, the induced simplicial map  $f_k \circ \dots \circ f_1 : K \rightarrow K_k$  equals  $f' : K \rightarrow f(K)$ . Furthermore, each  $f_i$  can be decomposed into a sequence of elementary collapses, each induced by a vertex map that maps only two vertices from  $A_i$  to  $v_i$ .