

ALGORITHMS AND HARDNESS RESULTS IN COMPUTATIONAL HOMOLOGY

By

Chao Chen

A Thesis Submitted to the Graduate
Faculty of Rensselaer Polytechnic Institute
in Partial Fulfillment of the
Requirements for the Degree of
DOCTOR OF PHILOSOPHY
Major Subject: COMPUTER SCIENCE

Approved by the
Examining Committee:

Daniel Freedman, Thesis Adviser

Herbert Edelsbrunner, Member

Mark K. Goldberg, Member

Bruce Piper, Member

Jeff Trinkle, Member

Rensselaer Polytechnic Institute
Troy, New York

July 2009
(For Graduation August 2009)

ALGORITHMS AND HARDNESS RESULTS IN COMPUTATIONAL HOMOLOGY

By

Chao Chen

An Abstract of a Thesis Submitted to the Graduate

Faculty of Rensselaer Polytechnic Institute

in Partial Fulfillment of the

Requirements for the Degree of

DOCTOR OF PHILOSOPHY

Major Subject: COMPUTER SCIENCE

The original of the complete thesis is on file
in the Rensselaer Polytechnic Institute Library

Examining Committee:

Daniel Freedman, Thesis Adviser

Herbert Edelsbrunner, Member

Mark K. Goldberg, Member

Bruce Piper, Member

Jeff Trinkle, Member

Rensselaer Polytechnic Institute
Troy, New York

July 2009
(For Graduation August 2009)

© Copyright 2013
by
Chao Chen
All Rights Reserved

CONTENTS

| | |
|---|----|
| LIST OF TABLES | iv |
| LIST OF FIGURES | v |
| 1. INTRODUCTION | 1 |
| 1.1 Outline | 4 |
| 2. PRELIMINARIES | 5 |
| 2.1 Simplicial Complex | 5 |
| 2.2 Homology Groups | 5 |
| 2.3 Persistent Homology | 8 |
| 2.3.1 Definition | 8 |
| 2.3.2 Algorithm | 9 |
| 2.4 Relative Homology | 10 |
| 2.5 Hardness of Approximability and Strict Reductions | 11 |
| 3. LITERATURE REVIEWS | 14 |
| 3.1 Topology Related Computations | 14 |
| 3.1.1 Computing the Homology Group | 14 |
| 3.1.2 Persistent Homology | 15 |
| 3.1.3 Discrete Morse Theory | 16 |
| 3.1.4 Topologically Faithful Manifold Reconstruction | 17 |
| 3.2 Homology Measurement and Localization | 17 |
| 4. MEASURING | 20 |
| 4.1 Definitions | 20 |
| 4.1.1 Measuring the Size of a Homology Class | 20 |
| 4.1.2 The Optimal Homology Basis | 20 |
| 4.2 The Algorithm | 22 |
| 4.2.1 Computing the Smallest Homology Class | 22 |
| 4.2.2 Computing the Optimal Homology Basis | 23 |
| 4.2.3 Complexity of the Non-Refined Algorithm | 24 |
| 4.3 An Improvement Using Finite Field Linear Algebra | 25 |

| | | |
|-------|---|----|
| 4.3.1 | Observation: Neighboring Vertices Have Similar Geodesic Distance Functions | 25 |
| 4.3.2 | Procedure Contain-Nonbounding-Cycle : Testing Whether a Subcomplex Carries Nonbounding Cycles of K | 27 |
| 4.3.3 | Complexity of the Improved Algorithm | 29 |
| 4.3.4 | Measuring a Given Class | 29 |
| 4.4 | Stability Result. | 30 |
| 4.4.1 | A Single Class | 30 |
| 4.4.2 | The Homology Group | 31 |
| 4.4.3 | Continuous Case | 32 |
| 5. | Homology Localization | 35 |
| 5.1 | Problem Formalization | 35 |
| 5.2 | Volume | 36 |
| 5.2.1 | LHV is NP-hard to approximate within any constant factor | 37 |
| 5.2.2 | Computing the minimal volume nonbounding cycle | 39 |
| 5.2.3 | A special case | 41 |
| 5.3 | Diameter | 42 |
| 5.4 | Radius | 44 |
| 6. | Open Problem: Enriched Persistent Homology | 47 |
| 6.1 | Motivation | 47 |
| 6.2 | Total Persistent Homology | 48 |
| 6.3 | Enriched Persistences and Instability Cases | 50 |
| 6.4 | Updating Total Persistent Homology | 52 |
| 7. | Application of Computational Topology Methods in Ribosome Drug Docking | 57 |
| 7.1 | Problem: Ribosome Drug Docking | 57 |
| 7.2 | Method: Applying Computational Topology Methods on Ribosome Drug Docking | 58 |
| 7.2.1 | Alpha Complexes | 59 |
| 7.2.2 | Pockets and Cavities | 59 |
| 7.2.3 | Results | 60 |
| 7.3 | Discussion of Binding Sites | 62 |
| 7.3.1 | Data | 62 |
| 7.3.2 | Shapes of binding sites | 62 |
| | REFERENCES | 65 |

LIST OF TABLES

| | | |
|-----|--|----|
| 6.1 | Bases formed by different sets of chains in total reduction. | 48 |
| 6.2 | Matrices for updating total persistence. | 53 |
| 7.1 | Numbers of different drugs binding to 30s or 70s ribosome. | 62 |
| 7.2 | List of drugs. | 63 |

LIST OF FIGURES

| | | |
|-----|--|----|
| 1.1 | Spaces with different topology in different scales. | 1 |
| 1.2 | Choosing a natural homology basis. | 2 |
| 1.3 | Motivating examples for localization. | 3 |
| 2.1 | Simplicial homology. | 7 |
| 2.2 | Relative homology. | 11 |
| 3.1 | Extended persistent homology. | 18 |
| 3.2 | Handles and related 1-cycles. | 19 |
| 4.1 | Geodesic balls carrying homology classes. | 21 |
| 4.2 | Computing the optimal homology basis in a greedy manner. | 24 |
| 4.3 | Updating persistence cannot improve the measurement algorithm. | 26 |
| 4.4 | Improving the measurement algorithm. | 27 |
| 5.1 | LHV is NP-hard to approximate. | 38 |
| 5.2 | A special case in which MVNC is polynomial. | 41 |
| 5.3 | Explanation of Theorem 5.3.4 proof. | 43 |
| 5.4 | Cycles with the minimal radius and diameter. | 45 |
| 6.1 | Enriched Persistent Homology: Examples. | 47 |
| 6.2 | Enriched Persistent Homology: Counterexamples. | 51 |
| 6.3 | Enriched Persistent Homology: Reduction/Inclusion Edges are Unstable. | 52 |
| 7.1 | 30s and 50s ribosomes. RNAs are in grey color and proteins are in other colors. A, P and E sites are pointed out. The pictures are from http://www.riboworld.com/ | 58 |
| 7.2 | An example of pockets. | 61 |
| 7.3 | 50s and the two functional regions we isolated. | 61 |
| 7.4 | 30s and the functional region we isolated. | 61 |

| | | |
|-----|--|----|
| 7.5 | Hygromycin B (left) and Kasugamycin (right) are bound in grooves with the right width. | 63 |
| 7.6 | Edeine B is bound in a tunnel, after pushing a few side chains away. . . | 64 |

CHAPTER 1

INTRODUCTION

In recent years, the problem of computing the topological features of a space has drawn much attention. There are two reasons for this. The first is a general observation: compared with geometric features, topological features are more qualitative and global, and tend to be more robust. If the goal is to characterize a space, therefore, features which incorporate topology seem to be good candidates.

The second reason is that topology plays an important role in a number of applications. Researchers in graphics need topological information to facilitate parameterization of surfaces and texture mapping [51, 15]. In the field of sensor networks, the use of homological tools is crucial for certain coverage problems [33]. Computational biologists use topology to study protein docking and folding problems [1, 28]. Finally, topological features are especially important in high dimensional data analysis, where purely geometric tools are often deficient, and full-blown space reconstruction is expensive and often ill-posed [11, 58].

Once we are able to compute topological features, a natural problem is to rank the features according to their importance. The significance of this problem can be justified from two perspectives.

1. In graphics and manifold learning, unavoidable errors are introduced in data acquisition, in the form of traditional signal noise, and finite sampling of continuous spaces. These errors may lead to the presence of many small topological features that are not “real”, but are simply artifacts of noise or of sampling [62, 89, 77].
2. Many problems are naturally hierarchical. This hierarchy – which is a kind of multi-scale or multi-resolution decomposition – implies that we want to capture the large scale features first. See Figure 1.1 for examples.

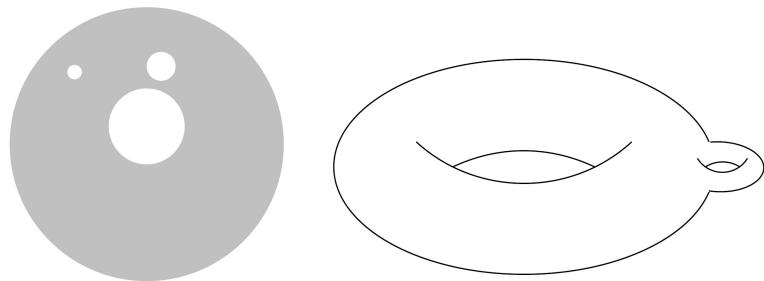


Figure 1.1: A disk with three holes and a 2-handled torus are really more like an annulus and a 1-handled torus, respectively, because the large features are more important.

There are a variety of ways of characterizing topological spaces in the literature, including fundamental groups, homology groups, and the Euler characteristic. The topological features we use are homology groups over \mathbb{Z}_2 , due to their ease of computation. (Thus, throughout this thesis, except where noted, all the additions are mod 2 additions.)

Ranking the homology classes according to their importance involves the following two subproblems.

1. **Measuring the size of a homology class:** We need a way to quantify the size of a given homology class, and this size measure should agree with intuition. For example, in Figure 1.1 (left), the measure should be able to distinguish the one large class (of the 1-dimensional homology group) from the two smaller classes. Furthermore, the measure should be easy to compute, and applicable to homology groups of any dimension. To compute this measurement, we have to use the geometry of the space. This means the space has a metric defined. Throughout this thesis, we may use either “space” or “topological space”. What we really mean is a topological space with a metric defined on it, i.e. a metric space.
2. **Choosing a natural basis for a homology group:** We would like to choose a “good” set of homology classes to be the generators for the homology group (of a fixed dimension). Suppose that β is the dimension of this group, and that we are using \mathbb{Z}_2 coefficients; then there are $2^\beta - 1$ nontrivial homology classes in total. For a basis, we need to choose a subset of β of these classes, subject to the constraint that these β generate the group. The criterion of goodness for a basis is based on an overall size measure for the basis, which relies in turn on the size measure for its constituent classes. For instance, in Figure 1.2, we must choose three from the seven nontrivial 1-dimensional homology classes: $\{[z_1], [z_2], [z_3], [z_1] + [z_2], [z_1] + [z_3], [z_2] + [z_3], [z_1] + [z_2] + [z_3]\}$. In this case, the intuitive choice is $\{[z_1], [z_2], [z_3]\}$, as this choice reflects the fact that there is really only one large cycle.

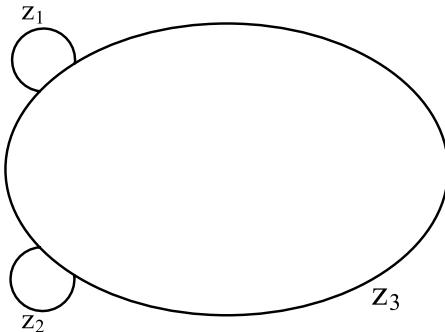


Figure 1.2: A topological space formed from three circles. See accompanying discussion in the text.

Besides detecting small homology classes, the need to “localize” any homology classes has been raised in a variety of applications. In graphics and manifold learning, finding concise cycles representing topological noise such as the small holes and handles could help us removing them [62, 89, 77]. In the area of sensor networks, holes of the coverage region, caused by physical constraints, should be accurately identified and described so as to produce as robust a network as possible [53, 82]. In the study of 3D shape, homology localization could help us identify topology related features like handles and tunnels [37, 38]. Shapes may be enriched with properties such as curvatures associated with tangent vectors at each tangent plane. The new augmented shape lives in high dimension, whose topological features can be localized and reveal geometric features of the original shape [10].

The *localization problem* is finding the smallest representative cycle of a homology class with regard to a given natural criterion of the size of a cycle. The criterion should be deliberately chosen so that the corresponding smallest cycle is concise in not only mathematics but also intuition. Such a cycle is a “well-localized” representative cycle of its class. See Figure 1.3 for examples.

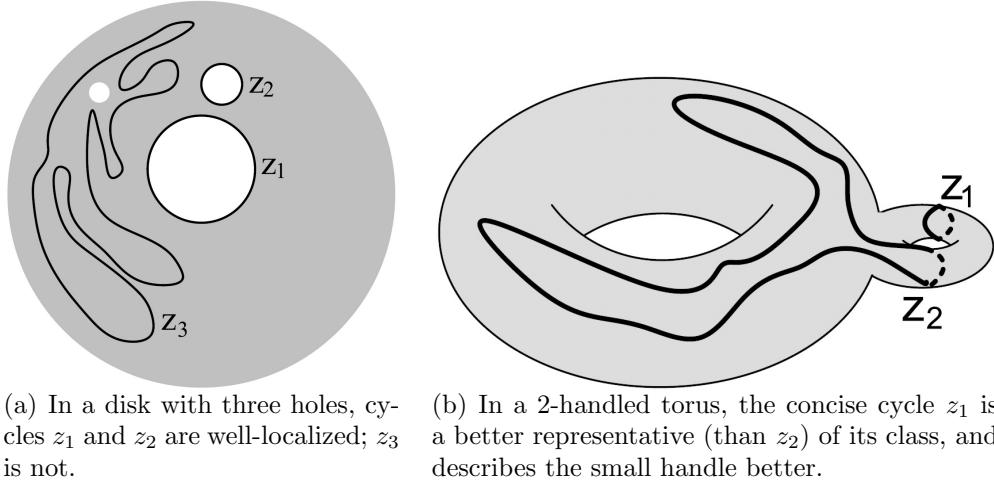


Figure 1.3: Motivating examples for localization.

Furthermore, we require the solution of aforementioned problems to be (a) computable for general dimensions (b) not involving the embedding space of the space. Although there are existing low dimensional techniques we may use, to our knowledge, there are no definitions and algorithms satisfying the two requirements. These requirements can be justified by the following two applications.

- In data analysis, it is often assumed that the data lives in a manifold whose dimension is much smaller than the dimension of the embedding space.
- In the study of shape, it is common to enrich the shape with other quantities, such as curvature, or color and other physical quantities. This leads to high dimensional manifolds embedded in high dimensional ambient spaces.

1.1 Outline

In Chapter 2, we prepare readers with background knowledge, including homology, persistent homology, relative homology, as well as hardness of approximation.

In the literature review (Chapter 3), we cover research in general topology related computations as well as works closely related to homology measurement and localization.

In Chapter 4, we present definitions and algorithms for the measurement of homology classes and the optimal homology basis.

In Chapter 5, we address homology localization problem. We prove hardness results of the problem.

In Chapter 6, we study enriched persistent homology, which has a close relationship with the localization of persistent homology classes.

CHAPTER 2

PRELIMINARIES

In this section, we briefly describe the background necessary for our work, including a discussion of simplicial complexes, homology groups, persistent homology, relative homology and hardness of approximation. Please refer to [75] for further details in algebraic topology, [?] for persistent homology and [7] for hardness of approximability.

2.1 Simplicial Complex

A *d-dimensional simplex* or *d-simplex*, σ , is the convex hull of $d + 1$ *affinely independent vertices*, which means for any of these vertices, v_i , the d vectors $v_j - v_i$, $j \neq i$, are linearly independent. A 0-simplex, 1-simplex, 2-simplex and 3-simplex are a vertex, edge, triangle and tetrahedron, respectively. The convex hull of a nonempty subset of vertices of σ is its *face*. A *simplicial complex* K is a finite set of simplices that satisfies the following two conditions.

1. Any face of a simplex in K is also in K .
2. The intersection of any two simplices in K is either empty or is a face for both of them.

The *dimension* of a simplicial complex is the highest dimension of its simplices. If a subset $K_0 \subseteq K$ is a simplicial complex, it is a *subcomplex* of K .

2.2 Homology Groups

In this thesis, except for the literature review, we only use simplicial homology of \mathbb{Z}_2 coefficients, which is introduced in this section. For completeness, in the end of this section, we briefly cover simplicial homology of other coefficient rings.

Within a given simplicial complex K , a *d-chain* is a formal sum of d -simplices in K ,

$$c = \sum_{\sigma \in K} a_\sigma \sigma, a_\sigma \in \mathbb{Z}_2.$$

All the d -chains form the *group of d-chains*, $C_d(K)$. The *boundary* of a d -chain is the sum of the $(d - 1)$ -faces of all the d -simplices in the chain. The boundary operator $\partial_d : C_d(K) \rightarrow C_{d-1}(K)$ is a group homomorphism.

A *d-cycle* is a d -chain without boundary. The set of d -cycles forms a subgroup of the chain group, which is the kernel of the boundary operator, $Z_d(K) = \ker(\partial_d)$. A *d-boundary* is the boundary of a $(d + 1)$ -chain. The set of d -boundaries forms a group, which is the image of the boundary operator, $B_d(K) = \text{img}(\partial_{d+1})$. It is not hard to see that a d -boundary is also a d -cycle. Therefore, $B_d(K)$ is a subgroup of

$\mathbb{Z}_d(K)$. A d -cycle which is not a d -boundary, $z \in \mathbb{Z}_d(K) \setminus \mathbb{B}_d(K)$, is a *nonbounding cycle*. In our case, the coefficients belong to a field, namely \mathbb{Z}_2 ; when this is the case, the groups of chains, boundaries and cycles are all vector spaces. Note that this is not true when the homology is over a ring which is not a field, such as \mathbb{Z} .

The d -dimensional homology group is defined as the quotient group $\mathsf{H}_d(K) = \mathbb{Z}_d(K)/\mathbb{B}_d(K)$. An element in $\mathsf{H}_d(K)$ is a *homology class*, which is a coset of $\mathbb{B}_d(K)$, $[z] = z + \mathbb{B}_d(K)$ for some d -cycle $z \in \mathbb{Z}_d(K)$. If z is a d -boundary, $[z] = \mathbb{B}_d(K)$ is the identity element of $\mathsf{H}_d(K)$. Otherwise, when z is a nonbounding cycle, $[z]$ is a *nontrivial homology class* and z is called a *representative cycle* of $[z]$. Cycles in the same homology class are *homologous* to each other, which means their difference is a boundary.

The dimension of the homology group, which is referred to as the *Betti number*,

$$\beta_d = \dim(\mathsf{H}_d(K)) \quad (2.1)$$

$$= \dim(\mathbb{Z}_d(K)) - \dim(\mathbb{B}_d(K)). \quad (2.2)$$

As the dimension of the chain group is upper bounded by the cardinality of K , n , so are the dimensions of $\mathbb{B}_d(K)$, $\mathbb{Z}_d(K)$ and $\mathsf{H}_d(K)$. The Betti number can be computed with a reduction algorithm based on row and column operations of the boundary matrices [75]. Various reduction algorithms have been devised for different purposes [64, ?, 92].

A *homology basis* is a set of β_d classes generating the group $\mathsf{H}_d(K)$. We call a set of β_d nonbounding cycles representing a homology basis a *homology cycle basis*. Any d -cycle can be written as the linear combination of a homology cycle basis and boundaries.

Note that since the field is \mathbb{Z}_2 , the set of d -chains is in one-to-one correspondence with the set of subsets of d -simplices. A d -chain corresponds to a n_d -dimensional vector, whose nonzero entries correspond to the included d -simplices. Here n_d is the number of d -simplices in K . Computing the boundary of a d -chain corresponds to multiplying the chain vector with a boundary matrix $[b_1, \dots, b_{n_d}]$, whose column vectors are boundaries of d -simplices in K . By slightly abusing notation, we call the boundary matrix ∂_d . By linear algebra, Equation (2.2) could be written as

$$\beta_d = (n_d - \text{rank}(\partial_d)) - \text{rank}(\partial_{d+1}).$$

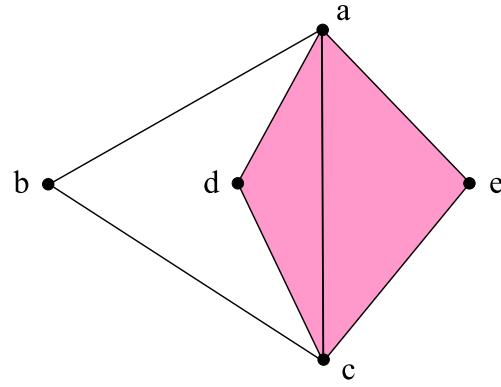


Figure 2.1: A simplicial complex K containing five 0-simplices, seven 1-simplices and two 2-simplices.

See Figure 2.1 for a simplicial complex whose boundary matrices are

$$\partial_1 = \left[\begin{array}{c|ccccccc} & ab & ac & ad & ae & bc & cd & ce \\ \hline a & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ b & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ c & 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ d & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ e & 0 & 0 & 0 & 1 & 0 & 0 & 1 \end{array} \right] \quad \text{and} \quad \partial_2 = \left[\begin{array}{c|cc} & acd & ace \\ \hline ab & 0 & 0 \\ ac & 1 & 1 \\ ad & 1 & 0 \\ ae & 0 & 1 \\ bc & 0 & 0 \\ cd & 1 & 0 \\ ce & 0 & 1 \end{array} \right].$$

This simplicial complex has 1 nontrivial homology class, represented by three different nonbounding cycles, $(ab + bc + cd + da)$, $(ab + bc + ca)$ and $(ab + bc + ce + ea)$, whose corresponding vectors are $(1, 0, 1, 0, 1, 1, 0)^T$, $(1, 1, 0, 0, 1, 0, 0)^T$ and $(1, 0, 0, 1, 1, 0, 1)^T$ respectively.

The following notation will prove convenient. We say that a d -chain $c \in C_d(K)$ is *carried by* a subcomplex K_0 when all the d -simplices of c belong to K_0 , formally, $c \subseteq K_0$. We denote $\text{vert}(K)$ as the set of vertices of the simplicial complex K , $\text{vert}(c)$ as that of the chain c . Denote $|K|$ as the underlying space of K , $|c|$ as that of the chain c . We denote the *interior* of a simplex as $\text{int}(\sigma) = |\sigma| \setminus |\partial\sigma|$.

Replacing simplices by their continuous images in a given topological space gives singular homology. The simplicial homology of a simplicial complex is naturally isomorphic to the singular homology of its geometric realization. This implies, in particular, that the simplicial homology of a space does not depend on the particular simplicial complex chosen for the space. In figures of this thesis, we often ignore the simplicial complex and only show the continuous images of chains.

Homology with Other Coefficient Rings. In general, the coefficients may belong to arbitrary abelian groups. A chain is a sum of equivalent classes, each such

class correspond to a simplex with certain orientation. In this thesis, we restrict the abelian group to a ring and call it the *coefficient ring*, R . When R is not a field, the homology group, $\mathsf{H}_d(K)$, may not be a vector space. As a finitely generated abelian group, it has the structure

$$\begin{aligned}\mathsf{H}_d(K) &\cong \mathsf{H}_d^F \oplus \mathsf{H}_d^T, \\ \mathsf{H}_d^F &\cong \mathbb{Z}^{\beta_d}, \\ \mathsf{H}_d^T &\cong \mathbb{Z}_{t_1} \oplus \mathbb{Z}_{t_2} \oplus \cdots \oplus \mathbb{Z}_{b_k}, \quad t_i | t_{i+1} \quad \forall 1 \leq i \leq k-1.\end{aligned}$$

The rank of the free part H_d^F is the Betti number of dimension d , β_d . The numbers t_1, \dots, t_k are called the *torsion coefficients* and H_d^T is called the *torsion subgroup*. The Betti number and the torsion coefficients can be computed with a reduction algorithm based on row and column operations of the boundary matrices. When R is a field, the torsion subgroup becomes trivial. The chain group, boundary group and homology group all become vector spaces. See [75] for more details.

2.3 Persistent Homology

2.3.1 Definition

We first give the intuition. Given a topological space \mathbb{X} and a *filter function* $f : \mathbb{X} \rightarrow \mathbb{R}$, *persistent homology* studies the homology classes of the sublevel sets, $\mathbb{X}^t = f^{-1}(-\infty, t]$. A nontrivial homology class in \mathbb{X}^{t_1} may become trivial in \mathbb{X}^{t_2} , $t_1 < t_2$, (formally, when induced by the inclusion homomorphism). Persistent homology tries to capture this phenomenon by measuring the times at which a homology class is born and dies. By born we mean the homology class comes into being. By die we mean it either becomes trivial or becomes identical to some other class born earlier. The persistence, or life time of the class is the difference between its death and birth times. Those with longer lives tell us something about the global structure of the space \mathbb{X} , as described by the filter function. Note that the *essential*, that is, nontrivial homology classes of the given topological space \mathbb{X} will never die.

Edelsbrunner et al. [48, 91] defined the persistent homology of a simplicial complex K and filtered by a function $f : K \rightarrow \mathbb{R}$, which assigns each simplex in K a real value. Simplices of K are sorted in ascending order according to their filter function values,

$$\{\sigma_1, \sigma_2, \dots, \sigma_m\}, \quad f(\sigma_i) \leq f(\sigma_{i+1}), \quad \forall 1 \leq i \leq m-1,$$

namely, the *simplex-ordering* of K with regard to f . This is the order in which simplices enter the sublevel set $f^{-1}(-\infty, t]$ while t increases. Note that within the simplex-ordering, a simplex must appear after all of its faces. With this restriction, any sublevel set is a subcomplex, denoted as K_i , which has and only has $\{\sigma_1, \dots, \sigma_i\}$

as its simplices. The nested sequence of sublevel sets

$$\emptyset = K_0 \subset K_1 \subset \cdots \subset K_m = K$$

is called a *filtration* of K . Let $f_i = f(\sigma_i)$ and $f_0 = -\infty$, $K_i = f^{-1}(-\infty, f_i]$.

For any $0 < i < j \leq m$, the inclusion mapping of K_i into K_j induces a group homomorphism of the corresponding homology group,

$$F_d^{i,j} : \mathbf{H}_d(K_i) \rightarrow \mathbf{H}_d(K_j).$$

A *persistent homology class*, h , is born at the time f_i if $h \in \mathbf{H}_d(K_i)$ but $h \notin \text{img}(F_d^{i-1,i})$. Given h is born at f_i , h dies at time f_j if $F_d^{i,j-1}(h) \notin \text{img}(F_d^{i-1,j-1})$ but $F_d^{i,j}(h) \in \text{img}(F_d^{i-1,j})$. Any class in the coset $h + \text{img}(F_d^{i-1,i})$ is born at f_i and dies at f_j . The *persistence* of a persistent homology class is defined as the difference between its death and birth times, which quantifies the significance of the feature. Not all the persistent homology classes die. Those which never die are *essential* classes, which correspond to nontrivial homology classes of K . An essential homology class has the $+\infty$ death time, and thus, an infinite persistence.

The persistent homology can be described using a *persistence diagram*, in which each persistent homology class corresponds to a point whose coordinates are its birth and death times. Important features correspond to points further away from the diagonal in the persistence diagram. The persistence diagram includes all the points corresponding to persistent homology classes, as well as the diagonal line. It is proved to be stable when the filter function has a small change [25].

2.3.2 Algorithm

Edelsbrunner et al. [48] devised an $O(n^3)$ algorithm to compute the persistent homology. Its input are a simplicial complex and a filter function f . Its output are the birth and death times of all the persistent homology classes. The latest version of this algorithm [28, ?] unifies boundary matrices of different dimensions into one overall *incidence matrix* D . Rows and columns of D correspond to simplices of K , indexed in the simplex-ordering. An entry of D is 1 if and only if its corresponding entry is 1 in the corresponding boundary matrix. The algorithm performs column reductions on D from left to right. Each new column is reduced by addition with the already reduced columns, until its lowest nonzero entry is as high as possible.

More specifically, during the reduction, record $\text{low}(i)$ as the lowest nonzero entry of each column i . To reduce column i , we repeatedly find column j satisfying $j < i$ and $\text{low}(j) = \text{low}(i)$; we then add column j to column i , until column i becomes a zero column or we cannot find a qualified j anymore. If column i is reduced to a zero column, $\text{low}(i)$ does not exist.

The reduction of D can be written as a matrix multiplication,

$$R = DV, \tag{2.3}$$

where R is the *reduced matrix* and V is an upper triangular matrix. The reduced matrix R provides $\text{rank}(D)$ many pairings of simplices, $(\sigma_i, \sigma_j) : \text{low}(j) = i$, in which each simplex appears at most once. Each pairing corresponds to a persistent homology class whose birth time is $f(\sigma_i)$ and death time is $f(\sigma_j)$. Unpaired simplices are paired with $+\infty$ and correspond to essential classes. Simplices paired with $+\infty$ or paired on the left are *positive*, and the rest are *negative*.

The reduction is completely recorded in the matrix V . Columns of V corresponding to positive simplices form bases of cycle groups. Columns corresponding to positive simplices paired with $+\infty$ are cycles representing essential classes and form homology cycle bases.

2.4 Relative Homology

Given a simplicial complex K and a subcomplex $K_0 \subseteq K$, we may wish to study the structure of K by ignoring all the chains in K_0 . We consider two d -chains, c_1 and c_2 to be the same if their difference is carried by K_0 . The objects we are interested in are then defined as these equivalence classes, which form a quotient group, $C_d(K, K_0) = C_d(K)/C_d(K_0)$. We call it the *group of relative chains*, whose elements (cosets), are called *relative chains*.

The boundary operator $\partial_d : C_d(K) \rightarrow C_{d-1}(K)$ induces a *relative boundary operator*, $\partial_d^{K_0} : C_d(K, K_0) \rightarrow C_{d-1}(K, K_0)$. Analogous to the way we define $Z_d(K)$, $B_d(K)$ and $H_d(K)$ in $C_d(K)$, we define the *group of relative cycles*, the *group of relative boundaries* and the *relative homology group* in $C_d(K, K_0)$, denoted as $Z_d(K, K_0)$, $B_d(K, K_0)$ and $H_d(K, K_0)$, respectively. An element in $Z_d(K, K_0) \setminus B_d(K, K_0)$ is a *nonbounding relative cycle*.

The following notation will prove convenient. We define a homomorphism $\phi_{K_0} : C_d(K) \rightarrow C_d(K, K_0)$ mapping d -chains to their corresponding relative chains, $\phi_{K_0}(c) = c + C_d(K_0)$. This homomorphism induces another homomorphism, $\phi_{K_0}^* : H_d(K) \rightarrow H_d(K, K_0)$, mapping homology classes of K to their corresponding relative homology classes, $\phi_{K_0}^*(h) = \phi_{K_0}(z) + B_d(K, K_0)$ for any $z \in h$.

Given a d -chain $c \in C_d$, its corresponding relative chain $\phi_{K_0}(c)$ is a relative cycle if and only if $\partial_d(c)$ is carried by K_0 . Furthermore, it is a relative boundary if and only if there is a $(d+1)$ -chain $c' \in C_{d+1}(K)$ such that $c - \partial_{d+1}(c')$ is carried by K_0 .

These ideas are illustrated in Figure 2.2. Although z_1 and z_2 are both nonbounding cycles in K , $\phi_{K_0}(z_1)$ is a nonbounding relative cycle whereas $\phi_{K_0}(z_2)$ is only a relative boundary. Although chains c_1 and c_2 are not cycles in K , $\phi_{K_0}(c_1)$ and $\phi_{K_0}(c_2)$ are relative cycles homologous to $\phi_{K_0}(z_1)$ and $\phi_{K_0}(z_2)$, respectively.

Note that $[z_1]$ and $[z_2]$ are both nontrivial homology classes in K . But their corresponding classes in the relative homology group may be trivial. We say a subcomplex K_0 carries a class h if h has a trivial image in the relative homology group $H_d(K, K_0)$, formally, $\phi_{K_0}^*(h) = 0 + B_d(K, K_0)$. Intuitively, this means that h disappears if we delete K_0 from K , by contracting it into a point and modding it out. The following lemma gives us more intuition behind this definition.

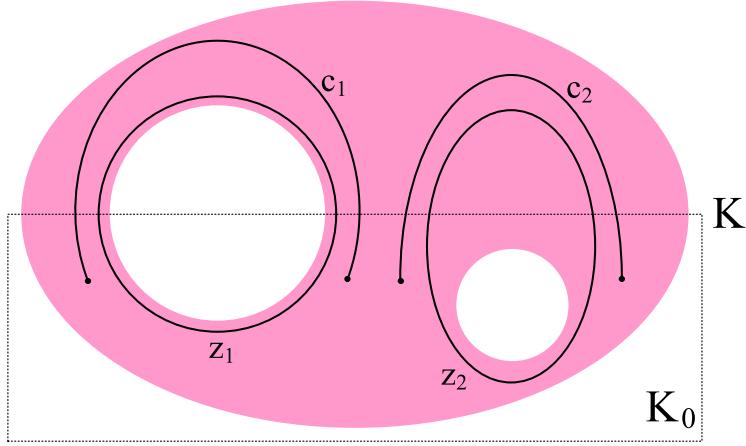


Figure 2.2: A disk with two holes, whose triangulation is K . Simplices of K lying completely in the dotted rectangle form a subcomplex K_0 . The 1-dimensional relative homology group $H_1(K, K_0)$ has dimension 1, although $H_1(K)$ has dimension 2. The nontrivial class $[z_2]$ is carried by K_0 .

Lemma 2.4.1. K_0 carries h if and only if it carries a cycle of h .

Proof. For any cycle $z \in h$, the relative chain $\phi_{K_0}(z)$ is a relative boundary if and only if there is a $(d+1)$ -chain $c' \in C_{d+1}(K)$ such that $z - \partial_{d+1}(c') \in h$ is carried by K_0 . \square

For example, in Figure 2.2, $\phi_{K_0}^*([z_1])$ is a nontrivial relative homology class, whereas $\phi_{K_0}^*([z_2])$ is trivial. We say that the class $[z_2]$ is carried by K_0 . This concept plays an important role in our definition of the size measure. Further details will be given in Section 4.1.1.

2.5 Hardness of Approximability and Strict Reductions

We will prove several optimization problems are NP-hard to approximate within any constant factor. Relevant definitions will be presented in this section.

An *NP optimization problem* Π is a four-tuple $(\mathcal{I}, \text{Sol}, m, \text{opt})$ such that

1. \mathcal{I} is the set of *instances* of Π ; any $I \in \mathcal{I}$ can be recognized in time polynomial with its size, $\text{card}(I)$;
2. given $I \in \mathcal{I}$, $\text{Sol}(I)$ denotes the set of *feasible solutions* of I ; for every $S \in \text{Sol}(I)$, $\text{card}(S)$ is polynomial in $\text{card}(I)$; given any S polynomial in $\text{card}(I)$, one can decide in polynomial time whether $S \in \text{Sol}(I)$;

3. given $I \in \mathcal{I}$ and $S \in \text{Sol}(I)$, $m(I, S)$ denotes the value of S ; m is polynomial time computable and is commonly called *objective function*;
4. $\text{opt} \in \{\min, \max\}$ indicates the *type* of optimization problem.

Given an NP optimization problem, Π , for an instance I and one of its feasible solutions, $S \in \text{Sol}(I)$, we define the *performance ratio*, $\rho_\Pi(I, S)$, as the ratio of the value $m(I, S)$ (assume $m(\cdot, \cdot) > 0$) over the value of the optimum solution, formally,

$$\rho_\Pi(I, S) = \frac{m(I, S)}{m(I, S^*(I))}$$

where $S^*(I)$ is the optimum solution of I . The quality of a polynomial approximation algorithm, A , is measured by the *approximation ratio* $\rho_A(I) = \rho_\Pi(I, A(I))$. For minimization problems, therefore, the approximation ratio is in $[1, \infty)$, while for maximization problems it is in $(0, 1]$.

An NP optimization problem Π belongs to the class APX if there exists a polynomial approximation algorithm A and a value $r \in \mathbb{Q}$ such that given any instance I of Π , $\rho_A(I) \leq r$ (resp., $\rho_A(I) \geq r$) if Π is a minimization problem (resp., a maximization problem). In such case, A is called an r -approximation algorithm of Π .

Given two problems Π_1 and Π_2 , we *reduce* Π_1 to Π_2 by providing two polynomial time computable functions f and g that satisfy

- $f : \mathcal{I}_{\Pi_1} \rightarrow \mathcal{I}_{\Pi_2}$ such that $\forall I_1 \in \mathcal{I}_{\Pi_1}$, $f(I_1) \in \mathcal{I}_{\Pi_2}$; in other words, given an instance I_1 in Π_1 , f allows to build an instance $I_2 = f(I_1)$ in Π_2 ;
- $g : \mathcal{I}_{\Pi_1} \times \text{Sol}_{\Pi_2} \rightarrow \text{Sol}_{\Pi_1}$ such that, for all $(I_1, S_2) \in \mathcal{I}_{\Pi_1} \times \text{Sol}_{\Pi_2}(f(I_1))$, $g(I_1, S_2) \in \text{Sol}_{\Pi_1}(I_1)$; in other words, starting from a solution S_2 of the instance $I_2 = f(I_1)$, g determines a solution $S_1 = g(I_1, S_2)$ of the initial instance I_1 .

When both problems are minimization problems. We say the reduction is *strict* if in addition, for any instance $I_1 \in \mathcal{I}_{\Pi_1}$ and any feasible solution of $f(I_1)$, $S_2 \in \text{Sol}_{\Pi_2}(f(I_1))$, the performance ratios satisfy

$$\rho_{\Pi_2}(f(I_1), S_2) \geq \rho_{\Pi_1}(I_1, g(I_1, S_2)). \quad (2.4)$$

Formally, we say $\Pi_1 \leq_S \Pi_2$. It is not hard to see that the strict reduction preserves the membership of APX. The following lemma will be useful for our inapproximability proof.

Lemma 2.5.1. *If $\Pi_1 \leq_S \Pi_2$ and $\Pi_1 \notin \text{APX}$, then $\Pi_2 \notin \text{APX}$.*

In other words, if Π_1 is strictly reducible to Π_2 and cannot be approximated within any constant factor, neither can Π_2 .

We conclude this chapter with other necessary concepts for the thesis.

Rank Computations of Sparse Matrices over Finite Fields. Wiedemann [86] presented a randomized algorithm to capture the rank of a sparse matrix over finite field. His method performs a binary search for the rank. For an $m \times n$ sparse matrix A , the algorithm starts with $s = \min(m, n)/2$. It tests if $s > \text{rank}(A)$ or not, and then decides whether $s = s/2$ or $s = 3s/2$. For each s , $s \times m$ and $s \times n$ matrices P and Q are randomly generated for several times. If PAQ is singular all the times, $s > \text{rank}(A)$ with high probability. The expected time of the algorithm is $O(n(\omega + n \log n) \log n)$, where n is the maximal dimension of the matrix and ω is the total number of nonzero entries in A .

The Discrete Geodesic Distance. To introduce the diameter and radius objective functions, we need a notion of distance. As we will deal with a simplicial complex K , it is most natural to introduce a discrete metric, and corresponding distance functions. We define the *discrete geodesic distance* from a vertex $p \in \text{vert}(K)$, $f_p : \text{vert}(K) \rightarrow \mathbb{R}$, as follows. Suppose each edge in K has a nonnegative weight, for any vertex $q \in \text{vert}(K)$, $f_p(q) = \text{dist}(p, q)$ is the length of the shortest path connecting p and q , in the 1-skeleton of K . We may then extend this distance function from vertices to higher dimensional simplices naturally. For any simplex $\sigma \in K$, $f_p(\sigma)$ is the maximal function value of the vertices of σ , $f_p(\sigma) = \max_{q \in \text{vert}(\sigma)} f_p(q)$. This extension has the same effect as linearly interpolating the function on the interiors of the simplices (the sublevel sets of the two extensions are homotopy equivalent). Finally, we define a *geodesic ball* B_p^r , $p \in \text{vert}(K)$, $r \geq 0$, as the subset of K , $B_p^r = \{\sigma \in K \mid f_p(\sigma) \leq r\}$. It is straightforward to show that these subsets are in fact subcomplexes.

Terminology from Coding Theory. We focus on binary linear codes and thus only use matrices over \mathbb{Z}_2 field. For consistency, we switch the roles of the row and column indices from the standard definition. Please refer to [71] for details.

Given an $m \times k$ ($m > k$) full rank matrix A , we define a *linear code* as the k -dimensional column space of A , namely, $\text{span}(A)$. Each element of the linear code is called a *codeword*. This matrix is called the *generator matrix* as it is a basis of the linear code. By slightly abusing notation, we call a full rank matrix A^\perp the *parity-check matrix* if its nullspace is the linear code. Given a generator matrix A , A^\perp may be computed in polynomial time. The dimension of A^\perp is $(m - k) \times m$.

CHAPTER 3

LITERATURE REVIEWS

We first generally review topology related computations in the literature. Second, we review works related to homology measurement and localization.

3.1 Topology Related Computations

3.1.1 Computing the Homology Group

The first, and the most fundamental problem is computing the structure of the homology group, including the Betti number and the torsion coefficients. For a simplicial homology whose coefficient ring $R = \mathbb{Z}$, the Betti number and the torsion coefficients are all important. The torsion coefficients disappear when the coefficient ring R is a field, e.g. the real field, \mathbb{R} , the rational field, \mathbb{Q} , as well as the finite field \mathbb{Z}_p with p a prime number. In this section, we present research in computing these structural information.

The Betti number and the torsion coefficients can be revealed by the Smith normal form of the integer boundary matrices, which can be computed by a straightforward matrix reduction algorithm [75]. Kannan and Bachem [66] devised the first polynomial algorithm to compute the Smith normal form. Since then, this problem has drawn attention from multiple researchers. The fastest known deterministic algorithm is by Storjohann [83, 84] requiring $O(n^4)$ time and $O(n^2)$ space. The fastest known probabilistic algorithm is by Eberly et al. [43] (of the Monte Carlo type) requiring $O(n^3)$ expected time, $O(n^{3.5})$ worst case time, and $O(n^2)$ space.

There are some nice properties in the boundary matrix, which have been utilized to devise faster Smith normal form computation algorithms specifically for computing homology. A probabilistic algorithm of the Monte Carlo type for sparse matrices is given by Giesbrecht [59]. Dumas et al. [41, 40] devised algorithms which run quickly in practice, using not only the sparsity but also the valence, which is the trailing coefficient of the minimal polynomial of the symmetrized but unpreconditioned matrix $\partial\partial^T$.¹

If the torsion coefficients disappear or only the Betti number is needed, computing ranks of the boundary matrices is all that is required [73, 39, 86]. However, when the coefficient ring is not a finite field, rank computations may involve precision problem and require the use of a multiple precision package which is computationally expensive.

Friedman [56] computed the Betti number of dimension d , β_d , by computing the dimension of the kernel of the d -dimensional combinatorial Laplacian matrix,

$$\Delta_d = \partial_d \partial_d^T + \partial_{d+1}^T \partial_{d+1},$$

¹The valence is proved to be small in many boundary matrices, and thus believed to be generally small for boundary matrices.

which is symmetric and sparse. According to Hodge theory, the kernel of Δ_d , which is the space spanned by the harmonic d -forms, is isomorphic to the d -dimensional homology group, and thus, has the dimension β_d . Friedman used the power method to compute β_d , the multiplicity of the zero eigenvalue of Δ_d , with high confidence. The algorithm runs quickly in practice and relies on a unknown ratio, i.e. the smallest nonzero eigenvalue over the largest eigenvalue of Δ_d .

Delfinado and Edelsbrunner [34] devised an almost linear algorithm which computes the Betti number of a simplicial complex embedded in S^3 . The algorithm starts with an empty simplicial complex whose Betti numbers in all dimensions vanish. These Betti numbers are updated while simplices are being added into the simplicial complex one by one, until the whole input simplicial complex is in. For each d -simplex, the algorithm checks whether it belongs to a d -cycle in the accumulated complex or not, and accordingly, increases β_d by one or decreases β_{d-1} by one. The algorithm runs in $O(n\alpha(n))$ time with $O(n)$ storage, where $\alpha(n)$ is the extremely slowly growing inverse of Ackermann's function. This algorithm works quickly in practice. However, the d -cycle checking procedure is based on low dimensional topological inference, and thus, restricts the algorithm to simplicial complexes embedded in S^3 .

Kaczynski et al. [64, 65] used a reduction algorithm to compute the Betti number of the homology group of the cubical complex, in which a d -simplex corresponds to a d -cube. The algorithm repeatedly removes one pair of coinciding simplices with two consecutive dimensions until all boundary maps vanish. The chain groups left are isomorphic to the homology groups. The Betti number of dimension d is the dimension of the d -dimensional chain group, more specifically, the number of d -simplices left. This method has practical advantages in studying dynamical systems and can be extended to compute the homology of continuous maps.

3.1.2 Persistent Homology

The idea of persistent homology is rooted in the concept of alpha shapes first developed by Edelsbrunner et al. [47, 50].² Roughly speaking, an alpha shape is a union of balls centered at a given set of points, S , with radius r , formally, $\bigcup_{p \in S} B_p^r$. Insight into the structure of the point set is revealed by the evolution of the topology of the alpha shape as the parameter r increases from $-\infty$ to $+\infty$.

The concepts and algorithms of persistent homology were first introduced by Edelsbrunner et al. [48, 91]. The persistence diagram was further studied and proved to be stable: for a same topological space, the Hausdorff distance and even the bottleneck distance between two persistence diagrams are bounded by the L_∞ distance between the corresponding filter functions [25]. A linear algorithm was devised for updating persistences while changing the filtration [28]. The theory was extended such that essential homology classes also have finite persistences [26]. The extended persistence diagram is proved to carry some nice properties, like duality and sym-

²For completeness, we also refer to two other independent works on persistence homology, namely, Frosini et al. [57, 9] and Robins [81].

metry. The proof is based on Poincaré and Lefschetz duality and is restricted to d -manifolds and their triangulations. Persistences of relative homology and cohomology were also studied [80, 31].

Zomorodian and Carlsson [92] studied the algebraic structure of persistent homology. They formalized the sequence of homology groups in all dimensions and group homomorphisms induced by chain inclusion as an algebraic object, namely, the *persistent module* of finite type, $\mathcal{M} = \{H_*(K_i), F_*^{i,i+1}\}_{i \geq 0}$ over the coefficient ring R . Using Artin-Rees theory in commutative algebra, they proved that \mathcal{M} corresponds to a finitely generated non-negative graded module over $R[t]$.³ This result implies that persistent homology has no simple structure when R is not a field, e.g. $R = \mathbb{Z}$. On the other hand, when R is a field, e.g. \mathbb{R}, \mathbb{Q} or \mathbb{Z}_p , the structure of \mathcal{M} can be completely described by the birth and death times of persistent homology classes. An efficient matrix reduction algorithm was suggested to compute this structure. In a recent paper [14], the authors extended persistent homology by extending the filtration to a multi-filtration, namely, a family of subcomplexes parameterized along multiple geometric dimensions. They proved that this extension cannot retain the structural properties of the 1-dimensional persistent homology.

Persistent homology has been applied to various problems. In terms of theoretical problems, it was used to localize homology classes [93] and prove geometric inequalities about curvatures of curves and surfaces [24]. In terms of applications, persistent homology was used to study problems in biology such as protein docking [1, 85], protein folding [28] and membrane fusion [67]. It was also used in inferring homology from manifold sampling [25], analyzing high dimensional data [11, 13], generating shape signatures [12, 10] and studying sensor networks [33].

3.1.3 Discrete Morse Theory

In the mathematics literature, Morse theory [72] is closely related to persistent homology. Given a smooth manifold and a smooth function with a mild assumption of non-degeneracy, Morse theory studies topological changes of the sub-level set $f^{-1}(-\infty, t]$ as t increases. The result includes not only the time t 's when changes happen (*critical values*), but also the locations where changes happen (*critical points*). Edelsbrunner et al. [46, 63] migrated Morse theory into the discrete framework, more specifically, piecewise linear manifolds (e.g. triangulations) and piecewise linear functions defined on them (e.g. height functions). Difficulties of the migration include unavoidable non-smoothness and degenerate critical points, which were solved by simulation of differentiability and local triangulation perturbation, respectively. Important facts in the original Morse theory like the weak and strong Morse inequalities are proved in the discrete version. The discrete analog of the Morse-Smale complex was computed and provides hierarchical segmentations of the manifolds which has deep relationships with both geometry and topology. These works offer tools for studying shapes [44, 76] and functions [49, 63]. So far, they have not yet been extended to high-dimensional manifolds.

³ $R[t]$ is the ring of polynomials in the variable t with coefficients in R .

For completeness, we also refer to other approaches in migrating geometrical concepts in the mathematics literature into discrete frameworks [54, 35].

3.1.4 Topologically Faithful Manifold Reconstruction

A tangentially related topic is given samples of a manifold, how to topologically faithfully reconstruct the manifold, namely, how to construct a triangulation whose underlying space is homeomorphic to the manifold. This involves two subproblems: (a) what sampling condition can guarantee the correct reconstruction, (b) how to reconstruct the manifold using a good sampling. Amenda et al. [2, 3] provided the first sampling condition and corresponding reconstruction algorithms for smooth manifolds embedded in \mathbb{R}^3 . These works have been recently extended to general dimensions and noisy samples [74, 36, 23]. It was proved that under certain sampling conditions, the alpha shape of the sampling has the same homotopy type of the smooth manifold [77, 21]. Sampling conditions and reconstruction algorithms are also provided for non-smooth manifolds [20, 25, 19]. Researchers are also interested in inferring topology of the manifold with a small portion of samples [32, 60, 8, 6].

3.2 Homology Measurement and Localization

Persistent homology [?] is designed to track the lifetimes of homological features over the course of a filtration of a topological space. At first blush, it might seem that the powerful techniques of this theory are ideally suited to solving the problems we have set out. However, due to their somewhat different motivation, these techniques do not quite yield a solution. There are two reasons for this. First, the persistence of a feature depends not only on the space in which the feature lives, but also on the filtering function chosen. In the absence of a geometrically meaningful filter, it is not clear whether the persistence of a feature is a meaningful representation of its size. Second, and more importantly, the persistence only gives information for homology classes which ultimately die; for classes which are intrinsically part of the topological space, and which thus never die, the persistence is infinite. However, it is precisely these essential (or non-persistent) classes that we care about.

In more recent work, Cohen-Steiner et al.[26] have extended persistent homology in such a way that essential homology classes also have finite persistences. However, the persistences thus computed still depend on the filter function, and furthermore, do not always seem to agree with an intuitive notion of size. See Figure 3.1.

Zomorodian and Carlsson [93] take a different approach to solving the localization problem. Their method starts with a topological space and a cover, a set of spaces whose union contains the original space. A blowup complex is built up which contains homology classes of all the spaces in the cover. The authors then use persistent homology to identify homology classes in the blowup complex which correspond to a same homology class in the given topological space. The persistent

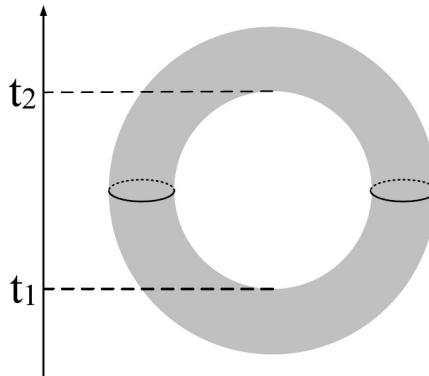


Figure 3.1: Computing the extended persistent homology of a torus using the height function as the filter function. The (birth,death time) pairs of the two 1-dimensional homology classes are (t_1, t_2) and (t_2, t_1) , respectively. The persistences are not consistent with our intuition of their sizes.

homology algorithm produces a complete set of generators for the relevant homology group, which forms a basis for the group. However, both the quality of the generators and the complexity of the algorithm depend strongly on the choice of cover; there is, as yet, no suggestion of a canonical cover.

The problem is relative easier when the pertinent homology is of 1-dimension. Researchers have been interested in localizing 1-dimensional homology classes with the shortest representative 1-cycle. Using Dijkstra's shortest path algorithm, Erickson and Whittlesey [52] computed the *shortest homology basis*, namely, the 1-dimensional homology cycle basis whose elements have the minimal total length. Their algorithm computes the basis in a greedy manner and takes $O(n^2 \log n + n^2\beta_1 + n\beta_1^3)$ or $O(n^2\beta_1 + n\beta_1^3)$ if β_1 is nearly linear in n , where n is the size of the given simplicial complex. The authors also showed how the idea carries over to finding the optimal generators of the first fundamental group, though the proof is considerably harder in this case.

This polynomial basis computation algorithm cannot localize an arbitrarily given 1-dimensional class. To fill this void, Chambers et al. [18] devised an algorithm to localize a given class. Their method precomputes the shortest representative cycles of all $2^{\beta_1} - 1$ nontrivial classes, and thus, is exponential to the 1-dimensional Betti number. It has been demonstrated that when $\beta_1 = \Theta(n)$, localizing a given 1-dimensional class with its shortest cycle is NP-hard, no matter the topological space is a manifold with [22] or without boundary [18]. However, Chambers et al. [17] showed that this is not the case for 1-dimensional homology over real or integer coefficients.

For completeness, we refer to some related works which compute a single or a set of non-trivial cycles satisfying certain topological and geometrical restrictions on 2-manifolds [51, 70, 16, 29].

Due to the difficulties in localizing with the shortest cycle, researchers focused on other criteria or heuristics. Some researchers computed 1-dimensional cycles closely related to handles which are much more meaningful in low dimensional applications such as graphics and CAD. Guskov and Wood [62, 89] detected small handles of a 2-manifold using the Reeb graph of the manifold.⁴

Given a 2-manifold, M , embedded in S^3 , Dey et al. [37] classified 1-cycles on each handle into *handle cycles* and *tunnel cycles*. A handle cycle (resp. tunnel cycle) is a 1-cycle which is a boundary in the component of S^3 bounded inside M (resp. outside M). These handle-related cycles can be generated by computing the deformation retractions of the two components of the embedding space bounded by the given 2-manifold. A recent extension [38] improved their result based on geometric heuristics and the persistent homology. Their work helps identifying handles which are meaningful in real applications. (See Figure 3.2).

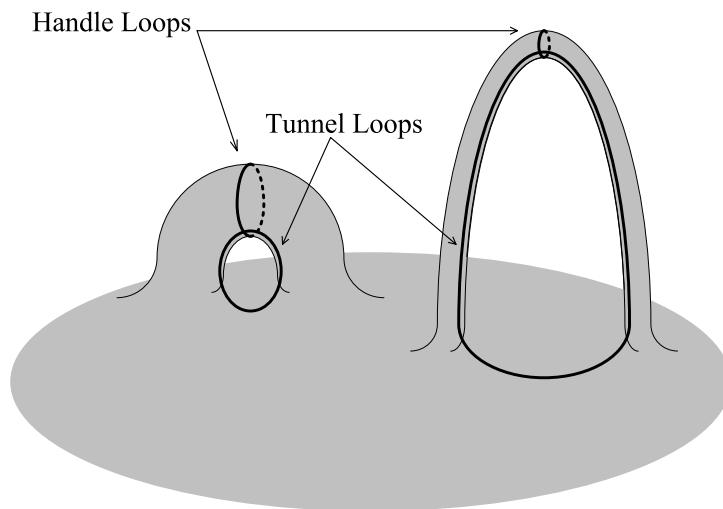


Figure 3.2: 2 handles and their tunnel and handle cycles in a 2-manifold.

⁴We use the idea of growing geodesic balls. A similar idea has been used in [62, 89]. However, this latter work depends on low dimensional geometric reasoning, and hence is restricted to 1-dimensional homology classes in 2-manifold.

CHAPTER 4

MEASURING

In this chapter, we provide a technique for ranking homology classes according to their importance. Specifically, we solve the two problems mentioned in Chapter 1 by formally defining (1) a meaningful size measure for homology classes that is computable in arbitrary dimension; and (2) an optimal homology basis which distinguishes large classes from small ones effectively. We provide algorithms and stability proof for these definitions.

4.1 Definitions

4.1.1 Measuring the Size of a Homology Class

Using relative homology, we define the size of a homology class as follows. Given a simplicial complex K , assume we are given a collection of subcomplexes $\mathcal{L} = \{L \subseteq K\}$. Furthermore, each of these subcomplexes is endowed with a size. In this case, we define the size of a homology class h as the size of the smallest L carrying h (assuming one such L exists, which can be guaranteed if \mathcal{L} is properly chosen).

Definition 4.1.1. *The size of a class h , $S(h)$, is the size of the smallest measurable subcomplex carrying h , formally,*

$$S(h) = \min_{L \in \mathcal{L}} \text{size}(L) \quad s.t. \quad \phi_L^*(h) = \mathbf{B}_d(K, L).$$

In this chapter, we take \mathcal{L} to be the set of discrete geodesic balls, $\mathcal{L} = \{B_p^r \mid p \in \text{vert}(K), r \geq 0\}$. The size of a geodesic ball is naturally its radius r . The smallest geodesic ball carrying h is denoted as $B_{\min}(h)$ for convenience, whose radius is $S(h)$. See Figure 4.1 for two examples. Note that these geodesic balls may not look like Euclidean balls in the embedding space.

4.1.2 The Optimal Homology Basis

For the d -dimensional \mathbb{Z}_2 homology group whose dimension (Betti number) is β_d , there are $2^{\beta_d} - 1$ nontrivial homology classes. However, we only need β_d of them to form a basis. The basis should be chosen wisely so that we can easily distinguish important homology classes from noise. In Figure 1.2, there are $2^3 - 1 = 7$ nontrivial homology classes; we need three of them to form a basis. We would prefer to choose $\{[z_1], [z_2], [z_3]\}$ as a basis, rather than $\{[z_1] + [z_2] + [z_3], [z_2] + [z_3], [z_3]\}$. The former indicates that there is one big cycle in the topological space, whereas the latter gives the impression of three large classes.

In keeping with this intuition, the *optimal homology basis* is defined as follows.

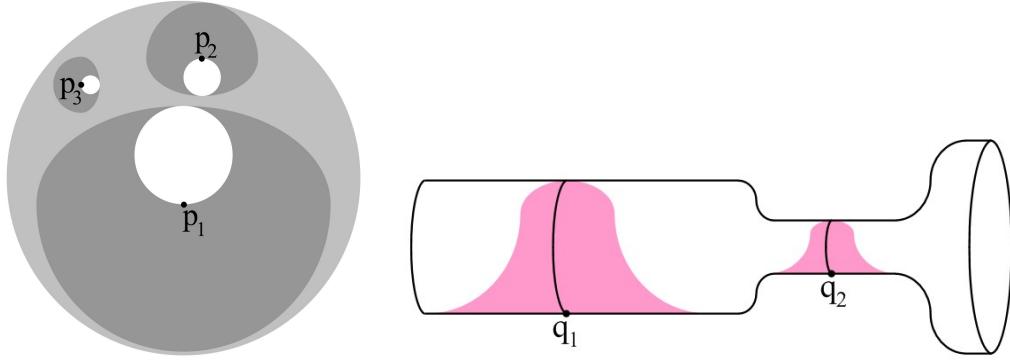


Figure 4.1: Left: On a disk with three holes, the three shaded regions are the three smallest geodesic balls (centered at p_1 , p_2 and p_3) carrying the three corresponding classes.
Right: On a tube, both balls (centered at p_1 and p_2) carry the only nontrivial class. The one centered at p_2 is the smallest and provides measurement.

Definition 4.1.2. *The optimal homology basis is the basis for the homology group whose elements' size have the minimal sum, formally,*

$$\mathcal{H}_d = \underset{\{h_1, \dots, h_{\beta_d}\}}{\operatorname{argmin}} \sum_{i=1}^{\beta_d} S(h_i), \text{ s.t. } \dim(\{h_1, \dots, h_{\beta_d}\}) = \beta_d.$$

This definition guarantees that large homology classes appear as few times as possible in the optimal homology basis. In Figure 1.2, the optimal basis will be $\{[z_1], [z_2], [z_3]\}$, which has only one large class.

This definition uses L_1 -norm on the vector of sizes. Since all class sizes are nonnegative, and further, since the problem has a matroid structure (to be demonstrated in the next section), it will turn out that we can use any L_p -norm in the definition and still get the same optimal homology basis. An exception, however, is the L_∞ -norm. In this case, there may be many different optimal bases. The optimal basis defined using L_1 -norm is one of them.

For each class in the basis, we need a cycle representing it. According to Lemma 2.4.1, $B_{\min}(h)$, the smallest geodesic ball carrying h , carries at least one cycle of h . We represent h with these cycles, denoted as *localized cycles* of h . This is a fair choice because it is consistent with the size measure of h and it is computable in polynomial time.

Please note that the localized cycle may not be the simplest one. The cycle may wiggle a lot inside the geodesic ball, $B_{\min}(h)$. Further discussions about homology localization can be found in Chapter 5.

4.2 The Algorithm

In this section, we introduce an algorithm to compute the optimal homology basis as defined in Definition 4.1.2. For each class in the basis, we measure its size, and represent it with one of its localized cycles.

We first introduce an algorithm to compute the smallest homology class, namely, **Measure-Smallest**(K). Based on this procedure, we provide the algorithm **Measure-All**(K), which computes the optimal homology basis. The algorithm takes $O(\beta_d n^4)$ time, where β_d is the Betti number for d -dimensional homology group and n is the cardinality of the input simplicial complex K . Please note that in the rest of the chapter, we assume d , the dimension of the relevant homology group, is given.

4.2.1 Computing the Smallest Homology Class

The procedure **Measure-Smallest**(K) measures and localizes the smallest non-trivial homology class, namely, the one with the smallest size,

$$h_{min} = \operatorname{argmin}_{h \in H_d(K) : h \neq B_d(K)} S(h).$$

The output of this procedure will be a pair (S_{min}, z_{min}) , namely, the size and a localized cycle of h_{min} . According to the definitions, this pair is determined by the smallest geodesic ball carrying h_{min} , namely, $B_{min}(h_{min})$.

It is straightforward to see that the ball $B_{min}(h_{min})$ is also the smallest geodesic ball carrying any nontrivial homology class of K . It can be computed by computing $B_p^{r(p)}$ for all vertices p , where $B_p^{r(p)}$ is the smallest geodesic ball centered at p which carries any nontrivial homology class. When all the $B_p^{r(p)}$'s are computed, we compare their radii, $r(p)$'s, and pick the smallest ball as $B_{min}(h_{min})$.

For each vertex p , we compute $B_p^{r(p)}$ by applying the persistent homology algorithm to K with the discrete geodesic distance from p , f_p , as the filter function. Note that a geodesic ball B_p^r is the sublevel set $f_p^{-1}(-\infty, r] \subseteq K$. Nontrivial homology classes of K are essential homology classes in the persistent homology algorithm. (In the rest of this chapter, we may use “essential homology classes” and “nontrivial homology classes of K ” interchangeable.) Therefore, the birth time of the first essential homology class, namely, the filter function value of the very first d -simplex that is positive and paired with $+\infty$, is $r(p)$, and the subcomplex $f_p^{-1}(-\infty, r(p)]$ is $B_p^{r(p)}$.

Once we determine $B_{min}(h_{min})$, whose center is denoted as p_{min} , the size S_{min} is the radius $r(p_{min})$. A localized cycle can be decided by the persistent homology algorithm with the filter function $f_{p_{min}}$. Recall that the matrix V in the persistence reduction $R = DV$ provides cycles representing essential classes. The localized cycle z_{min} is the column of V corresponding to the very first d -simplex that is positive and paired with $+\infty$. The reason is it represents the youngest essential class, which is h_{min} . Plus it is carried by $B_{min}(h_{min})$.

4.2.2 Computing the Optimal Homology Basis

In this section, we present the algorithm for computing the optimal homology basis defined in Definition 4.1.2, namely, \mathcal{H}_d . We first show that the optimal homology basis can be computed in a greedy manner. Second, we introduce an efficient greedy algorithm.

As has been noted, over the \mathbb{Z}_2 field, the homology group is a vector space. It, together with the family of its linearly independent subsets, form a vector matroid. Using the size of homology classes as a weight function, we have a weighted matroid. Matroid theory [30, 78] suggests a greedy method to compute the optimal homology basis as follows.

For convenience, let H be the set of nontrivial d -dimensional homology classes (i.e. the homology group minus the trivial class). Denote $seq(H) = (h_1, h_2, \dots, h_{(2^{\beta_d}-1)})$ as the sequence of all the classes of H sorted in the monotonically increasing order according to size, formally, $S(h_i) \leq S(h_{i+1})$ for all i . Repeatedly compute the smallest class in $seq(H)$ and pick each one which is linearly independent of those we have already picked, until β_d are picked. The collected β_d classes $\{h_{i_1}, h_{i_2}, \dots, h_{i_{\beta_d}}\}$ form the optimal homology basis \mathcal{H}_d . (Note that the h 's are ordered by size, i.e. $S(h_{i_k}) \leq S(h_{i_{k+1}})$.)

However, this naive method may be exponential in β_d . For example, we may have to compute all the linear combinations of $\{h_{i_1}, h_{i_2}, \dots, h_{i_{(\beta_d-1)}}\}$ before we find $h_{i_{\beta_d}}$. Next, we present our greedy algorithm which is polynomial.

Instead of computing the smallest classes in $seq(H)$ one by one, our algorithm uses a sealing technique and takes time polynomial in β_d and n . Intuitively, when the smallest l classes in \mathcal{H}_d are picked, we make them trivial by adding new cells to the given complex. In the augmented complex, any linear combination of these picked classes becomes trivial, and the smallest nontrivial class is the $(l+1)$ 'th one in \mathcal{H}_d .

The algorithm starts by measuring and localizing the smallest homology class of the given simplicial complex K (using the procedure **Measure-Smallest(K)** introduced in Section 4.2.1), which is also the first class we choose for \mathcal{H}_d . We make this class trivial by sealing one of its cycles – i.e. the localized cycle we computed – with a new cell. Next, we measure and localize the smallest homology class of the augmented complex K' . This class is the second smallest homology class in \mathcal{H}_d . We make this class trivial again and proceed for the third smallest class in \mathcal{H}_d . This process is repeated for β_d rounds, yielding \mathcal{H}_d .

We make a homology class trivial by sealing the class's localized cycle, which we have computed. To seal this cycle z , we add a new $(d+1)$ -cell whose boundary is exactly this cycle. In Figure 4.2, a 1-cycle with four edges, z_1 , is sealed up with one new 2-cell. Please note that the new cell is not a simplex and the augmented complex K' is a cell complex, not a simplicial complex.

It is essential to make sure the new cell does not influence our measurement. We assign the new cell $+\infty$ filter function values, formally, $f_p(\sigma) = +\infty$ for all $p \in \text{vert}(K)$ and $\sigma \in K' \setminus K$.

The algorithm is illustrated in Figure 4.2. Assuming unit edge lengths, the 4-edge cycle, z_1 , and the 8-edge cycle, z_2 , are the localized cycles of the smallest and the second smallest homology classes ($S([z_1]) = 2, S([z_2]) = 4$). The nonbounding cycle $z_3 = z_1 + z_2$ corresponds to the largest nontrivial homology class $[z_3] = [z_1] + [z_2]$ ($S([z_3]) = 5$). After the first round, we choose $[z_1]$ as the smallest class in \mathcal{H}_1 . Next, we destroy $[z_1]$ by sealing z_1 , which yields the augmented complex K' . This time, we choose $[z_2]$, giving $\mathcal{H}_1 = \{[z_1], [z_2]\}$.

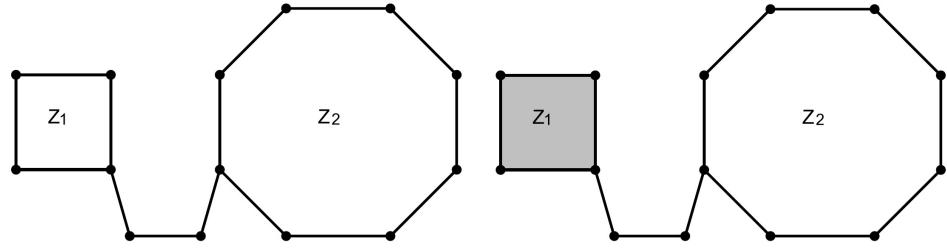


Figure 4.2: Left: the original complex K .
Right: the augmented complex K' after destroying the smallest class, $[z_1]$.

Theorem 4.2.1. *The procedure Measure-All(K) computes \mathcal{H}_d .*

Proof. We prove the theorem by showing that the sealing technique produces the same result as the naive greedy algorithm, namely, $\mathcal{H}_d = \{h_{i_1}, h_{i_2}, \dots, h_{\beta_d}\}$, assuming the classes are sorted according to size, $S(h_{i_k}) \leq S(h_{i_{k+1}})$. We show that for any $l \in [0, \beta_d]$, after computing and sealing the first l classes of \mathcal{H}_d , i.e. $\{h_{i_1}, \dots, h_{i_l}\}$, the next class we choose is exactly $h_{i_{l+1}}$. In other words, the localized cycle and size of the smallest class of the augmented complex K^l are equal to that of $h_{i_{l+1}}$.

First, any class between h_{i_l} and $h_{i_{l+1}}$ in $\text{seq}(H)$ will not be chosen. Any such class h_j is linearly dependent on classes that have already been chosen, namely, $\{h_{i_1}, \dots, h_{i_l}\}$. Since these classes have been sealed up, a cycle of h_j is a boundary in K^l . Thus, h_j cannot be chosen.

Second, according to algebra, one new cell can only destroy one class. Therefore, for any class in $\text{seq}(H)$ that is not linearly dependent on $\{h_{i_1}, \dots, h_{i_l}\}$, it is nontrivial in K^l .

Third, the smallest class of K^l , $h_{\min}(K^l)$, corresponds to $h_{i_{l+1}}$: any new simplex belonging to $K^l \setminus K$ will not change the computation of the geodesic balls B_p^r with finite radius r , and thus will change neither the size measurement nor the localization. Thus, the $h_{\min}(K^l)$ computed by the sealing technique is identical to $h_{i_{l+1}}$ computed by the naive greedy method, in terms of the size and the localized cycle. \square

4.2.3 Complexity of the Non-Refined Algorithm

Throughout the algorithm, at most β_d new cells are added. The size of the augmented cell complex K' is $O(n + \beta_d)$. The procedure Measure-All(K) runs the

procedure **Measure-Smallest** β_d times with K' as input. The procedure **Measure-Smallest** runs the persistent homology algorithm on K' using filter function f_p for each vertex of the original complex, K , and thus takes $O(n(n + \beta_d)^3) = O(n^4)$ time. In total, it takes $O(\beta_d n^4)$ time to compute the optimal basis.

4.3 An Improvement Using Finite Field Linear Algebra

In this section, we present an improvement on the algorithm presented in the previous section, more specifically, an improvement on computing the smallest geodesic ball carrying any nontrivial class (the procedure **Measure-Smallest**). The idea is based on the finite field linear algebra behind the homology.

In Section 4.3.1, we observe that for neighboring vertices, p_1 and p_2 , the birth times of the first essential homology class using f_{p_1} and f_{p_2} as filter functions are close (Theorem 4.3.2). This observation suggests that for each p , instead of computing $B_p^{r(p)}$, we may just test whether the geodesic ball centered at p with a certain radius carries any essential homology class. In Section 4.3.2, with some algebraic insight, we reduce the problem of testing whether a geodesic ball carries any essential homology class to the problem of comparing dimensions of two vector spaces. Furthermore, we use Lemma 4.3.3 to reduce the problem to rank computations of sparse matrices on the \mathbb{Z}_2 field, for which we have ready tools from the literature. In Section 4.3.3, we conclude with detailed complexity analysis.

In this section, we will consider all edges to have weights of 1, for simplicity of exposition. However, please note that it is possible to generalize all results to deal with general (real) edge weights, though the algorithm becomes somewhat messier. We also assume that K has a single component; multiple components can be accommodated with a simple modification.

Complexity. In doing so, we improve the complexity to $O(\beta_d n^3 \log^2 n)$. More detailed complexity analysis is provided in Section 4.3.3.

Remark 4.3.1. Cohen-Steiner et al./[28] provided a linear algorithm to maintain the persistences while changing the filter function. However, this algorithm is not directly applicable in our context. The reason is that it takes $O(n)$ time to update the persistences for a transposition in the simplex-ordering. In our case, even for filter functions of two neighboring vertices, often it takes $O(n^2)$ transpositions to transform one simplex-ordering into the other. See Figure 4.3 for example. Therefore, updating the persistences while changing the filter function takes $O(n^2) \times O(n) = O(n^3)$ time. This is the same amount of time it would take to compute the persistences from scratch.

4.3.1 Observation: Neighboring Vertices Have Similar Geodesic Distance Functions

Since the filter functions of two neighboring vertices, f_{p_1} and f_{p_2} , are close to each other, the birth times of the first nonbounding cycles in both filters are close

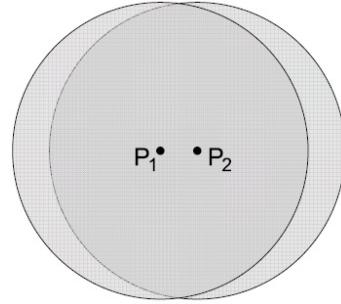


Figure 4.3: On the plane, when we change the filter function from f_{p_1} to f_{p_2} , in order to update the simplex-ordering, we should switch the order of the two blocks of simplices $B_{p_1}^r \setminus B_{p_2}^r$ and $B_{p_2}^r \setminus B_{p_1}^r$, in which $B_{p_1}^r$ and $B_{p_2}^r$ are geodesic balls centered at p_1 and p_2 with a same radius, r . When r is big, these two blocks can have $O(n)$ simplices. We then need $O(n^2)$ transpositions to update the simplex-ordering.

as well. This leads to Theorem 4.3.2. A simple proof is provided.

Theorem 4.3.2. *If two vertices p_1 and p_2 are neighbors, the birth times of the first nonbounding cycles for filter functions f_{p_1} and f_{p_2} differ by no more than 1.*

Proof. p_1 and p_2 are neighbors implies that for any point q ,

$$f_{p_2}(q) \leq f_{p_2}(p_1) + f_{p_1}(q) = 1 + f_{p_1}(q),$$

in which the inequality follows the triangular inequality. Therefore, $B_{p_1}^{r(p_1)}$ is a subset of $B_{p_2}^{r(p_1)+1}$. The former carries nonbounding cycles implies that the latter does too, and thus $r(p_2) \leq r(p_1) + 1$. Similarly, we have $r(p_1) \leq r(p_2) + 1$. \square

This theorem suggests a way to avoid computing $B_p^{r(p)}$ for all $p \in \text{vert}(K)$ in the procedure **Measure-Smallest**. Since our objective is to find the minimum of the $r(p)$'s, we do a breadth-first search through all the vertices with global variables r_{min} and p_{min} recording the smallest $r(p)$ we have found and its corresponding center p , respectively. We start by applying the persistent homology algorithm on K with filter function f_{p_0} , where p_0 is an arbitrary vertex of K . Initialize r_{min} as the birth time of the first nonbounding cycle of K , $r(p_0)$, and p_{min} as p_0 . Next, we do a breadth-first search through the rest vertices. For each vertex $p_i, i \neq 0$, there is a neighbor p_j we have visited (the parent vertex of p_i in the breath-first search tree). We know that $r(p_j) \geq r_{min}$ and $r(p_i) \geq r(p_j) - 1$ (Theorem 4.3.2). Therefore, $r(p_i) \geq r_{min} - 1$. We only need to test whether the geodesic ball $B_{p_i}^{r_{min}-1}$ carries any nonbounding cycle of K . If so, r_{min} is decremented by one, and p_{min} is updated to p_i . After all vertices are visited, p_{min} and r_{min} give us the ball we want.

However, testing whether the subcomplex $B_{p_i}^{r_{min}-1}$ carries any nonbounding cycle of K is not as easy as computing nonbounding cycles of the subcomplex. A

nonbounding cycle of $B_{p_i}^{r_{min}-1}$ may not be nonbounding in K as we require. For example, in Figure 4.4, the simplicial complex K is a torus with a tail. The shaded geodesic ball in the first figure does not carry any nonbounding cycle of K , although it carries its own nonbounding cycles. The geodesic ball in the second figure is the one that carries nonbounding cycles of K . Therefore, we need algebraic tools to distinguish nonbounding cycles of K from those of the subcomplex $B_{p_i}^{r_{min}-1}$.

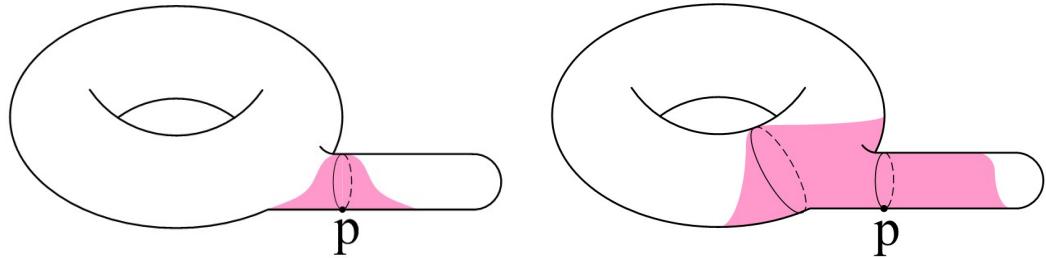


Figure 4.4: Only the ball in the second figure carries nonbounding cycles of K , although in both figures the balls have nontrivial topology.

4.3.2 Procedure Contain-Nonbounding-Cycle: Testing Whether a Subcomplex Carries Nonbounding Cycles of K

In this section, we present the procedure for testing whether a subcomplex K_0 carries any nonbounding cycle of K . A chain in K_0 is a cycle if and only if it is a cycle of K . However, solely from K_0 , we are not able to tell whether a cycle carried by K_0 bounds or not in K . Instead, we write the set of cycles of K carried by K_0 , $Z_d^{K_0}(K)$, and the set of boundaries of K carried by K_0 , $B_d^{K_0}(K)$, as sets of linear combinations with certain constraints. Consequently, we are able to test whether any cycle carried by K_0 is nonbounding in K by comparing the dimensions of $Z_d^{K_0}(K)$ and $B_d^{K_0}(K)$. Lemma 4.3.3 shows that these dimensions can be computed by rank computations of sparse matrices.⁵

The set of boundaries and the set of cycles of K carried by K_0 are

$$\begin{aligned} B_d^{K_0}(K) &= B_d(K) \cap C_d(K_0) \quad \text{and} \\ Z_d^{K_0}(K) &= Z_d(K) \cap C_d(K_0), \end{aligned}$$

respectively. They are both vector spaces and the former is a subspace of the latter. It is not hard to show that the subcomplex K_0 carries nonbounding cycles of K if and only if the dimensions of these two vector spaces are different. We now express them as linear combinations with certain constraints such that we can compute their dimensions using algebraic tools.

⁵To some extent, the idea of this section is similar in spirit to [27]. However, note that the two works developed independently.

Let $\hat{H}_d = [z_1, \dots, z_{\beta_d}]$ be the matrix whose column vectors form a d -dimensional homology cycle basis. The boundary group and the cycle group of K are column spaces of the matrices ∂_{d+1} and $\hat{Z}_d = [\partial_{d+1}, \hat{H}_d]$, respectively.

$C_d(K_0)$ corresponds to the set of vectors each of whose i -th entry is zero for any simplex $\sigma_i \notin K_0$. We write $Z_d^{K_0}(K)$ and $B_d^{K_0}(K)$ as elements of $Z_d(K)$ and $B_d(K)$ whose i -th entries are zero. Consequently, we can write them as linear combinations with certain constraints,

$$\begin{aligned} B_d^{K_0}(K) &= \{\partial_{d+1}\gamma \mid \gamma \in \mathbb{Z}_2^{n(d+1)}, \partial_{d+1}^i\gamma = 0 \quad \forall \sigma_i \notin K_0\} \\ Z_d^{K_0}(K) &= \{\hat{Z}_d\gamma \mid \gamma \in \mathbb{Z}_2^{\beta_d+n(d+1)}, \hat{Z}_d^i\gamma = 0 \quad \forall \sigma_i \notin K_0\} \end{aligned}$$

where ∂_{d+1}^i and \hat{Z}_d^i are the i -th rows of the matrices ∂_{d+1} and \hat{Z}_d , respectively. Here $n_{(d+1)}$ is the number of $(d+1)$ -simplices in K , and thus the number of columns of ∂_{d+1} .

With the following lemma, we can compute the dimensions of these two vector spaces $Z_d^{K_0}(K)$ and $B_d^{K_0}(K)$ by matrix rank computations. The proof is based on finite field linear algebra.

Lemma 4.3.3. *For any matrix $A = \begin{bmatrix} A_1 \\ A_2 \end{bmatrix}$, $\dim(\{A\gamma \mid A_2\gamma = 0\}) = \text{rank}(A) - \text{rank}(A_2)$*

Proof. Denote $P = \text{span } A = \{A\gamma\}$. $P_1 = \{A\gamma \mid A_2\gamma = 0\}$ is its subspace. The quotient vector space P/P_1 is isomorphic to $P_2 = \text{span}(A_2) = \{A_2\gamma\}$. Therefore, we have

$$\begin{aligned} \dim(P_1) &= \dim(P) - \dim(P/P_1) \\ &= \dim(P) - \dim(P_2) \\ &= \text{rank}(A) - \text{rank}(A_2) \end{aligned}$$

□

It is trivial to see that the order of the rows in these matrices does not interfere with the correctness of the theorem. The matrix A_2 can be a certain subset of the rows of A , not necessarily the last few rows. Therefore, we can compute the dimensions of $B_d^{K_0}(K)$ and $Z_d^{K_0}(K)$ as

$$\begin{aligned} \dim(B_d^{K_0}(K)) &= \text{rank}(\partial_{d+1}) - \text{rank}(\partial_{d+1}^{K \setminus K_0}), \text{ and} \\ \dim(Z_d^{K_0}(K)) &= \text{rank}(\hat{Z}_d) - \text{rank}(\hat{Z}_d^{K \setminus K_0}), \end{aligned}$$

where $\partial_{d+1}^{K \setminus K_0}$ and $\hat{Z}_d^{K \setminus K_0}$ are the matrices formed by rows of ∂_{d+1} and \hat{Z}_d whose corresponding simplices do not belong to K_0 .

Algorithm. The procedure **Contain-Nonbounding-Cycle** tests whether K_0 carries any nonbounding cycle of K by testing whether $\dim(B_d^{K_0}(K))$ and $\dim(Z_d^{K_0}(K))$ are

different. Since columns in \hat{H}_d correspond to β_d nonbounding cycles representing a homology basis, the ranks of \hat{Z}_d and ∂_{d+1} differ by β_d . K_0 carries nonbounding cycles of K if and only if

$$\text{rank}(\hat{Z}_d^{K \setminus K_0}) - \text{rank}(\partial_{d+1}^{K \setminus K_0}) \neq \beta_d.$$

We use the algorithm of Wiedemann[86] for the rank computation.

In our algorithm, the boundary matrix ∂_{d+1} is given. The matrix \hat{H}_d can be precomputed by running persistent homology algorithm once, with an arbitrary filter function. Columns of \hat{H}_d are simply columns of matrix V corresponding to positive simplices paired with $+\infty$.

4.3.3 Complexity of the Improved Algorithm

The algorithm **Measure-All**(K) runs the improved procedure **Measure-Smallest** β_d times, with the augmented complex K' as the input complex. **Measure-Smallest**(K') applies the persistent homology algorithm on K' once to compute \hat{H}_d and $r(p_0)$. Next, for each vertex, it runs the rank computation on submatrices of ∂_{d+1} and $\hat{Z}_d = [\partial_{d+1}, \hat{H}_d]$. Denoting m as the time of two rank computations, the algorithm takes $O(\beta_d(n^3 + nm))$, as the size of K' is $O(n + \beta_d) = O(n)$.

To know m , we need the number of nonzero entries in matrices ∂_{d+1} and \hat{Z}_d , as we are using a sparse matrix rank computation algorithm. Recall that in the augmented complex K' , we added $O(\beta_d)$ new $(d+1)$ -dimensional cells, each of which has $O(n)$ d -faces. Therefore, ∂_{d+1} has $O(n + \beta_d) = O(n)$ columns and $O(n(d+2) + n\beta_d) = O(nd + n\beta_d)$ nonzero entries. Since \hat{H}_d has β_d columns and $O(n\beta_d)$ nonzero entries, the size and number of nonzero entries of \hat{Z}_d are asymptotically the same as ∂_{d+1} .

Running Wiedemann's rank computation on such matrices takes $m = O(n \log n(nd + n\beta_d + n \log n))$. If d and β_d are small enough – that is, $O(\log n)$ or less – then we have improved the **Measure-All**(K) to $O(\beta_d(n^3 + n^3 \log^2 n)) = O(\beta_d n^3 \log^2 n)$. If we are dealing with an unusual situation in which d or β_d is large – say $\Theta(n)$ – then the matrices are not sparse. We would prefer to use the old algorithm, with complexity $O(\beta_d n^4)$.

4.3.4 Measuring a Given Class

One interesting question is, instead of computing the optimal basis, can we measure a single given class, $[z]$. We modify the procedure **Measure-Smallest** to achieve this. Again, we iterate through all vertices. For each vertex p , we find the smallest geodesic ball centered at p carrying $[z]$, namely, $B_p^{r(p)}$. We apply persistent homology on the complex using f_p as the filter function. We pick all the columns in V corresponding to positive simplices that are paired with $+\infty$, namely, $z_1, z_2, \dots, z_{\beta_d}$, sorted according to their order in the filtration. We find the smallest index i so that z is a linear combination of boundaries and z_1, z_2, \dots, z_i , namely,

$$z = [\partial_{d+1}, z_1, z_2, \dots, z_i]\gamma \tag{4.1}$$

The positive simplex corresponding to this smallest i gives us $r(p)$. Replacing ∂_{d+1} with 0, we get a representative cycle of $[z]$ carried by $B_p^{r(p)}$, $[0, z_1, z_2, \dots, z_i]\gamma$. Iterating through every vertex p , we find the smallest ball carrying $[z]$, $B_{p_{\min}}^{r(p_{\min})}$, and consequently the size and localized cycle of $[z]$.

The algorithm applies persistent homology once, which takes $O(n^3)$. Next, the algorithm runs rank computation $O(n)$ times, on a matrix with $O(nd + n\beta_d)$ nonzero entries. Using Wiedemann's rank computation, the complexity is $O(n^3 \log^2 n)$ when d and β_d are both $O(\log n)$, and $O(n^4)$ otherwise.

4.4 Stability Result.

In this section, we prove that our measurement of homology is stable: small changes of the geometry of the space imply small changes of our measurement. We define a change of the geometry of the space as a change of the metric in the space. We measure this change by measuring the L_∞ -norm difference of geodesic distance functions before and after the change. To facilitate the proof, we assume that during the change, the simplicial complex remains the same except in terms of its edge weights, and thus, the discrete geodesic distances. Formally, we quantify the change of the geometry as

$$\epsilon = \max_{p \in \text{vert}(K)} |f_p^1 - f_p^2|_\infty, \quad (4.2)$$

where f_p^1 and f_p^2 are the discrete geodesic distance functions from p before and after the change.

In this section, we prove the stability of our measurement by showing that (1) for a single homology class, the size is stable; and (2) for the whole homology group, although the optimal homology basis is not stable, the group structure filtered by the size is stable. For convenience, we drop the dimension of the pertinent homology, d , in notations.

4.4.1 A Single Class

For a single homology class, the size measure remains stable. Denote $S^1(h)$ and $S^2(h)$ as the size of class h before and after the change (computed using f_p^1 and f_p^2 , respectively). We have the following theorem.

Theorem 4.4.1. $|S^1(h) - S^2(h)| \leq \epsilon$, where ϵ is the upper bound of the geometry change as defined in Equation (4.2).

Proof. Denote $r^1(p)$ and $r^2(p)$ as the radii of the smallest geodesic balls carrying h computed using the geodesic distance f_p^1 and f_p^2 , respectively. We show that for any specific vertex p , $|r^1(p) - r^2(p)| \leq \epsilon$. This leads to the fact that $S^1(h) = \min_{p \in \text{vert}(K)} r^1(p)$ and $S^2(h) = \min_{p \in \text{vert}(K)} r^2(p)$ differ in no more than ϵ .

For any simplex σ in the ball $B_p^{r^1(p)}$ calculated using f_p^1 , $f_p^1(\sigma) \leq r^1(p)$, and thus $f_p^2(\sigma) \leq f_p^1(\sigma) + \epsilon \leq r^1(p) + \epsilon$. This means that the ball $B_p^{r^1(p)}$ calculated

using f_p^1 is a subcomplex of the ball $B_p^{r^1(p)+\epsilon}$ calculated using f_p^2 , which thus carries h . Therefore, according to the definition of $r^2(p)$, it is no greater than $r^1(p) + \epsilon$. Similarly, $r^1(p) \leq r^2(p) + \epsilon$. \square

4.4.2 The Homology Group

Since the size of different classes can be very close, the optimal homology basis is not stable. For example, in Figure 1.2, either $\{[z_1], [z_2], [z_3]\}$ or $\{[z_1], [z_2], [z_3] + [z_1]\}$ can be the optimal homology basis for little geometry changing, because the sizes of $[z_3]$ and $[z_1] + [z_3]$ are quite close. However, there is still some stability property in the homology group structure if we filter it with the class size. More specifically, the subgroup generated by small homology classes remains stable. For example, in Figure 1.2, although the optimal homology basis is unstable, the subgroup generated by the two smaller classes in the optimal homology basis will always be the one generated by $[z_1]$ and $[z_2]$.

We formalize this stability by defining the subgroup filtration of a topological space and the distance between two such filtrations. A subgroup filtration is a sequence of subgroups of the homology group generated by subsets of the optimal homology basis filtered by the class size. A formal definition is as follows.

Definition 4.4.2 (Subgroup Filtration). *Given an optimal homology basis $\mathcal{H} = \{h_1, h_2, \dots, h_\beta\}$, where we assume $S(h_i) \leq S(h_{i+1})$, a subgroup filtration is a sequence of subgroups of the homology group, $\mathcal{X} = \{\psi_0, \psi_1, \psi_2, \dots, \psi_\beta\}$, where $\psi_i = \text{span}(h_1, h_2, \dots, h_i)$ is the subgroup generated by the classes h_1, h_2, \dots, h_i .*

Since here the homology group and all its subgroups are vector spaces, we use the notation $\psi_i = \text{span}(h_1, h_2, \dots, h_i)$ when we say h_1, h_2, \dots, h_i generates ψ_i .

Obviously, the subgroup filtration is a sequence of subgroups of $H(K)$ with a nested structure

$$\emptyset = \psi_0 \subset \psi_1 \subset \dots \subset \psi_\beta = H(K).$$

For convenience, we denote the size of a subgroup, ψ_i , as the size of the largest class in the optimal homology basis generating ψ_i , formally, $S(\psi_i) = S(h_i)$. Please note that $S(\psi_i)$ is not the size of the largest class in ψ_i .

Given two different sets of discrete geodesic distance functions (different geometries) of a same topological space, f_*^1 and f_*^2 , we have two different subgroup filtrations \mathcal{X}^1 and \mathcal{X}^2 . Next, we define their distance, which requires the definition of the projection of one subgroup in one filtration onto the other filtration.

Definition 4.4.3 (Projection). *Given two subgroup filtrations of a same homology group $\mathcal{X}^1 = \{\psi_0^1, \psi_1^1, \psi_2^1, \dots, \psi_\beta^1\}$ and $\mathcal{X}^2 = \{\psi_0^2, \psi_1^2, \psi_2^2, \dots, \psi_\beta^2\}$, define the projection of ψ_i^1 onto \mathcal{X}^2 as the first subgroup in \mathcal{X}^2 that carries ψ_i^1 , formally,*

$$\text{proj}(\psi_i^1, \mathcal{X}^2) = \psi_j^2, \quad \text{s.t. } j = \min_{\psi_i^1 \subseteq \psi_k^2} k.$$

Definition 4.4.4 (Distance). Define the distance between \mathcal{X}^1 and \mathcal{X}^2 as the maximal difference between the sizes of any subgroup in \mathcal{X}^1 or \mathcal{X}^2 and its projection onto the other filtration, formally,

$$\text{dist}(\mathcal{X}^1, \mathcal{X}^2) = \max\{\max_i |S^1(\psi_i^1) - S^2(\text{proj}(\psi_i^1, \mathcal{X}^2))|, \max_i |S^2(\psi_i^2) - S^1(\text{proj}(\psi_i^2, \mathcal{X}^1))|\}.$$

Let \mathcal{X}^1 and \mathcal{X}^2 be the subgroup filtrations of the original space and the one after the change. We can prove the following stability result.

Theorem 4.4.5.

$$\text{dist}(\mathcal{X}^1, \mathcal{X}^2) \leq \epsilon = \max_{p \in \text{vert}(K)} |f_p^1 - f_p^2|_\infty. \quad (4.3)$$

Proof. Take a subgroup ψ_i^1 , generated by $h_1^1, h_2^1, \dots, h_i^1$, the smallest i elements of the optimal homology basis \mathcal{H}^1 , determined by f_*^1 . For any $j \in [1, i]$, we have

$$S^2(h_j) \leq S^1(h_j) + \epsilon \leq S^1(\psi_i^1) + \epsilon,$$

in which the first and the second inequalities are due to Theorem 4.4.1 and the definition of ψ_i^1 , respectively. Therefore, we have

$$S^2(\text{proj}(\psi_i^1, \mathcal{X}^2)) \leq \max_{j \in [1, i]} S^2(h_j) \leq S^1(\psi_i^1) + \epsilon.$$

This is true for all $i \in [1, \beta]$. Similarly we can prove for any subgroup of \mathcal{X}^2 , its distance from its projection onto \mathcal{X}^1 is upper-bounded by ϵ . Equation (4.3) is proved. \square

4.4.3 Continuous Case

Our algorithm computes the size measure of homology classes using a simplicial complex and the discrete geodesic distance on it. For a general space \mathbb{X} , we want to make sure that our algorithm is sufficient for the measurement. In this section, we prove that as long as we have a simplicial complex K approximating \mathbb{X} well enough in geometry, our algorithm on K will produce a size measure which is close to that of \mathbb{X} . To simplify the proof, we assume that the underlying space of the simplicial complex is equal to the given topological space, $|K| = \mathbb{X}$.

To facilitate the proof, we formalize the size measure of the underlying space \mathbb{X} , $\tilde{S}(h)$, and the size measure our algorithm computed using K and its discrete geodesic distance, $S(h)$. Denote $\widetilde{\text{dist}} : \mathbb{X} \times \mathbb{X} \rightarrow \mathbb{R}$ as the true geodesic distance between points in \mathbb{X} . We define the size of a homology class h as

$$\tilde{S}(h) = \min_{p \in \mathbb{X}} r_{\tilde{f}_p}(h), \quad (4.4)$$

where $\tilde{f}_p : \mathbb{X} \rightarrow \mathbb{R}$ is the geodesic distance from p and $r_{\tilde{f}_p}(h)$ is the radius of the smallest geodesic ball centered at p carrying h , formally, $r_{\tilde{f}_p}(h) = \min r$, such that

$\tilde{f}_p^{-1}(-\infty, r]$ carries h .⁶

Next, we define $S(h)$. Denote $\text{dist} : \text{vert}(K) \times \text{vert}(K) \rightarrow \mathbb{R}$ as the discrete geodesic distance between vertices of K . Our algorithm computes the smallest ball centered at vertices of K , using the discrete geodesic distance

$$f_p(q) = \begin{cases} \text{dist}(p, q), & q \in \text{vert}(K) \\ \max_{v \in \text{vert}(\sigma)} \text{dist}(p, v), & q \in \text{int}(\sigma) \end{cases} \quad (4.5)$$

where $p \in \text{vert}(K)$ and $q \in \mathbb{X}$. Our algorithm finds the minimum from the radii of the smallest geodesic balls carrying h centered at all vertices, formally,

$$S(h) = \min_{p \in \text{vert}(K)} r_{f_p}(h). \quad (4.6)$$

We define τ and l to parameterize how well the simplicial complex K approximates \mathbb{X} in geometry. τ bounds the difference between the discrete geodesic distance between vertices and the real geodesic distance. Formally, for any two vertices $p, q \in \text{vert}(K)$,

$$|\text{dist}(p, q) - \widetilde{\text{dist}}(p, q)| \leq \tau.$$

l bounds the size of each simplex. For any simplex $\sigma \in K$ and $p, q \in |\sigma|$,

$$\widetilde{\text{dist}}(p, q) \leq l.$$

Next we present our theorem.

Theorem 4.4.6. $|S(h) - \widetilde{S}(h)| \leq \tau + 2l$.

Proof. Given any point $p \in |\sigma_1|$, take a vertex $p' \in \text{vert}(\sigma_1)$. We show that for any point q ,

$$\tilde{f}_p(q) \leq f_{p'}(q) + \tau + 2l. \quad (4.7)$$

Suppose $q \in |\sigma_2|$. Take $q' = q$ if $q \in \text{vert}(\sigma_2)$. Otherwise, take $q' \in \text{vert}(\sigma_2)$ as the one with the maximal $f_{p'}$. Due to triangle inequality and definitions of τ and l , we have

$$\begin{aligned} \tilde{f}_p(q) &\leq \widetilde{\text{dist}}(p, p') + \widetilde{\text{dist}}(p', q') + \widetilde{\text{dist}}(q', q) \\ &\leq \widetilde{\text{dist}}(p', q') + 2l \\ &\leq f_{p'}(q') + \tau + 2l \\ &= f_{p'}(q) + \tau + 2l. \end{aligned}$$

Equation (4.7) leads to the fact that any geodesic ball defined by \tilde{f}_p with radius r is carried by a discrete geodesic ball defined by $f_{p'}$ with radius $r + \tau + 2l$. Similarly, we could prove any discrete geodesic ball defined by f_p ($p \in \text{vert}(K)$) with radius

⁶ To guarantee the existence of the minimum, we assume the topological space \mathbb{X} is compact.

r is carried by a geodesic ball defined by \tilde{f}_p with radius $r + \tau + 2l$. The theorem is proved as the size of a homology class is defined by the smallest geodesic ball carrying it. \square

CHAPTER 5

Homology Localization

5.1 Problem Formalization

Given an objective function defined on all the d -cycles,

$$\text{cost} : \mathbb{Z}_d(K) \rightarrow \mathbb{R},$$

we formalize the localization problem as a combinatorial optimization problem.

Problem 5.1.1 (Localizing Homology.).

INPUT: a simplicial complex K with size n , a d -dimensional nontrivial homology class $h = [z_0]$

OUTPUT: a cycle $z \in h$

MINIMIZE: $\text{cost}(z)$

In this chapter, we explore three options of the objective function, $\text{cost}(z)$, i.e. the *volume*, *diameter* and *radius*.

1. **Volume:** the number of simplices in a cycle. The definition can be extended to incorporate weight function defined on simplices.
2. **Diameter:** the maximum of pairwise discrete geodesic distances between vertices of the cycle.
3. **Radius:** the radius of the smallest discrete geodesic ball carrying the cycle.

For convenience, we call the localization problems using these three objective functions LHV, LHD and LHR, respectively. Our contributions include:

1. For the volume function,
 - We prove LHV is NP-hard to approximate within any constant factor when the Betti number of the pertinent homology is $\Theta(n)$;
 - For homology of 2-dimension or higher, we prove a simpler problem, finding the nonbounding cycle with the minimal volume (MVNC), is NP-hard to approximate within any constant factor. This leads to the inapproximability of LHV with fixed Betti number and computing the homology cycle basis with minimal total volume;
 - We provide a polynomial algorithm for a special case, using min-cut algorithm.
2. Prove that LHD is NP-hard to compute.
3. Discussions concerning LHR.

5.2 Volume

The first choice of the objective function is volume.

Definition 5.2.1 (Volume). *The volume of a cycle is the number of its simplices, $\text{vol}(z) = \text{card}(z)$.*

For example, the volume of a 1-cycle, a 2-cycle and a 3-cycle are the numbers of their edges, triangles and tetrahedra, respectively. The cycle with the smallest volume, denoted as z_v , agrees intuitively with the notion of a “well-localized” cycle. The problem of localizing a homology class with its minimal volume cycle, z_v , has been proven to be NP-hard [22, 18]. In this section, we prove that this problem is even NP-hard to approximate.

More generally, we can extend the the volume definition to be the sum of the weights assigned to simplices of the cycle, given an arbitrary weight function, $w : K \rightarrow \mathbb{R}$, defined on all the simplices of K , formally,

$$\text{vol}'(z) = \sum_{\sigma \in z} w(\sigma).$$

Computing z_v using this general volume definition is at least as hard as using Definition 5.2.1, which is in fact a special case (when $w(\sigma) = 1, \forall \sigma \in K$). Therefore, we will only treat the unweighted volume function.

Results. There are some existing hardness results, when the homology classes in question are 1-dimensional.

1. When the Betti number, $\beta_1 = \Theta(n)$, LHV has been proven to be NP-hard by polynomial reductions from a special case of MAX-2SAT [22] and MIN-CUT with negative edge weights [18].
2. Chambers et al [18] provided a polynomial algorithm when β_1 is fixed. The algorithm computes the shortest representative cycle for each of the $2^{\beta_1} - 1$ nontrivial classes.
3. Erickson and Whittlesey [52] computed in polynomial time the homology cycle basis with the minimal total volume, even when $\beta_1 = \Theta(n)$.

All these existing results are about 1-dimensional homology. We will study whether LHV is difficult in general dimension, and more importantly, how difficult it is.

The existing results suggest that the problem might be easier if we assume fixed Betti number, or if we compute the homology cycle basis with the minimal total volume instead. Therefore, we would also like to find out how difficult these problems could be. We prove hardness result for a simpler problem, computing the nonbounding cycle with the minimal volume, denoted as MVNC, which in turn

shows that all the problems we are interested in are NP-hard to approximate when the homology is 2-dimensional or higher.

For clearance, we list all the new results as follows.

1. When the homology in question is 1-dimensional or higher and the Betti number is $\Theta(n)$, it is NP-hard to approximate LHV within any constant factor (Theorem 5.2.5).
2. Concerning the problem MVNC, we have the following results:
 - When the homology in question is 2-dimensional or higher, we prove that MVNC is NP-hard to approximate within any constant factor even with fixed Betti number (Theorem 5.2.11).
 - A polynomial algorithm for MVNC for a special case: when the pertinent space is embedded in \mathbb{R}^N and the pertinent homology is $(N - 1)$ -dimensional.

5.2.1 LHV is NP-hard to approximate within any constant factor

We prove by a strict reduction from the nearest codeword problem (NCP), which cannot be approximated within any constant factor [5]. Problems used in previous reductions [22, 18] have constant approximation ratios, and thus cannot be used for our proof.

Problem 5.2.2 (Nearest Codeword Problem).

INPUT: an $m \times k$ generator matrix A over \mathbb{Z}_2 and a vector $y_0 \in \mathbb{Z}_2^m \setminus \text{span}(A)$.

OUTPUT: a vector $x \in \mathbb{Z}_2^k$

MINIMIZE: the Hamming distance between Ax and y_0

This is equivalent to finding the vector $y = Ax + y_0$ with the minimal Hamming weight.

Lemma 5.2.3. LHV cannot be approximated within any constant factor.

Proof. We prove by a strict reduction from NCP, namely, $\text{NCP} \leq_S \text{LHV}$.

Given an instance of NCP, namely, a generator matrix A and a vector y_0 , we first construct a cell complex, T , whose 2-dimensional boundary matrix is A . T has m 1-cells and k 2-cells corresponding to the m rows and k columns of A . Each 1-cell is a 1-dimensional cycle. Each 2-cell is a pipe with multiple openings. See Figure 5.1 for an example. Please note that we are abusing notation when we call T a cell complex, as these cells may not be homeomorphic to closed balls.

As each 1-chain of T is a 1-cycle, it is not hard to see that NCP is identical to the problem of computing the minimal volume representative cycle of a given 1-dimensional class of T , denoted as LHV_T . However, LHV_T is different from LHV, whose input is a simplicial complex which is supposed to be a triangulation of a topological space. Next, we embed T in Euclidean space \mathbb{R}^4 , and then triangulate

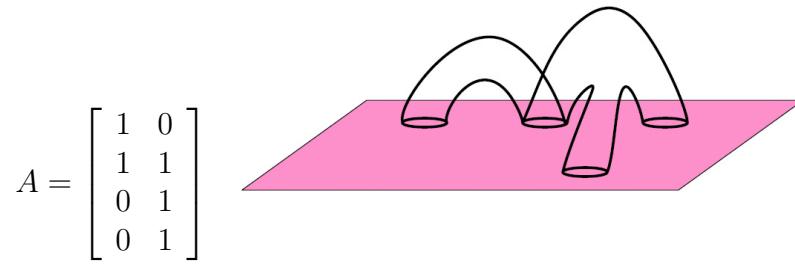


Figure 5.1: A 4×2 generator matrix A and the constructed cell complex, T .

$|T|$ into a simplicial complex K . With this construction, we will strictly reduce LHV_T to LHV .

The embedding is an analog of book embedding an arbitrary graph in \mathbb{R}^3 [87]. Take a 2-dimensional plane P in \mathbb{R}^4 . There are infinitely many 3-dimensional subspaces passing P . We embed all the 1-cells of T in the plane P . We embed each 2-cell (pipe) in one of those 3-dimensional subspaces. Since these 3-dimensional subspaces only intersect in P , these 2-cells do not intersect each other except for their boundaries.

We first triangulate each 1-cell of T into t_1 edges, with t_1 fixed and small. For convenience, we call the triangulation of all 1-cells K_1 , which is a subcomplex of K . There is a one-to-one correspondence between 1-cycles of T and 1-cycles of K_1 , denoted as ϕ . For any 1-cycle of T and its corresponding 1-cycle of K_1 , the ratio of their volumes is $1/t_1$.

Next, we triangulate interior of 2-cells of T (pipes) while keeping K_1 intact. The triangulation is fine enough so that for any 1-cycle of K , z , homologous to some 1-cycle of K_1 , we can compute in polynomial time a cycle z' of K_1 , which is homologous to z and has smaller or equal volume.

Our construction provides a polynomial transformation of every instance of LHV_T , (T, y_0) , into an instance of LHV , $(K, z_0 = \phi(y_0))$. For any such instance, and any feasible solution $z \in [z_0]$, we transform z into z' and then into a solution of LHV_T , $\phi^{-1}(z')$. This reduction is a strict reduction, as the ratio of the volumes of the optimal solution of the two problems is always $1/t_1$, and

$$\text{vol}(z) \geq \text{vol}(z') = t_1 \text{vol}(\phi^{-1}(z')).$$

□

Remark 5.2.4 (Extension to a manifold). *We extend the proof to the case when K is the triangulation of a manifold without boundary. The idea is as follows. Thicken the underlying space of T , then take its boundary as a new topological space. We get a 3-manifold (one less dimension than the ambient space), which can be triangulated similarly. This extension applies to all proofs in the rest of this chapter.*

Lemma 5.2.3 is about 1-dimensional homology. We extend the result to homology of any higher dimension.

Theorem 5.2.5. *For any $N \geq 1$, LHV for N -dimensional homology cannot be approximated within any constant factor.*

Proof. We show that when $N \geq 2$, LHV for $(N - 1)$ -dimensional homology can be strictly reduced to LHV for N -dimensional homology, namely, $\text{LHV}_{N-1} \leq_S \text{LHV}_N$. Together with Lemma 5.2.3, the theorem is proved.

Next, we explain briefly the reduction. Take any given simplicial complex of LHV_{N-1} and build a *suspension* of it, namely, two cones of the complex glued together at their base [88]. There is a one-to-one correspondence between the $(N - 1)$ -dimensional cycle group of the original complex and the N -dimensional cycle group of the new complex. This correspondence also works for the boundary groups. Since the volume of each $(N - 1)$ -cycle is $1/2$ of the volume of its corresponding N -cycle, this is a strict reduction. \square

5.2.2 Computing the minimal volume nonbounding cycle

In the previous section, the simplicial complex we constructed for LHV has $\Theta(n)$ Betti number. It has been revealed for 1-dimensional homology that LHV is polynomial for fixed Betti number, and it is polynomial to compute a homology cycle basis with the minimal total volume. In this section, we show that these two problems are both NP-hard to approximate when the pertinent homology is of 2-dimension or higher.

We prove the inapproximability of a simpler problem, denoted as MVNC.

Problem 5.2.6 (Minimal Volume Nonbounding Cycle Problem.).

INPUT: a simplicial complex K with size n

OUTPUT: a nonbounding d -cycle z

MINIMIZE: $\text{vol}(z)$

We prove that this problem is NP-hard to approximate within any constant factor, even when the pertinent Betti number is 1. This trivially leads to the inapproximability of the problem of LHV with fixed Betti number and the problem of computing the optimal homology cycle basis, as MVNC for $\beta_d = 1$ is a special case of these two problems.

We prove the inapproximability of MVNC by a strict reduction from the minimum distance problem (MDP) for binary linear code, which cannot be approximated within any constant factor [42].

Problem 5.2.7 (Minimum Distance Problem).

INPUT: an $m \times k$ generator matrix A over \mathbb{Z}_2 .

OUTPUT: a vector $x \in \mathbb{Z}_2^k$.

MINIMIZE: the Hamming weight of Ax .

We compute the corresponding parity-check matrix A^\perp . The MDP problem is then equal to finding the nonzero vector in the nullspace of A^\perp with the minimal Hamming weight.

Lemma 5.2.8. *MVNC for 2-dimensional homology cannot be approximated within any constant factor.*

Proof. We prove by a strict reduction from MDP, namely, $\text{MDP} \leq_S \text{MVNC}$.

Similar to Lemma 5.2.3, we construct a cell complex, T , with A^\perp as the 2-dimensional boundary matrix. Again, we embed T in \mathbb{R}^4 and triangulate it into a simplicial complex. In the triangulation, instead of triangulating 2-cells as fine as possible, we triangulate all of them into a same number of triangles, say t_2 . This guarantees a one-to-one correspondence between the group of 2-cycles of K and the nullspace of A^\perp . The volume of each 2-cycle is t_2 times of the Hamming weight of its corresponding vector. Since any nonzero 2-cycle is nonbounding, we have a strict reduction. \square

In Lemma 5.2.8, the constructed simplicial complex has $\Theta(n)$ Betti number. Next, we prove that the problem is NP-hard to approximate within any constant factor, even when β_2 is 1. We prove this by a reduction from the NCP problem. To facilitate the proof, we rewrite the problem in the following form.

Problem 5.2.9 (Nearest Codeword Problem (revised version)).

INPUT: an $m \times k$ generator matrix A over \mathbb{Z}_2 and a vector $y_0 \in \mathbb{Z}_2^m$

OUTPUT: a vector $z \in y_0 + \text{span}(A)$

MINIMIZE: the Hamming weight of z

Lemma 5.2.10. *For 2-dimensional homology, MVNC is NP-hard to approximate within any constant factor, even when $\beta_2 = 1$.*

Proof. We prove by a strict reduction from NCP.

Given a generator matrix $C = [A, y_0]$, we compute its parity-check matrix C^\perp . Following the method in Lemma 5.2.8, we construct a cell complex T using C^\perp as the 2-dimensional boundary matrix. There is a one-to-one correspondence between the 2-dimensional cycle group of T and $\text{nullspace}(C^\perp) = \text{span}(C)$. Next, for each column vector of A , we find the corresponding 2-cycle in T and seal it with a 3-cell. In the augmented complex, T' , the one and only nontrivial 2-dimensional homology class is identical to the coset $y_0 + \text{span}(A)$. Finding the smallest volume nonbounding 2-cycle of T' , denoted as $\text{MVNC}_{T'}$, is equal to finding the minimal weight vector in this coset and thus equal to solving the nearest codeword problem. We next strictly reduce $\text{MVNC}_{T'}$ to MVNC.

We first embed T in \mathbb{R}^4 . Although the 3-cells in $T' \setminus T$ might intersect each other in \mathbb{R}^4 . By extending the ambient space to \mathbb{R}^6 and using an analog of book embedding, we could safely embed them.

We triangulate T' into a simplicial complex K' as follows (in a similar scheme as in Lemma 5.2.3). We first triangulate the 2-skeleton of T' , T , into a simplicial complex K , in which all 2-cells are triangulated into the same number of triangles (t_2). There is a one-to-one correspondence between $\mathbb{Z}_2(K)$ and $\mathbb{Z}_2(T) = \mathbb{Z}_2(T')$.

Next, while keeping K intact, we triangulate interior of 3-cells as fine as possible so that for any nonbounding 2-cycle of K' , we could always find in polynomial

time a nonbounding 2-cycle of K which is homologous to it. As the volume of any 2-cycle of K is t_2 times of that of its corresponding 2-cycle of T' , we have a strict reduction from MVNC $_{T'}$ to MVNC. \square

Similar to Theorem 5.2.5, we extend the result to any higher dimension. We conclude this section with the following theorem.

Theorem 5.2.11. *MVNC is NP-hard to approximate within any constant factor for homology of 2-dimension or higher, even when the pertinent Betti number is 1.*

5.2.3 A special case

There is, however, a special case in which MVNC can be computed in polynomial time: when K is a N -dimensional complex embedded in \mathbb{R}^N and the pertinent nonbounding cycle is $(N - 1)$ -dimensional. In this section, we provide a polynomial algorithm, inspired by [68].

We add new N -cells to K to get a new complex K' , whose underlying space is \mathbb{R}^N . Each new cell covers one component of $\mathbb{R}^N \setminus |K|$. There are $\beta_{N-1} + 1$ new cells, one of which covers the infinity component. The boundary of each new cell is one component of the $(N - 1)$ -dimensional boundary of K . Here we are abusing notation again as the new cells may not be homeomorphic to closed balls.

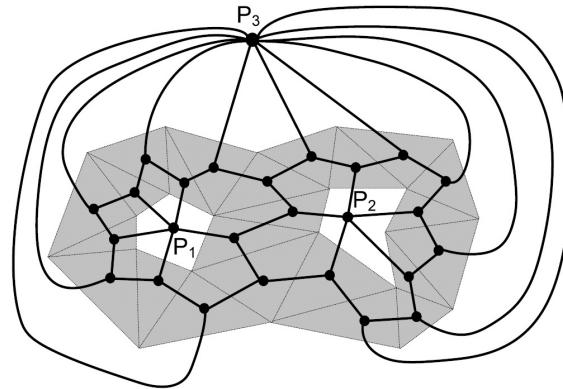


Figure 5.2: A 2-dimensional simplicial complex embedded in \mathbb{R}^2 . The dual graph G and G' are drawn in solid lines and vertices. Their difference, $G' \setminus G$, includes vertices p_1 , p_2 , p_3 and their incident edges.

We use the MIN-CUT algorithm on the dual graphs to solve the problem. The dual graph of K , G , is a subgraph of the dual of K' , G' . The set of new vertices $V' \setminus V$ is dual to the set of new N -cells. See Figure 5.2 for an example when $N = 2$.

We call a cycle *minimal* if none of its non-empty subsets is a cycle. We denote $C(G', G)$ as the set of *minimal edge cuts* (cuts whose subsets are not cuts) of G' which cut G' into two partitions each of which contains at least one vertex of $V' \setminus V$. There is a one-to-one correspondence between the set of minimal nonbounding $(N - 1)$ -cycles of K and the set of cuts $C(G', G)$. The volume of each cycle is equal to

the cardinality of its corresponding cut. As the nonbounding $(N - 1)$ -cycle with the smallest volume has to be one of the minimal cycles, it can be computed by computing the cut in $C(G', G)$ with the smallest cardinality.

To compute the minimal cardinality cut in $C(G', G)$, we enumerate all pairs of vertices, $(v_1, v_2) \in (V' \setminus V) \times (V' \setminus V)$. Compute the minimal (v_1-v_2) -cut for each pair. The one with the smallest cardinality is the desired one.

Remark 5.2.12. *The idea can be carried to the weighted volume function, but only if the weight function is non-negative.*

5.3 Diameter

When LHV is proved to be NP-hard to approximate, we resort to discrete geodesic distance related objective functions, diameter and radius.

Definition 5.3.1 (Diameter). *The diameter of a cycle is the diameter of its vertex set, $\text{diam}(z) = \text{diam}(\text{vert}(z))$, in which the diameter of a set of vertices is the maximal discrete geodesic distance between them, formally,*

$$\text{diam}(S) = \max_{p,q \in S} \text{dist}(p, q).$$

Intuitively, a representative cycle of h with the minimal diameter, denoted z_d , is the cycle whose vertices are as close to each other as possible. The intuition will be further illustrated in Section 5.4 by comparison against other criteria. We prove in Theorem 5.3.4 that computing z_d of h is NP-hard, by reduction from a special case of the NP-hard *Multiple-Choice Cover Problem* (MCCP) of Arkin and Hassin [4]. But the proof has not been published.

Remark 5.3.2. *We do not address the approximability of LHD, as we realize that z_d suffers from a “wiggling problem” and consequently may be geometrically complex (see Section 5.4). However, it is not hard to see that the reduction in Theorem 5.3.4 is strict, which implies that LHD cannot be approximated any better than this special case of MCCP. MCCP, and thus LHD, cannot be approximated within $2 - \epsilon$ for any $\epsilon > 0$, though we do not establish this formally.*

Problem 5.3.3 (Multiple-Choice Cover Problem).

INPUT: a set of vertices, $V = \{v_1, v_2, \dots, v_n\}$; a distance function $\text{dist} : V \times V \rightarrow \mathbb{R}^+$ satisfying triangular inequality; Disjoint subsets of V , S_1, S_2, \dots, S_m , such that $\bigcup_{i=1}^m S_i = V$

OUTPUT: a cover $C \subseteq V$ containing one and exactly one vertex from each subset S_i

MINIMIZE: $\text{diam}(C)$

Please note that the original MCCP problem of Arkin and Hassin only requires the cover to have nonempty intersection with each subset S_i . We revise the problem to facilitate our proof, without influencing the NP-hardness. The reason is the

optimal result of the revised problem is clearly an optimal result of the original problem.

Theorem 5.3.4. LHD is NP-hard to compute.

Proof. We present a polynomial-time algorithm transforming an input of MCCP into an input of LHD. Later we will show that the solution of LHD gives us the solution of MCCP. As part of the input of LHD, the constructed simplicial complex K consisting of m tubes, T_1, \dots, T_m , as well as extra edges connecting vertices.

We first embed the vertex set V in any metric space preserving the pairwise distance dist . Without loss of generality, we assume V is embedded in the Euclidean plane \mathbb{R}^2 , for ease of explanation. For each vertex subset S_i , we find a simple path in \mathbb{R}^2 , going through each vertex of S_i once without self-intersection, $\xi_i = (v_1, v_2, \dots, v_{\text{card}(S_i)})$, which contains $\text{card}(S_i) - 1$ edges. The edge lengths are the same as the distances between corresponding vertices. See Figure 5.3(a). We construct a

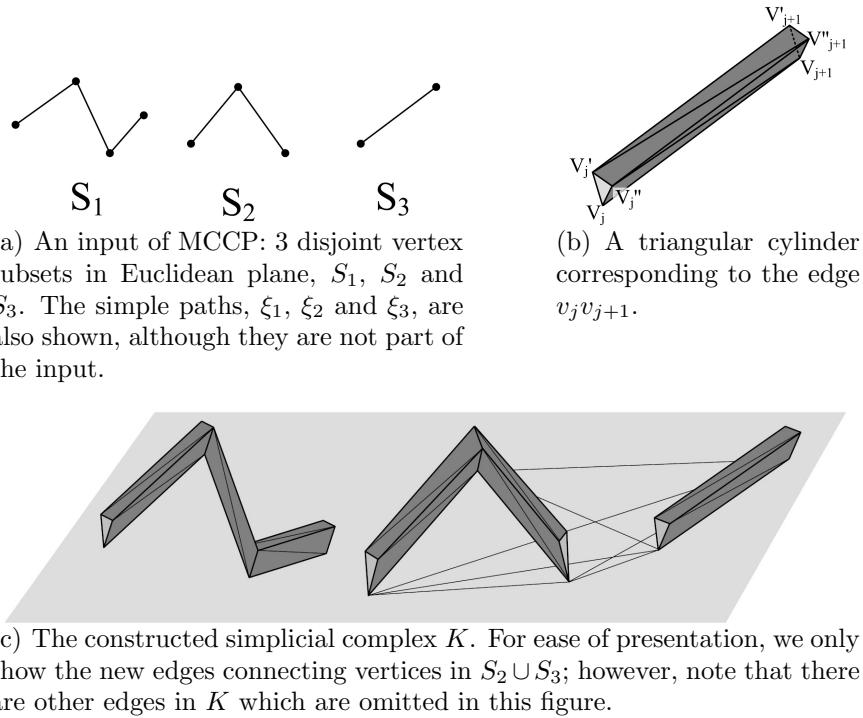


Figure 5.3: Explanation of Theorem 5.3.4 proof.

a slender threadlike tube T_i , which carries the path ξ_i . T_i has $(3 \text{ card}(S_i))$ vertices, $S_i \cup S'_i \cup S''_i$, where

$$S'_i = \{v'_1, v'_2, \dots, v'_{\text{card}(S_i)}\}, \quad \text{and} \quad S''_i = \{v''_1, v''_2, \dots, v''_{\text{card}(S_i)}\}.$$

For any j , v'_j and v''_j live very close to v_j . Corresponding to the $\text{card}(S_i) - 1$ edges in ξ_i , T_i consists of $\text{card}(S_i) - 1$ triangular cylinder concatenated together. By a *triangular cylinder* we mean the surface of a 3-prism with the two end triangles

missing. To facilitate the concatenation, corresponding edges of the end triangles may not be parallel to each other, as in a standard 3-prism. Each edge $v_j v_{j+1}$ corresponds to a triangular cylinder with vertices $v_j, v'_j, v''_j, v_{j+1}, v'_{j+1}$ and v''_{j+1} . In the triangular cylinder, the short edges are very short, say, no longer than ϵ . The long edges have the length similar to the length of edge $v_j v_{j+1}$. See Figure 5.3(b) for one such triangular cylinder.

We construct the simplicial complex, K , as follows. For any $i, T_i \subseteq K$, and for any two vertices $v_1, v_2 \in V$, if they are not neighbors, add an edge connecting them, whose length is their Euclidean distance in the Euclidean plane \mathbb{R}^2 . See Figure 5.3(c) for the complex constructed from the input in Figure 5.3(a). Please note that although in the figure, the embedding of K in \mathbb{R}^3 has self-intersection, the simplicial complex K can be embedded in Euclidean space of higher dimension, as we did in previous proofs.

For the constructed complex K , we use LHD to localize the 1-dimensional class $\sum_{i=1}^m h_i$, where h_i is the only 1-dimensional class carried by the tube T_i . We need a cycle to represent it as the input for LHD. We use $z_0 = \sum_{i=1}^m z_{i_0}$, where z_{i_0} is the 1-cycle whose vertices are v_{i_0}, v'_{i_0} and v''_{i_0} , in which v_{i_0} is an arbitrary vertex in S_i .

Next, we construct a cover C from the solution of LHD, z , and show that C is the solution of MCCP. We construct an intermediate vertex set $C_0 \subseteq V$ as follows. A vertex v belongs to C_0 if and only if any of v_i, v'_i and v''_i belongs to the vertex set of z , $\text{vert}(z)$. The solution z is in the form $\sum_{i=1}^m z_i$, where z_i represents class h_i . Therefore, C_0 has nonempty intersection with each vertex set S_i . We compute the cover C by picking one vertex from each $S_i \cap C_0$.

Within the simplicial complex,

$$\text{diam}(C) = \text{diam}(C_0) \quad \text{and} \quad |\text{diam}(C_0) - \text{diam}(z)| \leq 2\epsilon.$$

Furthermore, C has the same diameter in the simplicial complex, K , and in the Euclidean plane, \mathbb{R}^2 . Since ϵ is arbitrarily small, we can see that C is the cover with the minimal diameter in the Euclidean plane, and thus, is the solution of MCCP. \square

5.4 Radius

The third option for the objective function is radius.

Definition 5.4.1 (Radius). *The radius of a cycle is the radius of the smallest geodesic ball carrying it, formally,*

$$\text{rad}(z) = \min_{p \in \text{vert}(K), z \subseteq B_p^r} r$$

Given a homology class, the representative cycle with the minimal radius, denoted as z_r , is the same as the localized cycle defined in Chapter 4. Intuitively, z_r is the cycle whose vertices are as close to a vertex of K as possible. Our polynomial algorithms show that z_r can be computed in polynomial time.

However, in spite of its ease of computation, z_r may not necessarily be concise in an intuitive sense. It wiggles freely inside the smallest geodesic ball carrying it. See Figure 5.4(a) for example, in which we localize the only nontrivial homology class of an annulus (the light gray area). The dark gray area is the smallest geodesic ball carrying the class, whose center is p .

By contrast, the cycle with the minimal diameter, z_d , avoids this wiggling problem in this case and is concise in an intuitive sense (Figure 5.4(b)). This figure also illustrates that the radius and the diameter of a cycle are not strictly related. For the cycle z_r in Figure 5.4(a), its diameter is twice of its radius. For the cycle z_d in Figure 5.4(b), its diameter is equal to its radius.

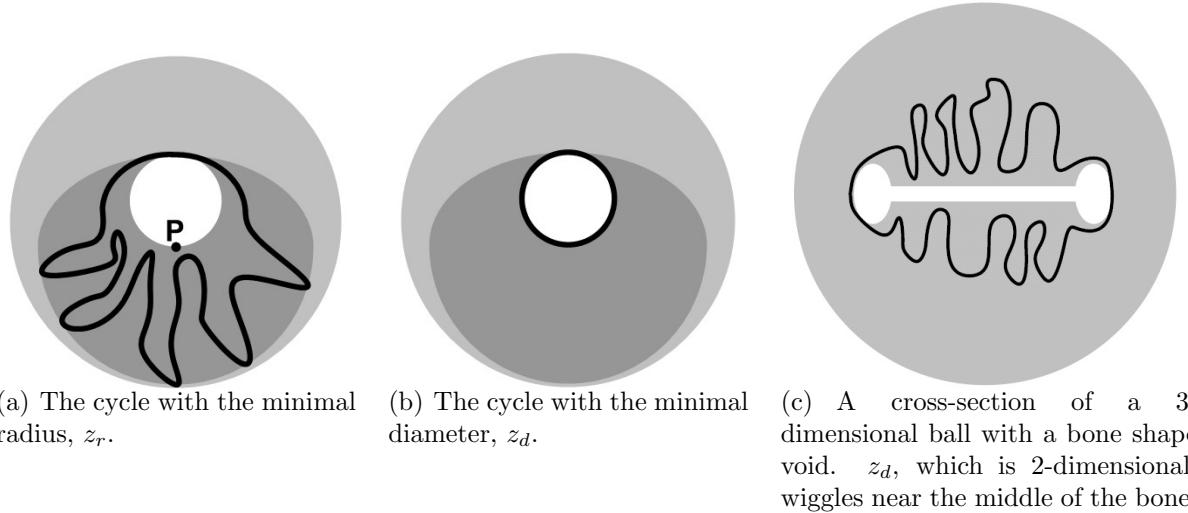


Figure 5.4: Cycles with the minimal radius and diameter.

We prove that z_r is a 2-approximation of z_d .

Theorem 5.4.2. $\text{diam}(z_r) \leq 2 \text{ diam}(z_d)$.

Proof. First, the triangle inequality of the geodesic distance suggests that for any two vertices of z_r , p_1 and p_2 , their geodesic distance is

$$\text{dist}(p_1, p_2) \leq \text{dist}(p_1, p_0) + \text{dist}(p_0, p_2) \leq 2 \text{ rad}(z_r),$$

where p_0 is the center of the smallest geodesic ball carrying the cycle z_r and the class. This implies that the diameter of z_r is no greater than twice of its radius.

Second, the diameter of z_d is no less than its radius. To see this, pick a geodesic ball centered at any vertex of z_d with radius $\text{diam}(z_d)$. This ball carries z_d . Finally,

$$\text{diam}(z_r) \leq 2 \text{ rad}(z_r) \leq 2 \text{ rad}(z_d) \leq 2 \text{ diam}(z_d).$$

□

As shown in Figure 5.4(a) and 5.4(b), this bound is a tight bound.

However, in general, the minimal diameter cycle also suffers from the wiggling problem. In Figure 5.4(c), we show an example in which the topological space is a closed 3-dimensional ball with a bone shape void in the middle. The minimal diameter 2-cycle, z_d , representing the only nontrivial 2-dimensional class, can freely wiggle near the middle of the bone, as the diameter is determined by the distance of the two ends of the bone. The reason for this phenomenon is in finding the minimal diameter cycle, we minimize the maximum of all pairwise geodesic distances. It is not hard to see that z_d does not wiggle only if for any $v \in \text{vert}(z_d)$, its longest distance from other vertices in z_d is close to $\text{diam}(z_d)$.

Remark 5.4.3 (Open Question). *An open question is whether we can use other discrete geodesic distance related measures, besides diameter and radius, which do not suffer from the wiggling problem. For example, can we use the normalized sum of the pairwise geodesic distances? Furthermore, what if we restrict the geodesic distance to be within the cycle (rather than the entire complex)? It is conceivable that these distance related measures might be easier to compute, as localization with volume measure has been shown to be extremely hard.*

CHAPTER 6

Open Problem: Enriched Persistent Homology

6.1 Motivation

In persistent homology, when we say that a class h is born at time i , it is the whole coset $h + \text{img}(F^{i-1,i})$ that is born. When we say h dies at time j , one and only one class becomes trivial. The remaining classes become identical to classes that are born before time i . The standard persistent homology suggests a canonical choice, namely, the class that becomes trivial at time j , to represent the new born classes.

Persistent homology record the birth and death times of persistent classes. But this is not all we can see by human intuition. In Figure 6.1, persistent homology would suggest the classes $[z_3]$ and $[z_2 + z_3]$ as the new born classes for the two cases, respectively. However, human intuition would realize that the new born class is related to the existing class $[z_2]$ in the right case but not in the left case. The reason is in the right, the cycle representing the new born class carries a representative cycle of the existing class $[z_2]$.

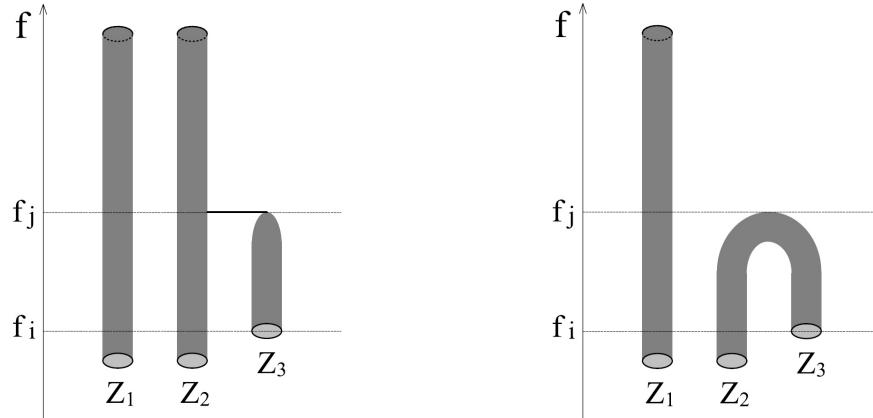


Figure 6.1: Motivating examples. The filter function f is the height function. Classes $[z_1]$ and $[z_2]$ are born earlier. Both cases have the same persistent diagram.

This leads to a question of whether we could reveal more structural relationship between persistent classes, in order to differentiate the two cases in Figure 6.1, when standard persistence cannot. In this chapter, we would address this problem. We intend to formalize an enriched persistent homology satisfying

1. The information it carries should be a superset of standard persistence;
2. The information should be stable with regard to geometric perturbation.

The second property is similar to the stability of standard persistence [25]. It guarantees that we could differentiate filtrations by comparing their enriched persistences.

We first present the *total persistent homology*, an extension of standard persistence invented by David Cohen-Steiner. We formalize total persistence and propose various enriched persistent homology derived from it. Unfortunately, we would show that all the proposed definitions fail to satisfy the stability restriction. For completeness, we conclude this chapter by an algorithm to update the total persistence when a simplex transposition happens in the simplex-ordering.

6.2 Total Persistent Homology

Total persistence is defined as follows. In standard persistence, we stop reducing a column when the lowest nonzero entry is unpaired. In total persistence, instead of stop, we keep reducing until there is no unpaired positive entries left except for the lowest one, which will be paired with the current column.

Similarly to standard persistence, we could apply these column reductions to D by multiplying it to a *upper unitriangular matrix*, T , namely, a upper triangular matrix whose diagonal entries are all 1's. Column vectors of T form a chain basis $\{\bar{\sigma}_i\}$, formally,

$$\begin{aligned}\bar{\sigma}_i &= (\sigma_1, \sigma_2, \dots, \sigma_n)T_{*,i}, \text{ and} \\ \sigma_i &= (\bar{\sigma}_1, \bar{\sigma}_2, \dots, \bar{\sigma}_n)T_{*,i}^{-1},\end{aligned}$$

where $T_{*,i}$ and $T_{*,i}^{-1}$ are the i -th column vectors of T and T^{-1} , respectively. Note that T^{-1} is also upper unitriangular.

We further transform the totally reduced matrix so that each of its row corresponds to $\bar{\sigma}_i$ rather than σ_i , formally,

$$R = T^{-1}DT. \quad (6.1)$$

The sets of chains form bases for different groups (Table 6.1). Recall $K_n = f^{-1}(-\infty, f_n]$ is the sublevel set with $t = f(\sigma_n)$. We denote K'_n as K_n excluding all the positive p -simplices and all simplices whose dimension is bigger than p , formally,

$$K'_n = K_n \setminus (\{\sigma_i \mid i \leq n, \dim(\sigma_i) = p, \sigma_i+\} \cup \{\sigma_i \mid \dim(\sigma_i) > p\}),$$

in which we say σ_i+ (resp. σ_i-) when σ_i is positive (resp. negative).

Table 6.1: Bases formed by different sets of chains in total reduction.

| | |
|--|--------------------------------------|
| $\{\bar{\sigma}_i \mid i \leq n, \dim(\sigma_i) = p\}$ | a basis of $C_p(K_n)$ |
| $\{\partial\bar{\sigma}_i \mid i \leq n, \dim(\sigma_i) = p+1, \sigma_i-\}$ | a basis of $B_p(K_n)$ |
| $\{\bar{\sigma}_i \mid i \leq n, \dim(\sigma_i) = p, \sigma_i+\}$ | a basis of $Z_p(K_n)$ |
| $\{\bar{\sigma}_i \mid i \leq n, \dim(\sigma_i) = p, \sigma_i + \text{unpaired}\}$ | a homology cycle basis of $H_p(K_n)$ |
| $\{\bar{\sigma}_i \mid i \leq n, \dim(\sigma_i) = p, \sigma_i-\}$ | a basis of $C_p(K'_n)$ |

Next, we prove that the total reduction is unique, in spite of the reduction order, as well as a formal definition of chains $\bar{\sigma}_i$.

Lemma 6.2.1 (Total Reduction Uniqueness). *Given D , the matrix T and R are unique as long as*

- T is upper unitriangular;
- $\forall \sigma_i = \text{low}(\sigma_j), R_{i,k} = 0 \quad \forall k > j$.

Proof. We prove by induction. Suppose the uniqueness is correct for K_n . We prove for K_{n+1} basing on Table 6.1.

For a simplex σ_{n+1} whose dimension is $p+1$. Its boundary is a p -cycle carried by K_n . We could write it uniquely as a linear combination of the homology cycle basis and the boundary basis, formally,

$$\partial\sigma_{n+1} = \sum_{i=1}^n \lambda_i^{n+1} \bar{\sigma}_i + \sum_{j=1}^n \mu_j^{n+1} \partial\bar{\sigma}_j,$$

in which for any $\bar{\sigma}_i$ in the first item, its corresponding σ_i is p -dimensional, positive and unpaired. For any $\bar{\sigma}_j$ in the second item, its corresponding σ_j is $(p+1)$ -dimensional and negative. We call the two items the nonbounding part and the boundary part of $\partial\sigma_{n+1}$. The nonbounding part is 0 if σ_{n+1} is positive.

We define the chain $\bar{\sigma}_{n+1} \in C_{p+1}(K_{n+1})$ as the one whose boundary is the nonbounding part,

$$\partial\bar{\sigma}_{n+1} = \sum_{i=1}^n \lambda_i^{n+1} \bar{\sigma}_i.$$

There is one and only one solution for this equation, namely,

$$\bar{\sigma}_{n+1} = \sigma_{n+1} + \sum_{j=1}^n \mu_j^{n+1} \bar{\sigma}_j,$$

as the set $\{\partial\bar{\sigma}_j \mid i \leq n, \dim(\sigma_j) = p+1, \sigma_j-\}$ is a basis of the boundary group. It is straightforward to verify that $(\mu_1^{n+1}, \mu_2^{n+1}, \dots, \mu_n^{n+1}, 1, 0, \dots, 0)^T$ and $(\lambda_1^{n+1}, \lambda_2^{n+1}, \dots, \lambda_n^{n+1}, 0, \dots, 0)^T$ correspond to the $(n+1)$ -th column vectors of T^{-1} and R , respectively. The uniqueness of all the coefficients λ_i^{n+1} 's and μ_j^{n+1} 's leads to the uniqueness of R and T .

Intuitively, $\mu_j^{n+1} = 1$ ($j \leq n$) iff the j -th fully reduced column is used to reduce the $(n+1)$ -th column. $\sum_{i=1}^n \lambda_i^{n+1} \bar{\sigma}_i$ is the nonbounding cycle sealed by σ_{n+1} . It is the linear combination of the homology cycle basis of $H_p(K_n)$, representing a persistent homology class destroyed by σ_{n+1} . Among all the σ_i 's whose λ_i^{n+1} are 1, the one with the largest i creates the class and is paired with σ_{n+1} . \square

6.3 Enriched Persistences and Instability Cases

Total persistence and its uniqueness provide a nice foundation for defining enriched persistence. The basic idea is constructing a graph whose vertices are persistent classes in the persistence diagram. We explore several different options for the edges. Let $j_1 < j_2$.

Definition 6.3.1 (Cycle Edge). *Two persistent classes, $(\sigma_{i_1}, \sigma_{j_1})$ and $(\sigma_{i_2}, \sigma_{j_2})$ have a cycle edge connected iff there exists $k < i_1, i_2$ such that $\lambda_k^{j_1} = \lambda_k^{j_2} = 1$. Intuitively, the representative cycles of the two classes share a common cycle in the homology cycle basis of $H_p(K_{j_2-1})$, namely, $\{\bar{\sigma}_i \mid \sigma_i + , \text{ unpaired in } K_{j_2-1}\}$.*

The relationship we discussed in Figure 6.1 is this cycle edge.

Definition 6.3.2 (Reduction Edge). *Two persistent classes, $(\sigma_{i_1}, \sigma_{j_1})$ and $(\sigma_{i_2}, \sigma_{j_2})$ have a reduction edge connected iff $\mu_{j_1}^{j_2} = 1$. Intuitively, we used the fully reduced j_1 -th column to reduce the j_2 -th column during the reduction.*

Definition 6.3.3 (Inclusion Edge). *Two persistent classes, $(\sigma_{i_1}, \sigma_{j_1})$ and $(\sigma_{i_2}, \sigma_{j_2})$ have an inclusion edge connected iff the entry (j_1, j_2) of T , $T_{j_1, j_2} = 1$. Intuitively, the chain $\bar{\sigma}_{j_2}$ contains the simplex σ_{j_1} .*

All these options are meaningful in terms of both algebra and geometric intuition. However, we found counterexamples showing that these edges are not stable when we perturb the geometry by transposing two consecutive simplices in the simplex-ordering.

In Figure 6.2(a), we show an example in which cycle edges are not stable. The complex is filtered by the height function. τ_i and σ_i ($i = 0, 1$) are constructors and destructors of 1-dimensional persistent classes, respectively. For ease of exposition, we say τ_i 's correspond to the two 1-cycles (thickened lines) and σ_i 's correspond to the two tube shaped patches. In a real triangulation, each τ_i is the last edge of the corresponding 1-cycle, and each σ_i is the last triangle of the corresponding tube patch. We transpose the simplices σ_0 and σ_1 in the simplex-ordering

$$(\tau_0, \tau_1, \sigma_0, \sigma_1).$$

If we only take the rows corresponding to $\bar{\tau}_i$'s and columns corresponding to $\bar{\sigma}_i$'s, the totally reduced matrix R before and after the perturbation would be

$$R = \begin{array}{c|cc} & \bar{\sigma}_0 & \bar{\sigma}_1 \\ \hline \bar{\tau}_0 & 1 & 1 \\ \bar{\tau}_1 & 1 & 0 \end{array} \text{ and } R' = \begin{array}{c|cc} & \bar{\sigma}_1 & \bar{\sigma}_0 \\ \hline \bar{\tau}_0 & 1 & 0 \\ \bar{\tau}_1 & 0 & 1 \end{array}$$

One cycle edge (connecting (τ_1, σ_0) and (τ_0, σ_1)) disappears due to the perturbation. Figure 6.2(b) is an example in which the reduction edges and inclusion edges are not stable. τ_i 's and σ_i 's are constructors and destructors of 1-dimensional persistent classes, respectively. τ_i 's correspond to the three 1-cycles (thickened lines) and σ_i 's

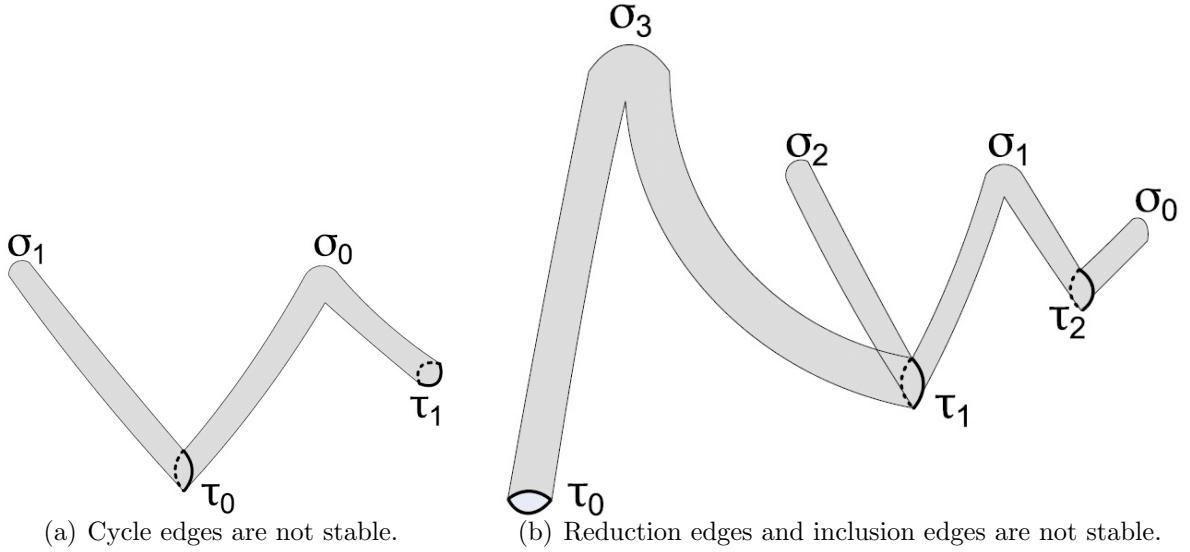


Figure 6.2: Examples in which the proposed enriched persistent homology is not stable with regard to geometric perturbations.

correspond to the four tube shaped patches. τ_1 is shared by three 2-cells, σ_1, σ_2 and σ_3 . We transpose the simplices σ_1 and σ_2 in the simplex-ordering

$$(\tau_0, \tau_1, \tau_2, \sigma_0, \sigma_1, \sigma_2, \sigma_3).$$

The matrices T^{-1} and T before and after the perturbation (denoted as $(T^{-1})'$ and T') would be

$$\begin{array}{c} T^{-1} = \begin{array}{c|cccc} & \sigma_0 & \sigma_1 & \sigma_2 & \sigma_3 \\ \hline \bar{\sigma}_0 & 1 & 1 & 0 & 0 \\ \bar{\sigma}_1 & 0 & 1 & 1 & 1 \\ \bar{\sigma}_2 & 0 & 0 & 1 & 0 \\ \bar{\sigma}_3 & 0 & 0 & 0 & 1 \end{array}, \quad T = \begin{array}{c|cccc} & \bar{\sigma}_0 & \bar{\sigma}_1 & \bar{\sigma}_2 & \bar{\sigma}_3 \\ \hline \sigma_0 & 1 & 1 & 1 & 1 \\ \sigma_1 & 0 & 1 & 1 & 1 \\ \sigma_2 & 0 & 0 & 1 & 0 \\ \sigma_3 & 0 & 0 & 0 & 1 \end{array}, \\ (T^{-1})' = \begin{array}{c|cccc} & \sigma_0 & \sigma_2 & \sigma_1 & \sigma_3 \\ \hline \bar{\sigma}_0 & 1 & 0 & 1 & 0 \\ \bar{\sigma}_1 & 0 & 1 & 1 & 1 \\ \bar{\sigma}_2 & 0 & 0 & 1 & 0 \\ \bar{\sigma}_3 & 0 & 0 & 0 & 1 \end{array}, \quad T' = \begin{array}{c|cccc} & \bar{\sigma}_0 & \bar{\sigma}_2 & \bar{\sigma}_1 & \bar{\sigma}_3 \\ \hline \sigma_0 & 1 & 0 & 1 & 0 \\ \sigma_1 & 0 & 1 & 1 & 1 \\ \sigma_2 & 0 & 0 & 1 & 0 \\ \sigma_3 & 0 & 0 & 0 & 1 \end{array}. \end{array}$$

There are three 1-dimensional persistent homology classes, (τ_2, σ_0) , (τ_0, σ_3) and (τ_1, σ_1) (resp. (τ_1, σ_2) after perturbation). See Figure 6.3 for the reduction edges (solid) and inclusion edges (dashed) drew on the persistent diagram. The diagram before perturbation is on the left and after perturbation is on the right. It is shown that the graph is disconnected due to the geometric perturbation.

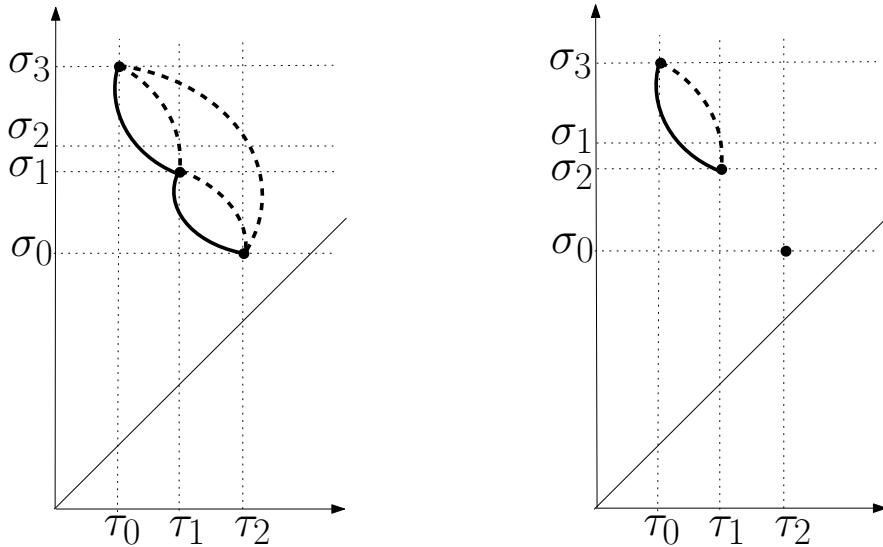


Figure 6.3: Persistent diagrams with reduction/inclusion edges before and after the perturbation.

6.4 Updating Total Persistent Homology

For completeness, we conclude this chapter with an algorithm which updates the total persistence when a transposition of the simplex-ordering happens. We discuss cases one by one, following the scheme of [28]. We show that the update algorithm takes $O(N)$ time ($N = \text{card}(K)$) except for Case 2.2 ($O(N^3)$) and Case 3.1 ($O(N^2)$). Although in the worst case, the update algorithm is as expensive as recomputing completely, this case by case discussion helps us analyze various proposed enriched persistence definitions and discover their instability cases. We believe it will be useful for further study of enriched persistence.

Suppose we transpose simplices σ_n and σ_{n+1} . The chains $\bar{\sigma}_n$, $\bar{\sigma}_{n+1}$, as well as any $\bar{\sigma}_m$ for $m > n + 1$ might change. Denote the new simplices $\sigma'_n = \sigma_{n+1}$ and $\sigma'_{n+1} = \sigma_n$. We derive new chains $\bar{\sigma}'_m$ for all $m \geq n$ ($\bar{\sigma}_m$ remains the same for $m < n$).

Recall that in total persistences, we write any $\partial\sigma_n$ as the sum of a nonbounding part and boundary part, and then define chain $\bar{\sigma}_n$ accordingly.

$$\begin{aligned}\partial\sigma_n &= \sum_{i=1}^{n-1} \lambda_i^n \bar{\sigma}_i + \sum_{j=1}^{n-1} \mu_j^n \partial\bar{\sigma}_j \\ \bar{\sigma}_n &= \sigma_n + \sum_{j=1}^{n-1} \mu_j^n \bar{\sigma}_j\end{aligned}$$

Case 0: $\mu_n^{n+1} = 1$, $\lambda_n^{n+1} = 1$.

This case will not happen as $\mu_n^{n+1} = 1$ iff σ_n has the same dimension as σ_{n+1} , whereas $\lambda_n^{n+1} = 1$ iff the dimensions of σ_n and σ_{n+1} differ by 1.

Case 1: $\mu_n^{n+1} = 0, \lambda_n^{n+1} = 1$.

In this case, as σ_n and σ_{n+1} are consecutive in the simplex-ordering, $\sigma_n \in \partial\sigma_{n+1}$. They cannot be transposed.

Case 2: $\mu_n^{n+1} = 0, \lambda_n^{n+1} = 0$.

Case 2.1: $\sigma_n-, \sigma_{n+1}-, \dim(\sigma_n) = \dim(\sigma_{n+1}) = p, \lambda_{i_{n+1}}^n = 1$.

Denote $\sigma_{i_n} = \text{low}(\sigma_n)$, $\sigma_{i_{n+1}} = \text{low}(\sigma_{n+1})$. The submatrix of R specified by columns $\bar{\sigma}_n$ and $\bar{\sigma}_{n+1}$ and rows $\bar{\sigma}_{i_n}$ and $\bar{\sigma}_{i_{n+1}}$ is
$$\left[\begin{array}{c|cc} & \bar{\sigma}_n & \bar{\sigma}_{n+1} \\ \hline \bar{\sigma}_{i_{n+1}} & 1 & 1 \\ \bar{\sigma}_{i_n} & 1 & 0 \end{array} \right]$$
. After transposition, we switch column n and $n+1$ of R , the new column n remains fully reduced, $\bar{\sigma}'_n = \bar{\sigma}_{n+1}$. But we need to use it to reduce the new column $n+1$, $\bar{\sigma}'_{n+1} = \bar{\sigma}_n + \bar{\sigma}_{n+1}$. For any $m > n+1$, the nonbounding part of $\partial\sigma_m$ remains unchanged as the $(p-1)$ -dimensional homology cycle basis of $H_{p-1}(K_{m-1})$ is the same, namely,

$$\begin{aligned} \partial\bar{\sigma}'_m &= \partial\bar{\sigma}_m \\ &= \sum_{j \neq n, n+1, m} \mu_j^m \partial\bar{\sigma}_j + \mu_n^m \partial\bar{\sigma}_n + \mu_{n+1}^m \partial\bar{\sigma}_{n+1} \\ &= \sum_{j \neq n, n+1, m} \mu_j^m \partial\bar{\sigma}_j + (\mu_n^m + \mu_{n+1}^m) \partial\bar{\sigma}'_n + \mu_n^m \partial\bar{\sigma}'_{n+1} \end{aligned}$$

Therefore, $\bar{\sigma}'_m = \bar{\sigma}_m$. The new matrices are $T' = PTQ$, $(T^{-1})' = Q^{-1}T^{-1}P$, $R' = Q^{-1}RQ$, where Q , Q^{-1} and P are listed in Table 6.2.

As the update algorithm needs $O(1)$ column/row additions or switchings, the time for this case is $O(N)$. The other cases are the same except for Case 2.2 and Case 3.1.

Table 6.2: Matrices for updating total persistence.

$$Q = \begin{pmatrix} & & \\ & 0 & 1 \\ & 1 & 1 \end{pmatrix} \quad Q^{-1} = \begin{pmatrix} & & \\ & 1 & 1 \\ & 1 & 0 \end{pmatrix} \quad P = \begin{pmatrix} & & \\ & 0 & 1 \\ & 1 & 0 \end{pmatrix} \quad L = \begin{pmatrix} & & \\ & 1 & 0 \\ & 1 & 1 \end{pmatrix}$$

Case 2.2: $\sigma_n+, \sigma_{n+1}+, \dim(\sigma_n) = \dim(\sigma_{n+1}) = p, \lambda_n^{i_{n+1}} = 1$.

Denote $\sigma_n = \text{low}(\sigma_{i_n})$, $\sigma_{n+1} = \text{low}(\sigma_{i_{n+1}})$ (assuming σ_n and σ_{n+1} are paired). The submatrix of R specified by columns $\bar{\sigma}_{i_n}$ and $\bar{\sigma}_{i_{n+1}}$ and rows $\bar{\sigma}_n$ and $\bar{\sigma}_{n+1}$ is

$$\left[\begin{array}{c|cc} & \bar{\sigma}_{i_{n+1}} & \bar{\sigma}_{i_n} \\ \hline \bar{\sigma}_n & 1 & 1 \\ \bar{\sigma}_{n+1} & 1 & 0 \end{array} \right]$$
. After transposition, $\bar{\sigma}'_n = \bar{\sigma}_{n+1}$, $\bar{\sigma}'_{n+1} = \bar{\sigma}_n$, $\bar{\sigma}'_{i_{n+1}} = \bar{\sigma}_{i_{n+1}}$.

For any $m > i_{n+1}$ (including $m = i_n$) and $\dim(\sigma_m) = p + 1$, if σ_m+ , similar to Case 2.1, $\bar{\sigma}'_m = \bar{\sigma}_m$. However, if σ_m- , we should use the totally reduced column i_{n+1} to reduce the column m ,

$$\begin{aligned}\partial\sigma_m &= \sum_{i<m}^{i \neq n, n+1} \lambda_i^m \bar{\sigma}_i + \lambda_n^m \bar{\sigma}_n + \sum_{j=1}^{m-1} \mu_j^m \partial\bar{\sigma}_j \\ &= \sum_{i<m}^{i \neq n, n+1} (\lambda_i^m + \lambda_n^m \lambda_i^{i_{n+1}}) \bar{\sigma}_i + \lambda_n^m \bar{\sigma}'_n + \sum_{j=1}^{m-1} \mu_j^m \partial\bar{\sigma}_j + \lambda_n^m \partial\bar{\sigma}'_{i_{n+1}}\end{aligned}$$

However, this reduction may introduce new entries into the column and require further reduction. Specifically, we need to reduce the new entries that are not paired by time i_{n+1} but are paired by time $m - 1$. This process is complicated and as far as we know, could not be written as matrix multiplication. The complexity is $O(N^3)$.

Case 2.3: Other Cases.

Simply switch the corresponding rows and columns of the matrices. Formally, $(T^{-1})' = PT^{-1}P$, $R' = PRP$, $T' = PTP$.

Case 3: $\lambda_n^{n+1} = 0$, $\mu_n^{n+1} = 1$, σ_n- , $\dim(\sigma_n) = \dim(\sigma_{n+1}) = p$.

Case 3.1: $\sigma_{n+1}+$.

Before the transposition, σ_{n+1} is totally reduced using $\bar{\sigma}_n$. Similar to standard persistence, after transposition, σ'_n- and $\sigma'_{n+1}+$,

$$\begin{aligned}\bar{\sigma}'_{n+1} &= \bar{\sigma}_{n+1} \\ \bar{\sigma}'_n &= \bar{\sigma}_n + \bar{\sigma}_{n+1} \\ \partial\bar{\sigma}'_n &= \partial\bar{\sigma}_n\end{aligned}$$

Due to the first equation, for any $m > n + 1$, if $\dim(\sigma_m) = p + 1$, $\bar{\sigma}'_m = \bar{\sigma}_m$. If $\dim(\sigma_m) = p$,

$$\begin{aligned}\partial\sigma_m &= \sum_{i=1}^{i < m} \lambda_i^m \bar{\sigma}_i + \sum_{j=1}^{j \neq n} \mu_j^m \partial\bar{\sigma}_j + \mu_n^m \partial\bar{\sigma}_n \\ &= \sum_{i=1}^{i < m} \lambda_i^m \bar{\sigma}_i + \sum_{j=1}^{j \neq n} \mu_j^m \partial\bar{\sigma}_j + \mu_n^m \partial\bar{\sigma}'_n \\ \partial\bar{\sigma}'_m &= \partial\bar{\sigma}_m\end{aligned}$$

Iff $\sigma_n \in \bar{\sigma}_m$, we replace $\bar{\sigma}_n$ with $\bar{\sigma}'_n = \bar{\sigma}_n + \bar{\sigma}_{n+1}$. Therefore, $R' = R$. $\bar{\sigma}'_m = \bar{\sigma}_m + T_{n,m} \bar{\sigma}_{n+1}$. Intuitively, for any column $T_{*,m}$, if the n -th entry is 1, $T'_{*,m} = T_{*,m} + T_{*,n+1}$.

Formally, $T' = PTG$ where G has 1's on its diagonals and $G_{n+1,*} = T_{n,*}$ (note G is not upper triangular as $G_{n+1,n} = 1$). Consequently $(T^{-1})' = GT^{-1}P$ as $G^{-1} = G$. In this case, we need $O(N)$ column additions. The time complexity is $O(N^2)$.

Case 3.2: $\sigma_{n+1}-$.

After transposition, the new column n should be the sum of n and $n+1$ column of R , whereas the new column $n+1$ should be the n -th or the $(n+1)$ -th column of R .

$$\begin{aligned}\partial\sigma'_n &= \partial\sigma_{n+1} \\ &= \sum_{i=1}^{n-1} \lambda_i^{n+1} \bar{\sigma}_i + \partial\bar{\sigma}_n + \sum_{j=1}^{n-1} \mu_j^{n+1} \partial\bar{\sigma}_j \\ &= \sum_{i=1}^{n-1} (\lambda_i^{n+1} + \lambda_j^n) \bar{\sigma}_i + \sum_{j=1}^{n-1} \mu_j^{n+1} \partial\bar{\sigma}_j \\ \partial\bar{\sigma}'_n &= \partial\bar{\sigma}_n + \partial\bar{\sigma}_{n+1} \\ \bar{\sigma}'_n &= \bar{\sigma}_n + \bar{\sigma}_{n+1}\end{aligned}$$

Denote $\sigma_{i_n} = \text{low}(\sigma_n)$, $\sigma_{i_{n+1}} = \text{low}(\sigma_{n+1})$. Then $\text{low}(\sigma'_n) = \max\{\sigma_{i_n}, \sigma_{i_{n+1}}\}$.

Case 3.2.1: $i_n > i_{n+1}$.

The submatrix of R specified by columns $\bar{\sigma}_n$ and $\bar{\sigma}_{n+1}$ and rows $\bar{\sigma}_{i_n}$ and $\bar{\sigma}_{i_{n+1}}$ is $\left[\begin{array}{c|cc} & \bar{\sigma}_n & \bar{\sigma}_{n+1} \\ \hline \bar{\sigma}_{i_{n+1}} & 0/1 & 1 \\ \bar{\sigma}_{i_n} & 1 & 0 \end{array} \right]$. After transposition, the new column $n+1$ is the same as the original $(n+1)$ -th column of R . Furthermore, $\bar{\sigma}'_{n+1} = \bar{\sigma}_{n+1}$. For any $m > n+1$,

$$\begin{aligned}\bar{\sigma}'_m &= \bar{\sigma}_m \\ &= \sigma_m + \sum_{j \neq n, n+1} \mu_j^m \bar{\sigma}_j + \mu_n^m \bar{\sigma}'_n + (\mu_n^m + \mu_{n+1}^m) \bar{\sigma}'_{n+1} \\ &= \sigma_m + \sum_{j < m} \mu_j^m \bar{\sigma}_j + \mu_n^m \bar{\sigma}'_n + (\mu_n^m + \mu_{n+1}^m) \bar{\sigma}'_{n+1}\end{aligned}$$

Therefore, $(T^{-1})' = UT^{-1}P$, $T' = PTU$, $R' = URU$, where U is defined in Table 6.2

Case 3.2.2: $i_n < i_{n+1}$.

The submatrix of R specified by columns $\bar{\sigma}_n$ and $\bar{\sigma}_{n+1}$ and rows $\bar{\sigma}_{i_n}$ and $\bar{\sigma}_{i_{n+1}}$ are $\left[\begin{array}{c|cc} & \bar{\sigma}_n & \bar{\sigma}_{n+1} \\ \hline \bar{\sigma}_{i_n} & 1 & 0 \\ \bar{\sigma}_{i_{n+1}} & 0 & 1 \end{array} \right]$. After transposition, the new column $n+1$ of R should

be the n -th column. Furthermore, $\bar{\sigma}'_{n+1} = \bar{\sigma}_n$. For any $m > n + 1$,

$$\begin{aligned}\bar{\sigma}'_m &= \bar{\sigma}_m \\ &= \sigma_m + \sum_{j \neq n, n+1} \mu_j^m \bar{\sigma}_j + \mu_{n+1}^m \bar{\sigma}'_n + (\mu_n^m + \mu_{n+1}^m) \bar{\sigma}'_{n+1}\end{aligned}$$

Therefore, $(T^{-1})' = QT^{-1}P$, $T' = PTQ^{-1}$, $R' = QRQ^{-1}$.

CHAPTER 7

Application of Computational Topology Methods in Ribosome Drug Docking

In this chapter, we will talk about an application of computational topology in biology. We first explain the ribosome drug docking problem. Second, we introduce the idea of using pockets to study the docking. Results will be explained. At last, we further discuss the geometry of the binding sites and talk about possibilities of using other topological features (2-dimensional critical points of the distance function).

7.1 Problem: Ribosome Drug Docking

Ribosomes are big complexes of RNA chains and proteins. They play important roles in life science as the “factories” of proteins. They synthesize proteins by decoding genetic information from messenger RNAs (mRNA), and accordingly, assembling amino acids delivered by transfer RNAs (tRNA). They exist in both eukaryote (including human) and prokaryote (including bacteria).

Ribosomes consist of two different subunits, a small one and a big one. They bind together to perform protein synthesis. In prokaryote (bacteria) the small subunit, big subunit and their composition are 30s, 50s and 70s, respectively, measured in the Svedberg unit (rates of sedimentation). The 50s subunit consists of a 5s RNA chain (120 nucleotides), a 23s RNA chain (2900 nucleotides) and 34 proteins. It has roughly 90000 atoms all together. The 30s subunit has a 16s RNA (1540 nucleotides) and 21 proteins (roughly 51000 atoms all together). Ribosomes in eukaryote (humans) are bigger (a 40s and a 60s form a 80s).

Although ribosomes consist of 65% of RNA and 35% of proteins, all their biological activities are carried out by the RNA, whereas the proteins reside on the surface and stabilize the geometric structure. The function area consists of three tRNA-binding sites, the A (aminoacyl), P (peptidyl), and E (exit) sites. The tRNA carrying amino-acid is first bound in A site, then transferred to the P site where amino-acid is consumed and finally the E site where it leaves the ribosome. The region near the P site is called the PTC (peptidyl-transfer center). Please see Figure 7.1 for example.

Despite the size difference of ribosomes in bacteria and human. They share a similar core structure and functioning mechanism. The key of antibiotic discovery is finding antibiotics that could malfunction bacteria ribosomes but not human ribosomes due to their subtle difference in both geometry and chemistry.

Due to their ubiquitousness and their important biological role, ribosomes have been one of the main antibiotic docking targets [90, 55, 61]. However, programs facilitating ribosome drug docking are still in a preliminary stage due to the following reasons.

1. Existing scoring function for proteins (with 70% – 90% success rate) does not

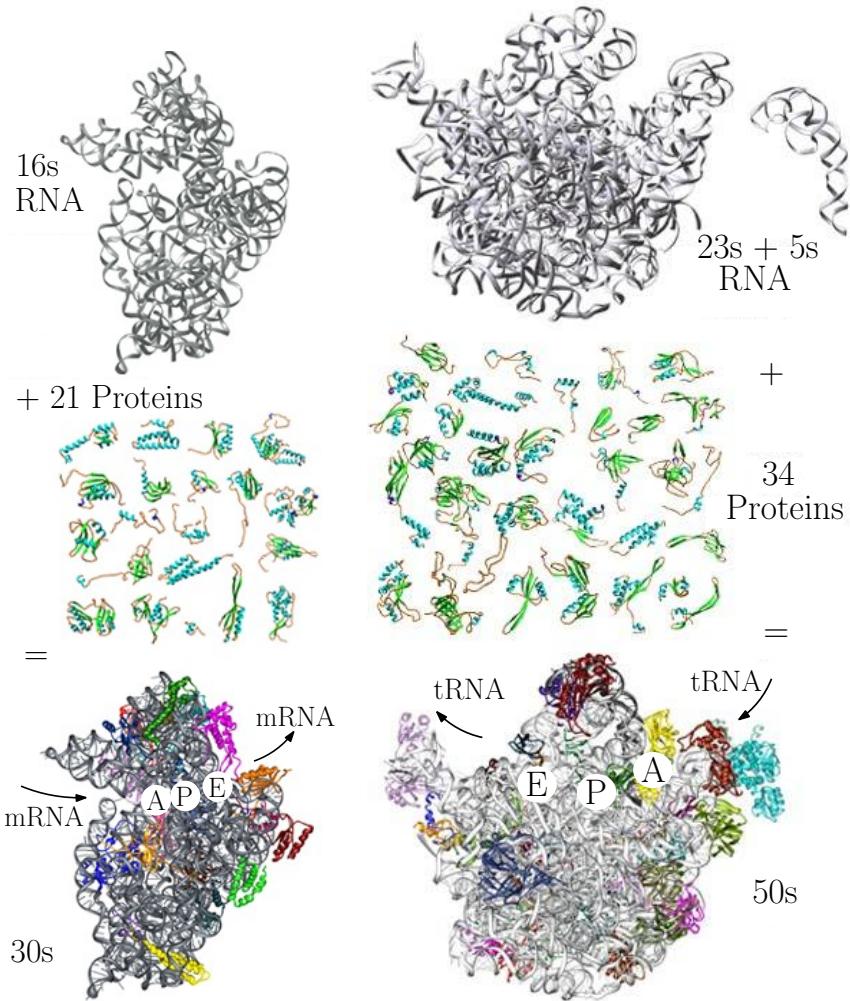


Figure 7.1: 30s and 50s ribosomes. RNAs are in grey color and proteins are in other colors. A, P and E sites are pointed out.
The pictures are from <http://www.riboworld.com/>.

fit RNAs due to the difference of their geometric and chemical properties.

2. Energy oriented method like minimizing root mean square error is computationally forbidden in terms of finding new candidate binding sites, due to the extremely big size and complicated surface of ribosome.

7.2 Method: Applying Computational Topology Methods on Ribosome Drug Docking

Whereas traditional drug docking methods are challenged, computational topology based docking method seems promising in the ribosome drug docking scenario. There are two justifications.

1. The geometric structure of RNA is different from proteins. The chain is loosely packed with a lot of holes and tunnels. These topological features are naturally signatures of drug binding sites.
2. RNA in ribosome is relatively more stable. Therefore, rigid shape analysis of the structure is more meaningful than small RNA molecules.

We study ribosome drug docking sites using a well developed computational topology tool from protein drug docking, namely, the pockets. Edelsbrunner et al. [45] defined pockets using the alpha complexes of molecules. These pockets are computed and measured for the alpha complex of a molecule with zero alpha value.

7.2.1 Alpha Complexes

A biological molecule could be described as a union of balls,

$$\begin{aligned}\bigcup \mathcal{B} &= \bigcup \{B_{p_i}^{r_i}\} \\ &= \{x \in \mathbb{R}^3 \mid \exists i, \|x - p_i\| \leq r_i\}\end{aligned}$$

Furthermore, we describe the union of balls with an additional parameter, α ,

$$\bigcup \mathcal{B}_\alpha = \{x \in \mathbb{R}^3 \mid \exists i, \|x - p_i\|^2 \leq r_i^2 + \alpha\}.$$

For different α , the Voronoi and Delaunay complexes defined with the distance function

$$d(x, \mathcal{B}_\alpha) = \min_i (\|x - p_i\|^2 - r_i^2 - \alpha).$$

remain the same, denoted as V and D . An alpha complex with a specified alpha value is a subcomplex of the Delaunay complex defined as

$$K(\alpha) = \{\sigma_i \in D \mid \nu_i \cap \bigcup \mathcal{B}_\alpha \neq \emptyset\},$$

in which ν_i is the Voronoi simplex which is dual to σ_i . As α increases from $-\infty$ to $+\infty$, the alpha complex grows from empty to the whole Delaunay complex, forming a filtration

$$\emptyset = K(-\infty) \subseteq K(\alpha_1) \subseteq K(\alpha_2) \subseteq \cdots \subseteq K(+\infty) = D$$

7.2.2 Pockets and Cavities

The *cavity* and *pocket* are defined with $K(0)$ although they could naturally be extended to any other alpha value. Intuitively, a cavity is the space enclosed by the union of balls, corresponding to a bounded component of $\mathbb{R}^3 \setminus \bigcup \mathcal{B}_\alpha$. A pocket is a dent/hole of the molecule which will become one or a set of cavities if we increase α to a certain positive value. Existing software has been provided to calculate and measure cavities and pockets [69].

Whereas standard definitions of cavities and pockets describe drug docking sites well, they are not appropriate for describing ribosome docking sites. The reason is that RNA is loosely packed in 3D space. The ribosome with $\alpha = 0$ only has one big pocket, which too big to describe any drug binding site.

In this work, we extend the definition of pockets and cavities to any nonnegative alpha values. The idea is as we increase the alpha value, the molecule grows and big pockets and cavities may evolve into smaller ones. We want to capture these smaller units as they might be specific enough to describe docking sites.

Our definition is based on the flow graph of the tetrahedra used to find standard pockets and cavities. In the Delaunay complex D , we say a triangle is *attached* to a tetrahedron when the circumcenter and the other vertex of the tetrahedron are in different sides of the plane defined by the triangle. If the triangle is not on the convex hull of D , there is another tetrahedron on the other side of the triangle and we say the first tetrahedron *flow* to the second. Please note that the second tetrahedron could flow to some other tetrahedron but not the first one, due to the properties of Delaunay triangulation. We use a node to represent each tetrahedron and a directed edge connecting two nodes if the first one flow to the second one. An extra node at infinity is introduced. A tetrahedron attached by a triangle on the convex hull is said to flow to the infinity node. The set of nodes and flow edges form a directed graph. Except the infinite node, we define pockets using each sink, namely, the node with no out-degree⁷, and all tetrahedra that has a directed patch connecting to it. We define the pocket/cavity as the molecule surface inside this group of tetrahedra. This provides a subdivision of a big pocket (in the traditional definition) into small ones.

Please see Figure 7.2. There are 4 pockets in the example, defined by the 4 sinks, t_1 , t_2 , t_3 and t_4 . But according to the standard definition, there are only two pockets.

7.2.3 Results

In experiments, we capture all pockets with regard to the new definition and observe the ones near the drug binding sites. There are two main results.

The new definition could provide insights of the ribosome structure. By grouping pockets which get disconnected with bigger alpha values, we are able to describe important functioning regions of ribosomes and provides measurements.

The computed pockets with bigger alpha values provide insights of ribosome structure. More specifically, these pockets/cavities successfully present tunnels and channels of ribosome through which mRNAs, tRNAs, and polypeptide chains move during the protein synthesis. See Figures 7.3 and 7.4. In the 50s, we isolated two big functional regions. One is the channel through which tRNA moves. All the A, P and E sites are in this channel. Another region is the polypeptide exit tunnel, through which the synthesized protein leave. In the 30s, we isolated the channel through which the mRNA move during the decoding.

⁷Sinks are 3-dimensional critical points of the distance function.

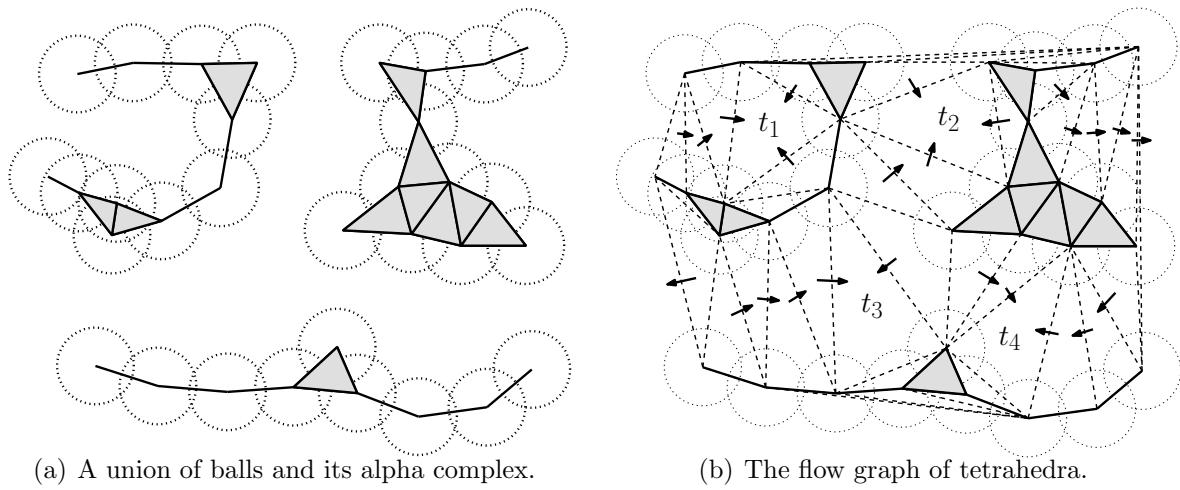


Figure 7.2: An example of pockets.

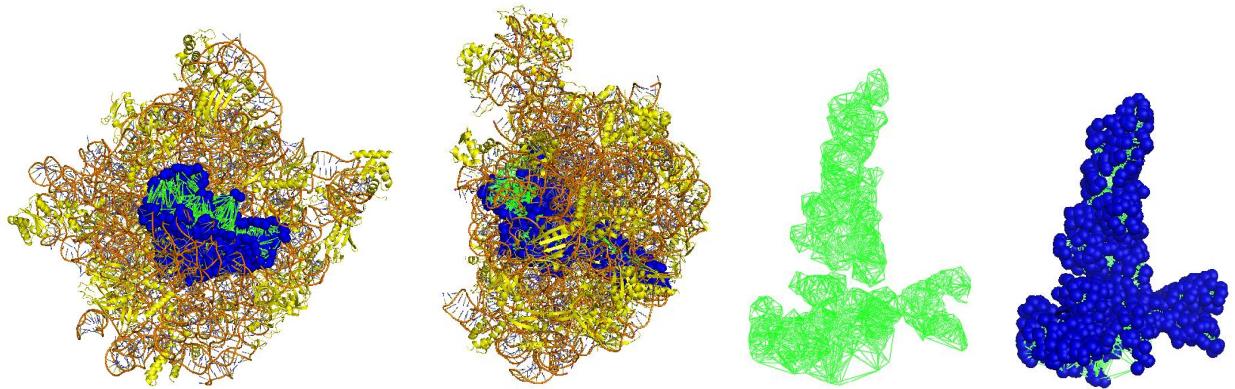


Figure 7.3: 50s and the two functional regions we isolated.

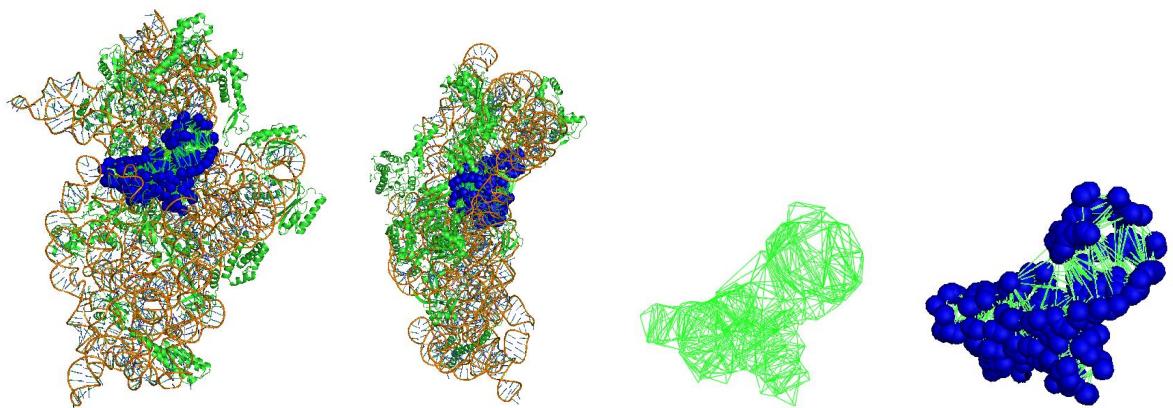


Figure 7.4: 30s and the functional region we isolated.

However, experiments reveal that the drug docking sites are actually not well described as pockets. Essentially, this new definition is to capture pockets of the molecule when we grow it to some extend. However, if we keep increasing the alpha value, we do get smaller pockets, which are essentially the centers of those tunnels and channels. These small pockets/cavities are quite far away from the surface of ribosome, and thus far away from the drug binding sites. In other words, with big alpha values, the drug binding sites have been smoothed out, which means, have been buried in the grown molecule.

As the observation revealed, the binding sites are more related to 2-dimensional critical points of the distance function, rather than 3-dimensional critical points corresponding to the pockets. Further details will be discussed in next section.

7.3 Discussion of Binding Sites

In this section, we further discuss the geometry and topology of binding sites.

7.3.1 Data

We first collect all available ribosome drug binding complexes from the protein data bank [79]. Available complexes include 30s, 50s and 70s. However, observation reveals that for 50s, the drug binding sites are concentrated around the peptidyl transferase center (PTC) and the entrance of the polypeptide exit tunnel. Therefore, we will mainly focus on 30s and 70s due to the variety of its binding sites in terms of geometry and location. The number of different drugs bind with 30s ribosomes are listed in table (7.1). A detailed list is in table (7.2). At this moment, we focus on complexes in which the whole RNA is present. Complexes with RNA fragments will be included later.

| System | T. Thermophilus | E. Coli | T. T. or E. C. |
|---------------------------|-----------------|---------|----------------|
| Number of different drugs | 10 | 6 | 12 |

Table 7.1: Numbers of different drugs binding to 30s or 70s ribosome.

7.3.2 Shapes of binding sites

Observation reveals that grooves/tunnels on the surface of ribosome with appropriate size can be potential binding sites. The size has to be right so that the groove has enough contact with the drug to trap it. For example, grooves in 30S are bound by drugs like aminoglycosides, whereas the polypeptide exit tunnel is bound by larger drugs like macrolides [55]. For 30s, this kind of binding sites appear in 8 out of 12 drugs. See Figure 7.5

Grooves and holes are closely related to critical points of the distance function. Please note that here to get sites that just fits the drugs, we need critical points with appropriate distance function value. Sticking with large distance values will

| Drug ID | Drug Name | Drug Family | Binding Sites | System |
|---------|---------------|-----------------|----------------|---------------|
| PAR | Paromomycin | Aminoglycosides | 1 | T.T. and E.C. |
| SRY | Streptomycin | Aminoglycosides | 1 | T.T. |
| SCM | Spectinomycin | Macrolides | 1 | T.T. and E.C. |
| HYG | Hygromycin B | | 1 | T.T. and E.C. |
| PCY | Pactamycin | | 1 | T.T. |
| TAC | Tetracycline | Tetracyclines | 3 | T.T. |
| EDE | Edeine B | | 1 | T.T. |
| AB9 | Design | | 1 | T.T. |
| D2C | Design | | 1 | T.T. |
| KSG | Kasugamycin | | 3 (very close) | T.T. and E.C. |
| LLL | Gentamicin | Aminoglycosides | 1 | E.C. |
| NMY | Neomycin | Aminoglycosides | 1 | E.C. |

Table 7.2: List of drugs.

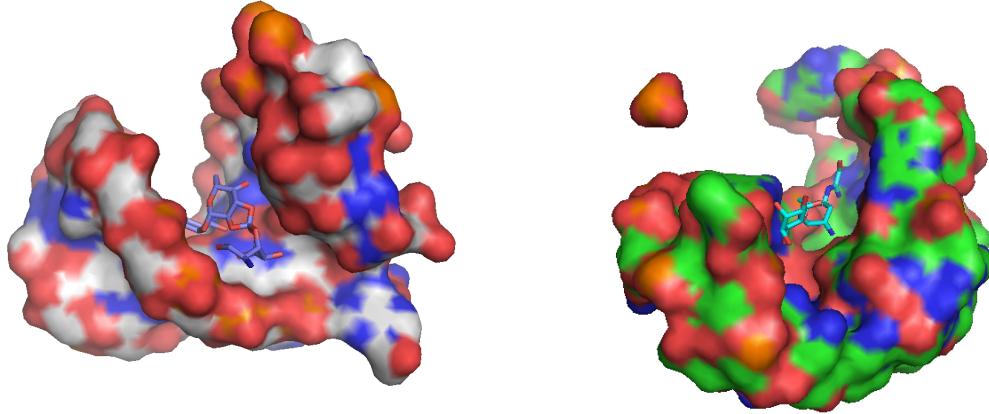


Figure 7.5: Hygromycin B (left) and Kasugamycin (right) are bound in grooves with the right width.

give us nothing more than what the aforementioned pocket method does. There are two difficulties with this idea.

1. **Complicated surface introduces noise.** Due to the complicated surface, it might be difficult to distinguish candidate critical points from noise, especially when we are looking for the ones with appropriate distance. Persistent homology might help.
2. **Flexibility of RNA has to be considered.** In several cases (aminoglycosides in E. Coli system and edeine B), the flexibility of RNA plays an important role. In the unbound structure, flexible side chains can be in wrong positions and hinder the discovery of interesting grooves/tunnels. When we go through

the ribosome surface and look for grooves and tunnels as good binding site candidates, flexible side-chains should be dealt with carefully. More specifically, we might consider a piece of helix as a good candidate if we can get a groove by rotating a few flexible side-chains. See Figure 7.6.

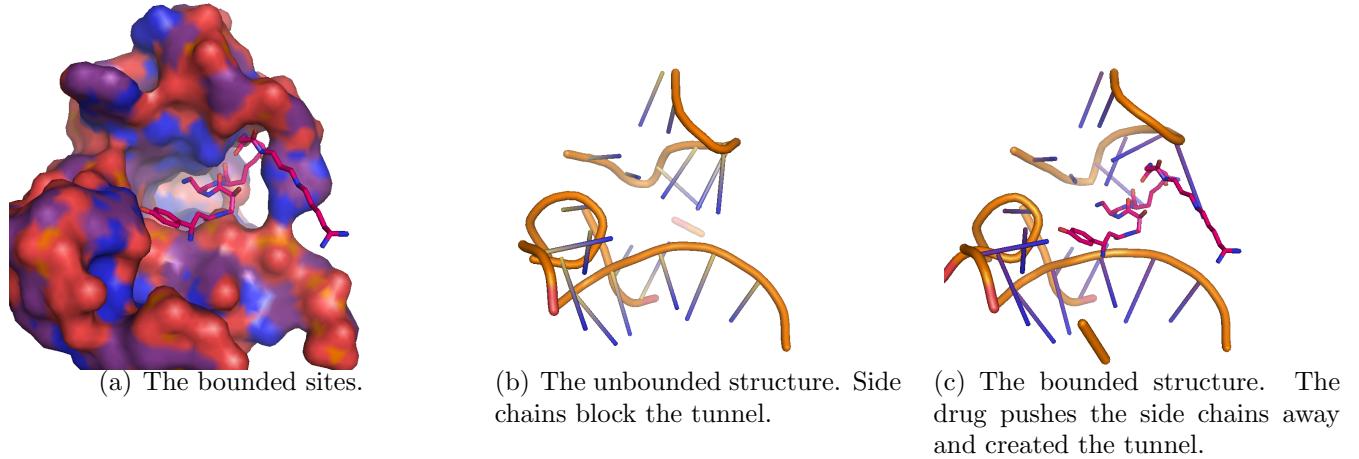


Figure 7.6: Edeine B is bound in a tunnel, after pushing a few side chains away.

REFERENCES

- [1] Pankaj K. Agarwal, Herbert Edelsbrunner, John Harer, and Yusu Wang. Extreme elevation on a 2-manifold. *Discrete & Computational Geometry*, 36:553–572, 2006.
- [2] Nina Amenta and Marshall W. Bern. Surface reconstruction by voronoi filtering. *Discrete & Computational Geometry*, 22(4):481–504, 1999.
- [3] Nina Amenta, Sunghee Choi, Tamal K. Dey, and N. Leekha. A simple algorithm for homeomorphic surface reconstruction. *Int. J. Comput. Geometry Appl.*, 12(1-2):125–141, 2002.
- [4] Esther M. Arkin and Refael Hassin. Minimum-diameter covering problems. *Networks*, 36(3):147–155, 2000.
- [5] Sanjeev Arora, László Babai, Jacques Stern, and Z. Sweedyk. The hardness of approximate optima in lattices, codes, and systems of linear equations. *J. Comput. Syst. Sci.*, 54(2):317–331, 1997.
- [6] Dominique Attali, Herbert Edelsbrunner, and Yuriy Mileyko. Weak witnesses for delaunay triangulations of submanifolds. In *Symposium on Solid and Physical Modeling*, pages 143–150, 2007.
- [7] Giorgio Ausiello and Vangelis Th. Paschos. Reductions, completeness and the hardness of approximability. *European Journal of Operational Research*, 172(3):719–739, 2006.
- [8] Jean-Daniel Boissonnat, Leonidas J. Guibas, and Steve Oudot. Manifold reconstruction in arbitrary dimensions using witness complexes. In *Proceedings of the 23rd ACM Symposium on Computational Geometry*, pages 194–203, 2007.
- [9] Francesca Cagliari, Massimo Ferri, and Paola Pozzi. Size functions from a categorical viewpoint. *Acta Applicandae Mathematicae*, 67(3):225–235, 2001.
- [10] Erik Carlsson, Gunnar Carlsson, and Vin de Silva. An algebraic topological method for feature identification. *Int. J. Comput. Geometry Appl.*, 16(4):291–314, 2006.
- [11] Gunnar Carlsson. Persistent homology and the analysis of high dimensional data. Fields-Ottawa Workshop on the Geometry of Very Large Data Sets, February 2005.
- [12] Gunnar Carlsson, Tigran Ishkhanov, Vin de Silva, and Leonidas J. Guibas. Persistence barcodes for shapes. *International Journal of Shape Modeling*, 11(2):149–188, 2005.

- [13] Gunnar Carlsson, Tigran Ishkhanov, Vin de Silva, and Afra Zomorodian. On the local behavior of spaces of natural images. *International Journal of Computer Vision*, 2007.
- [14] Gunnar Carlsson and Afra Zomorodian. The theory of multidimensional persistence. In *Proceedings of the 23rd ACM Symposium on Computational Geometry*, pages 184–193, 2007.
- [15] Christopher Carner, Miao Jin, Xianfeng Gu, and Hong Qin. Topology-driven surface mappings with robust feature alignment. In *Proceedings of the 16th IEEE Visualization Conference*, pages 543–550, 2005.
- [16] Erin W. Chambers, Éric Colin de Verdière, Jeff Erickson, Francis Lazarus, and Kim Whittlesey. Splitting (complicated) surfaces is hard. In *Proceedings of the 22nd ACM Symposium on Computational Geometry*, pages 421–429, 2006.
- [17] Erin W. Chambers, Jeff Erickson, and Amir Nayyeri. Homology flows, cohomology cuts. In *STOC*, 2009.
- [18] Erin W. Chambers, Jeff Erickson, and Amir Nayyeri. Minimum cuts and shortest homologous cycles. In *Symposium on Computational Geometry*, 2009.
- [19] Frédéric Chazal, David Cohen-Steiner, and André Lieutier. A sampling theory for compact sets in euclidean space. In *Proceedings of the 22nd ACM Symposium on Computational Geometry*, pages 319–326, 2006.
- [20] Frédéric Chazal and André Lieutier. Weak feature size and persistent homology: computing homology of solids in r^n from noisy data samples. In *Proceedings of the 21st ACM Symposium on Computational Geometry*, pages 255–262, 2005.
- [21] Frédéric Chazal and André Lieutier. Topology guaranteeing manifold reconstruction using distance function to noisy data. In *Proceedings of the 22nd ACM Symposium on Computational Geometry*, pages 112–118, 2006.
- [22] Chao Chen and Daniel Freedman. Quantifying homology classes ii: Localization and stability. *CoRR*, abs/0709.2512, 2007.
- [23] Siu-Wing Cheng, Tamal K. Dey, and Edgar A. Ramos. Manifold reconstruction from point samples. In *Proceedings of the 16th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1018–1027, 2005.
- [24] David Cohen-Steiner and Herbert Edelsbrunner. Inequalities for the curvature of curves and surfaces. In *Proceedings of the 21st ACM Symposium on Computational Geometry*, pages 272–277, 2005.
- [25] David Cohen-Steiner, Herbert Edelsbrunner, and John Harer. Stability of persistence diagrams. *Discrete & Computational Geometry*, 37:103–120, 2007.

- [26] David Cohen-Steiner, Herbert Edelsbrunner, and John Harer. Extending persistent homology using poincaré and lefschetz duality. *Foundations of Computational Mathematics*, 9(1):79–103, 2009.
- [27] David Cohen-Steiner, Herbert Edelsbrunner, John Harer, and Dmitriy Morozov. Persistent homology for kernels, images, and cokernels. In *SODA*, pages 1011–1020, 2009.
- [28] David Cohen-Steiner, Herbert Edelsbrunner, and Dmitriy Morozov. Vines and vineyards by updating persistence in linear time. In *Proceedings of the 22nd ACM Symposium on Computational Geometry*, pages 119–126, 2006.
- [29] Éric Colin de Verdière and Jeff Erickson. Tightening non-simple paths and cycles on surfaces. In *Proceedings of the 17th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 192–201, 2006.
- [30] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms*. MIT Press, 2001.
- [31] Vin de Silva. Persistent cohomology. Workshop on Application of Topology in Science and Engineering, September 2006. MSRI.
- [32] Vin de Silva and Gunnar Carlsson. Topological estimation using witness complexes. In *Proc. Sympos. Point-Based Graphics*, pages 157–166, 2004.
- [33] Vin de Silva and Robert Ghrist. Coverage in sensor networks via persistent homology. *Algebraic & Geometric Topology*, 7:339–358, 2007.
- [34] Cecil Jose A. Delfinado and Herbert Edelsbrunner. An incremental algorithm for betti numbers of simplicial complexes. In *Proceedings of the 9th Annual Symposium on Computational Geometry*, pages 232–239, 1993.
- [35] Mathieu Desbrun, Eva Kanso, and Yiyi Tong. Discrete differential forms for computational modeling. In *SIGGRAPH ’06: ACM SIGGRAPH 2006 Courses*, pages 39–54, New York, NY, USA, 2006. ACM Press.
- [36] Tamal K. Dey and Samrat Goswami. Provable surface reconstruction from noisy samples. *Comput. Geom.*, 35(1-2):124–141, 2006.
- [37] Tamal K. Dey, Kuiyu Li, and Jian Sun. On computing handle and tunnel loops. In *IEEE Proc. NASAGEM*, 2007.
- [38] Tamal K. Dey, Kuiyu Li, Jian Sun, and David Cohen-Steiner. Computing geometry-aware handle and tunnel loops in 3d models. *ACM Trans. Graph.*, 27(3), 2008.
- [39] Jean-Guillaume Dumas, Pascal Giorgi, and Clément Pernet. Ffpack: finite field linear algebra package. In *International Symposium on Symbolic and Algebraic Computation*, pages 119–126, 2004.

- [40] Jean-Guillaume Dumas, Frank Heckenbach, B. David Saunders, and Volkmar Welker. Computing simplicial homology based on efficient smith normal form algorithms. In *Algebra, Geometry, and Software Systems*, pages 177–206, 2003.
- [41] Jean-Guillaume Dumas, B. David Saunders, and Gilles Villard. On efficient sparse integer matrix smith normal form computations. *J. Symb. Comput.*, 32(1/2):71–99, 2001.
- [42] Ilya Dumer, Daniele Micciancio, and Madhu Sudan. Hardness of approximating the minimum distance of a linear code. *IEEE Transactions on Information Theory*, 49(1):22–37, 2003.
- [43] Wayne Eberly, Mark Giesbrecht, and Gilles Villard. Computing the determinant and smith form of an integer matrix. In *Proceedings of the 41st Annual Symposium on Foundations of Computer Science*, pages 675–685, 2000.
- [44] Herbert Edelsbrunner. Surface tiling with differential topology. In *Proceedings of the 21st ACM Symposium on Computational Geometry*, pages 9–11, 2005.
- [45] Herbert Edelsbrunner, Michael A. Facello, and Jie Liang. On the definition and the construction of pockets in macromolecules. *Discrete Applied Mathematics*, 88(1-3):83–102, 1998.
- [46] Herbert Edelsbrunner, John Harer, and Afra Zomorodian. Hierarchical morse - smale complexes for piecewise linear 2-manifolds. *Discrete & Computational Geometry*, 30(1):87–107, 2003.
- [47] Herbert Edelsbrunner, David G. Kirkpatrick, and Raimund Seidel. On the shape of a set of points in the plane. *IEEE Transactions on Information Theory*, 29(4):551–558, 1983.
- [48] Herbert Edelsbrunner, David Letscher, and Afra Zomorodian. Topological persistence and simplification. *Discrete & Computational Geometry*, 28(4):511–533, 2002.
- [49] Herbert Edelsbrunner, Dmitriy Morozov, and Valerio Pascucci. Persistence-sensitive simplification functions on 2-manifolds. In *Proceedings of the 22nd ACM Symposium on Computational Geometry*, pages 127–134, 2006.
- [50] Herbert Edelsbrunner and Ernst P. Mücke. Three-dimensional alpha shapes. *ACM Trans. Graph.*, 13(1):43–72, 1994.
- [51] Jeff Erickson and Sariel Har-Peled. Optimally cutting a surface into a disk. *Discrete & Computational Geometry*, 31(1):37–59, 2004.

- [52] Jeff Erickson and Kim Whittlesey. Greedy optimal homotopy and homology generators. In *Proceedings of the 16th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1038–1046, 2005.
- [53] Qing Fang, Jie Gao, and Leonidas Guibas. Locating and bypassing routing holes in sensor networks. In *Mobile Networks and Applications*, volume 11, pages 187–200, 2006.
- [54] Robin Forman. Combinatorial differential topology and geometry. In *New Perspectives in Algebraic Combinatorics*. MSRI, 1999.
- [55] François Franceschi and Erin M. Duffy. Structure-based drug design meets the ribosome. *Biochemical Pharmacology*, 71(7):1016–1025, March 2006.
- [56] Joel Friedman. Computing betti numbers via combinatorial laplacians. In *STOC*, pages 386–391, 1996.
- [57] Patrizio Frosini and Claudia Landi. Size theory as a topological tool for computer vision. *Pattern Recognition and Image Analysis*, 9(4):596–603, 1999.
- [58] Robert Ghrist. Barcodes: the persistent topology of data. *Bull. Amer. Math. Soc.*, 45(1):61–75, 2008.
- [59] Mark Giesbrecht. Fast computation of the smith form of a sparse integer matrix. *Computational Complexity*, 10(1):41–69, 2001.
- [60] Leonidas J. Guibas and Steve Oudot. Reconstruction using witness complexes. In *Proceedings of the 18th Annual ACM-SIAM Symposium on Discrete Algorithms*, 2007.
- [61] Christophe Guilbert and Thomas L. James. Docking to rna via root-mean-square-deviation-driven energy minimization with flexible ligands and flexible targets. *J. Chem. Inf. Model.*, 48(6):1257–1268, May 2008.
- [62] Igor Guskov and Zoë J. Wood. Topological noise removal. In *Proceedings of the Graphics Interface 2004 Conference*, pages 19–26, 2001.
- [63] Attila Gyulassy, Vijay Natarajan, Valerio Pascucci, Peer-Timo Bremer, and Bernd Hamann. A topological approach to simplification of three-dimensional scalar functions. *IEEE Trans. Vis. Comput. Graph.*, 12(4):474–484, 2006.
- [64] T. Kaczynski, M. Mrozek, and M. Slusarek. Homology computation by reduction of chain complexes. *Computers and Math. Appl.*, 35:59–70, 1998.
- [65] Tomasz Kaczynski, Konstantin Mischaikow, and Marian Mrozek. Computing homology. *Homology, Homotopy and Applications*, 5(2):233–256, 2003.

- [66] Ravindran Kannan and Achim Bachem. Polynomial algorithms for computing the smith and hermite normal forms of an integer matrix. *SIAM J. Comput.*, 8(4):499–507, 1979.
- [67] Peter M. Kasson, Afra Zomorodian, Sanghyun Park, Nina Singh, Leonidas J. Guibas, and Vijay S. Pande. Persistent voids: a new structural metric for membrane fusion. *Bioinformatics*, 2007.
- [68] Danil Kirisanov and Steven J. Gortler. A discrete global minimization algorithm for continuous variational problems. Technical Report TR-14-04, Harvard University, 2004.
- [69] Patrice Koehl, Michael Levitt, and Herbert Edelsbrunner. Proshape: Understanding the shape of proteins structures.
<http://biogeometry.cs.duke.edu/software/proshape/index.html>.
- [70] Martin Kutz. Computing shortest non-trivial cycles on orientable surfaces of bounded genus in almost linear time. In *Proceedings of the 22nd ACM Symposium on Computational Geometry*, pages 430–438, 2006.
- [71] David J. C. MacKay. *Information Theory, Inference & Learning Algorithms*. Cambridge University Press, 2002.
- [72] Yukio Matsumoto. *An Introduction to Morse Theory*. Translations of Mathematical Monographs. American Mathematical Society, 2002.
- [73] John P. May, David Saunders, and Zhendong Wan. Efficient matrix rank computation with application to the study of strongly regular graphs. In *International Symposium on Symbolic and Algebraic Computation*, 2007.
- [74] Boris Mederos, Nina Amenta, Luiz Velho, and Luiz Henrique de Figueiredo. Surface reconstruction for noisy point clouds. In *Symposium on Geometry Processing*, pages 53–62, 2005.
- [75] J. R. Munkres. *Elements of Algebraic Topology*. Addison-Wesley, Redwood City, California, 1984.
- [76] Xinlai Ni, Michael Garland, and John C. Hart. Fair morse functions for extracting the topological structure of a surface mesh. *ACM Trans. Graph.*, 23(3):613–622, 2004.
- [77] P. Niyogi, S. Smale, and S. Weinberger. Finding the homology of submanifolds with high confidence from random samples. *Discrete and Computational Geometry*, 39(1):419–441, 2008.
- [78] James G. Oxley. *Matroid Theory*. Oxford University Press, 1992.
- [79] Protein data bank. <http://www.rcsb.org/pdb/>.

- [80] Eric Rannaud. Persistence in computational topology. preprint, September 2004.
- [81] Vanessa Robins. Towards computing homology from finite approximations. *Topology Proceedings*, 24:503–532, 1999.
- [82] Rik Sarkar, Xiaotian Yin, Jie Gao, Feng Luo, and Xianfeng David Gu. Greedy routing with guaranteed delivery using ricci flows. In *Proc. of the 8th International Symposium on Information Processing in Sensor Networks (IPSN'09)*, pages 121–132, April 2009.
- [83] Arne Storjohann. Near optimal algorithms for computing smith normal forms of integer matrices. In *International Symposium on Symbolic and Algebraic Computation*, pages 267–274, 1996.
- [84] Arne Storjohann. *Algorithms for Matrix Canonical Forms*. PhD thesis, Department of Computer Science, Swiss Federal Institute of Technology – ETH, 2000.
- [85] Yusu Wang, Pankaj K. Agarwal, P. Brown, Herbert Edelsbrunner, and Johannes Rudolph. Coarse and reliable geometric alignment for protein docking. In *Pacific Symposium on Biocomputing*, pages 66–77, 2005.
- [86] Douglas H. Wiedemann. Solving sparse linear equations over finite fields. *IEEE Transactions on Information Theory*, 32(1):54–62, 1986.
- [87] Wikipedia. Book embedding.
http://en.wikipedia.org/wiki/Book__embedding.
- [88] Wikipedia. Suspension.
[http://en.wikipedia.org/wiki/Suspension_\(topology\)](http://en.wikipedia.org/wiki/Suspension_(topology)).
- [89] Zoë J. Wood, Hugues Hoppe, Mathieu Desbrun, and Peter Schröder. Removing excess topology from isosurfaces. *ACM Trans. Graph.*, 23(2):190–208, 2004.
- [90] Ada Yonath. Antibiotics targeting ribosomes: Resistance, selectivity, synergism, and cellular regulation. *Annual Review of Biochemistry*, 74:649–679, July 2005.
- [91] Afra Zomorodian. *Topology for Computing*. Cambridge Monographs on Applied and Computational Mathematics. Cambridge University Press, 2005.
- [92] Afra Zomorodian and Gunnar Carlsson. Computing persistent homology. *Discrete & Computational Geometry*, 33(2):249–274, 2005.
- [93] Afra Zomorodian and Gunnar Carlsson. Localized homology. In *Proceedings of the 2007 International Conference on Shape Modeling and Applications*, pages 189–198, 2007.