

# Discrete Morse Theoretic Algorithms for Computing Homology of Complexes and Maps

Shaun Harker\*    Konstantin Mischaikow†    Marian Mrozek‡    Vidit Nanda§

Communicated by Gunnar Carlsson.

## Abstract

We provide explicit and efficient reduction algorithms based on discrete Morse theory to simplify homology computation for a very general class of complexes. A set-valued map of top-dimensional cells between such complexes is a natural discrete approximation of an underlying (and possibly unknown) continuous function, especially when the evaluation of that function is subject to measurement errors. We introduce a new Morse theoretic pre-processing framework for deriving chain maps from such set-valued maps, and hence provide an effective scheme for computing the morphism induced on homology by the approximated continuous function.

**Keywords:** Computational homology, discrete Morse theory.

**Mathematics Subject Classification:** 55-04, 55N35, 57-04.

## 1 Introduction

The last two decades have witnessed a rapid growth in the range of applications of algebraic topology including sensor networks, image analysis, data analysis, material science and nonlinear dynamics [20, 13, 5, 12, 16, 15, 28, 10, 9, 29].

Recall that given a topological space  $X$ , algebraic topology provides mechanisms to compute associated *homology groups*  $H_k(X)$  for  $k \in \{0, 1, 2, \dots\}$  and furthermore, the same mechanisms construct abelian group homomorphisms  $f_*: H_*(X) \rightarrow H_*(Y)$  for a given continuous map  $f: X \rightarrow Y$ . Perhaps the best-known technique is that of simplicial homology. In this case, the topological spaces  $X$  and  $Y$  are represented in terms of simplicial complexes  $\mathcal{X}$  and  $\mathcal{Y}$  and the continuous map  $f$  is approximated via a simplicial map  $f_\#$ . This generates a map of chain complexes consisting of the graded free modules  $C_*(\mathcal{X})$  and  $C_*(\mathcal{Y})$  with respective bases  $\mathcal{X}$  and  $\mathcal{Y}$  along with corresponding boundary operators. Recall that the homology groups of  $\mathcal{X}$  and  $\mathcal{Y}$  are obtained via the Smith normal form diagonalization of the boundary operators [18, 20]. The Smith normal form is also used to explicitly determine the induced images of homology classes under  $f_*$  from  $f_\#$ .

The starting point for the aforementioned modern applications is different. One begins with finite complexes  $\mathcal{X}$  and  $\mathcal{Y}$  which are chosen according to the particular problem. Point cloud data is typically topologized via Rips or Čech complexes which leads to an abstract simplicial complex. Digital image data naturally leads to cubical complexes. Similarly, it is rare that a given numerical representation of a continuous function can be efficiently approximated via a simplicial map. Another important characteristic of modern applications is that the data sets – and consequently the associated complexes – are often large and high-dimensional.<sup>1</sup>

This disparity between the classical framework of algebraic topology and the modern applications has driven the development of efficient algorithms by which to compute the homology groups and the induced maps on homology. There are a variety of strategies that have been adopted. The most fundamental is to

---

\*Department of Mathematics, Rutgers University

†Department of Mathematics and BioMaPS Institute, Rutgers University

‡Division of Computational Mathematics, Faculty of Mathematics and Computer Science, Jagiellonian University

§Department of Mathematics, University of Pennsylvania

<sup>1</sup>See [5] for a description of a complex in  $\mathbb{R}^8$  with approximately  $4.5 \times 10^6$  vertices that arises from the study of natural images.

view Smith diagonalization as a purely algebraic problem and then to seek an optimal algorithm [11, 31]. The worst case analysis of such algorithms suggests a supercubical complexity with respect to the size of the complex, which is prohibitive for large datasets. An alternative strategy is to develop efficient algorithms for restricted problems, for example by restricting the dimension of the complex [8] or restricting the computation to that of Betti numbers [14]. The approach adopted in this paper is to pre-process the data by exploiting the geometry and the face relations associated with the input complex  $\mathcal{X}$ , thereby producing a smaller complex on which an algebraic Smith normal form computation can be performed.<sup>2</sup> Clearly, this is advantageous as long as the cost of preprocessing is low. This approach has proven to be effective in the past [22, 19, 20, 21, 26, 27, 28, 10].

The preprocessing proposed in this paper is based on the discrete Morse theory for finite complexes developed by Forman [14]. For our purposes, the central result of Forman's work is that given a complex  $(C_*(\mathcal{X}), \partial)$  it is possible to produce an alternative chain complex  $(C_*(\mathcal{M}), \Delta)$ , called the *Morse complex* where  $\mathcal{M} \subset \mathcal{X}$  such that  $H_*(\mathcal{M}) \cong H_*(\mathcal{X})$ . Furthermore, in general the size of the complex  $\mathcal{M}$  is significantly smaller than that of  $\mathcal{X}$ . With this in mind, the focus of this paper is threefold:

1. *Provide a framework for describing the complexes on which the pre-processing operations are performed.* This is done in Section 2 and makes use of complexes as developed by Tucker [33] and Lefschetz [24], along with more recent work on algebraic Morse theory (see [23] and references therein) and most closely related [26] where the focus is on incidence numbers that relate neighboring cells. In the contexts of data analysis and computational dynamics which motivate this work, this local information is often natural to the input whereas the associated boundary operator must be derived from this information. The explicit construction of this boundary operator, which can be costly for large complexes, is unnecessary in our approach.
2. *Present algorithms for efficiently computing Morse complexes.* This is the subject of Section 3. The chains of the Morse complex are obtained using an extension of the co-reduction algorithm introduced in [26]. The associated boundary operator agrees with that of [14], but the construction is done in terms of a more explicit algorithm.
3. *Use Morse complexes to efficiently compute the induced maps on homology.* This is presented in Section 4 and is similar in spirit to the approach taken by Allili and Kaczynski [1] and Mischaikow, Mrozek and Pilarczyk [25]. The fundamental difference is in employing the Morse complex to choose representations of homology generators used to define the induced map on homology.

The basic outline of our method for computing the map induced on homology by a relation  $\mathcal{F} : \mathcal{X} \rightrightarrows \mathcal{Y}$  between complexes is as follows. We first apply discrete Morse theory to produce Morse complexes  $\mathcal{M}_{\mathcal{X}}$  and  $\mathcal{M}_{\mathcal{Y}}$  of the domain  $\mathcal{X}$  and codomain  $\mathcal{Y}$  respectively. For each homology generator in  $\mathcal{M}_{\mathcal{X}}$ , we compute a representative cycle  $\mathbf{m}$  and consider an equivalent cycle  $\mathbf{x}$  in  $\mathcal{X}$ . We then lift  $\mathbf{x}$  to a corresponding cycle  $\mathbf{g}$  in the *graph*  $\Gamma$  of  $\mathcal{F}$  which is a subcomplex of the product  $\mathcal{X} \times \mathcal{Y}$ . More precisely, the  $\mathcal{X}$ -projection of  $\mathbf{g}$  equals  $\mathbf{x}$  and its construction involves solving the a preboundary problem in the fibers of  $\mathcal{F}$  lying in  $\Gamma$ . Again, discrete Morse theory can provide an efficient method for computing these preboundaries. Projecting  $\mathbf{g}$  to a cycle  $\mathbf{y}$  in the codomain  $\mathcal{Y}$ , we perform traditional algebraic operations to express the counterpart  $\mathbf{m}'$  of  $\mathbf{y}$  in the Morse complex  $\mathcal{M}_{\mathcal{Y}}$  as a linear combination of homology generators. Here is a schematic representation of this construction:

$$\mathcal{M}_{\mathcal{X}} \xleftarrow[\mathbf{m} \rightarrow \mathbf{x}]{\cong} \mathcal{X} \xrightarrow[\mathbf{x} \rightarrow \mathbf{g}]{\text{lift}} \Gamma \xrightarrow[\mathbf{g} \rightarrow \mathbf{y}]{\text{project}} \mathcal{Y} \xleftarrow[\mathbf{y} \rightarrow \mathbf{m}']{\cong} \mathcal{M}_{\mathcal{Y}}$$

As in the case of the method of homology models introduced in [28, Section 3.3], the central computational advantage of this technique is that all the expensive algebraic operations are performed over chains in the much smaller Morse complexes  $\mathcal{M}_{\mathcal{X}}$  and  $\mathcal{M}_{\mathcal{Y}}$ .

An outline of the paper is as follows. Section 2 provides fundamental definitions, notation and constructions that are used throughout the paper. The algorithms for computing homology are presented in Section 3 and the algorithms for computing the maps on homology are described in Section 4. We conclude in Section 5 with a discussion concerning the complexity in time and memory of our approach.

Before turning to the details, a final comment is in order. We have purposely chosen to present our algorithms on a fairly high level. We provide both proofs that the algorithms perform as expected and arguments concerning the complexity of our approach. At present there are at least three different implementations of these algorithms [7, 4, 30]. The efficiency of these algorithms is studied in [17] where experimentation shows that the different implementations have different run times. Which implementation is more favorable

---

<sup>2</sup>Since as indicated earlier, homology is computed via Smith normal form all these preprocessing algorithms can be interpreted algebraically. The difference is that the chain complex is constructed as late as possible and the algebraic operations are often motivated by geometric and combinatorial considerations.

depends on the type of problems being considered. Thus, we suspect that there is considerable room for improvement, and that variance on the implementation level is desirable.

## 2 Complexes and Maps

We present a framework for describing the complexes on which the preprocessing steps are performed. Section 2.1 discusses elementary properties of complexes and reviews important facts about the homology of these complexes from [26]. Section 2.2 provides the framework by which maps are represented. This framework is used in Section 2.3 to construct graph complexes which, as described in Section 2.4, play an essential role in the computation of the induced maps on homology.

### 2.1 Cell Complexes

Let  $\mathbb{N}$  be the set of natural numbers (including zero). Throughout this paper,  $\mathbf{R}$  denotes a principal ideal domain. The following definition of complex is based on [24] and is a straightforward reformulation of the definition of S-complex in [26, Section 2].

**Definition 2.1.** Consider a finite graded set  $\mathcal{X} = \bigsqcup_{q \in \mathbb{N}} \mathcal{X}_q$  along with a function  $\kappa: \mathcal{X} \times \mathcal{X} \rightarrow \mathbf{R}$ . An element  $\xi \in \mathcal{X}_q$  is called a *cell of dimension q*. We write  $\dim \xi$  to denote the dimension of  $\xi$ . The pair  $(\mathcal{X}, \kappa)$  is called a *complex* if the following properties are satisfied:

- (i) For each  $\xi$  and  $\xi' \in \mathcal{X}$ ,  $\kappa(\xi, \xi') \neq 0$  implies  $\dim \xi = \dim \xi' + 1$ , and
- (ii) For each  $\xi$  and  $\xi'' \in \mathcal{X}$ , the sum  $\sum_{\xi' \in \mathcal{X}} \kappa(\xi, \xi') \cdot \kappa(\xi', \xi'')$  equals  $0 \in \mathbf{R}$ .

Here  $\kappa$  is called the *incidence function* of the complex  $(\mathcal{X}, \kappa)$ .

We denote  $\kappa(\xi, \xi') \neq 0$  by  $\xi' \prec \xi$  and call  $\xi'$  a *facet* of  $\xi$ . The transitive closure  $\preceq$  of  $\prec$  generates the *face partial order* on  $\mathcal{X}$ , and we often write  $\xi'$  is a *face* of  $\xi$  to denote  $\xi' \preceq \xi$ .

**Definition 2.2.** Let  $(\mathcal{X}, \kappa)$  be a complex. The associated *chain complex* consists of the free modules  $C_q(\mathcal{X}) := \mathbf{R}(\mathcal{X}_q)$ , where the basis elements are the cells  $\xi \in \mathcal{X}_q$ , and the boundary operator is generated by the maps

$$\partial^{\mathcal{X}}(\xi) := \sum_{\xi' \in \mathcal{X}} \kappa(\xi, \xi') \xi'. \quad (1)$$

It follows from Definition 2.1(i) that  $\partial_q^{\mathcal{X}}: C_q(\mathcal{X}) \rightarrow C_{q-1}(\mathcal{X})$  and from Definition 2.1(ii) that  $\partial^{\mathcal{X}} \circ \partial^{\mathcal{X}} = 0$ . The  $q$ -cycles and  $q$ -boundaries are defined as  $Z_q(\mathcal{X}) = \ker \partial_q^{\mathcal{X}}$  and  $B_q(\mathcal{X}) = \text{img } \partial_{q+1}^{\mathcal{X}}$  respectively. The associated *homology groups* of this chain complex are defined as the quotient  $H_q(\mathcal{X}; \mathbf{R}) = Z_q/B_q$ . Since the coefficient ring  $\mathbf{R}$  remains fixed throughout, we will omit it from the notation and denote homology groups by  $H_*(\mathcal{X})$ . Given complexes  $\mathcal{X}$  and  $\mathcal{Y}$ , a *chain map* from  $C_*(\mathcal{X})$  to  $C_*(\mathcal{Y})$  is an  $\mathbf{R}$ -module morphism  $\phi: C_*(\mathcal{X}) \rightarrow C_*(\mathcal{Y})$  satisfying  $\phi \circ \partial^{\mathcal{X}} \equiv \partial^{\mathcal{Y}} \circ \phi$ . Chain maps induce well-defined morphisms of homology groups.

Given  $\mathcal{A} \subset \mathcal{X}$ , let  $\iota_{\mathcal{A}}$  and  $\pi_{\mathcal{A}}$  denote the inclusion and projection maps between chains  $C_*(\mathcal{X})$  and  $C_*(\mathcal{A})$ . Define  $\partial^{\mathcal{A}}$  on  $C_*(\mathcal{A})$  by  $\partial^{\mathcal{A}} = \pi_{\mathcal{A}} \circ \partial^{\mathcal{X}} \circ \iota_{\mathcal{A}}$ . If  $\partial^{\mathcal{A}} \circ \partial^{\mathcal{A}} \equiv 0$  then  $\mathcal{A}$  is called a *subcomplex* of  $(\mathcal{X}, \kappa)$ . Note that a subcomplex, as we define it, need not be a subcomplex in the sense of chain complexes. The following notation provides convenient local information that can be used to study or modify the complex  $(\mathcal{X}, \kappa)$ . Given any subset  $\mathcal{A} \subset \mathcal{X}$ , define

$$\text{bd}_{\mathcal{X}} \mathcal{A} := \{\xi \in \mathcal{X} \mid \xi \prec \alpha \text{ for some } \alpha \in \mathcal{A}\}.$$

The following result provides additional conditions, depending only on  $\kappa$ , which guarantee that  $\mathcal{A}$  is a subcomplex of  $\mathcal{X}$ . Define  $\mathcal{A} \subset \mathcal{X}$  to be *combinatorially closed* if

$$\bigcup_{\alpha \in \mathcal{A}} \text{bd}_{\mathcal{X}} \alpha \subset \mathcal{A}.$$

Henceforth, we will say that a subcomplex is *closed* if it is combinatorially closed and in this case it is easy to see that  $\partial^{\mathcal{A}}$  is just the restriction of  $\partial^{\mathcal{X}}$  to the chains  $C_*(\mathcal{A})$ . We say that  $\mathcal{A}$  is *open* in  $\mathcal{X}$  if  $\mathcal{X} \setminus \mathcal{A}$  is closed in  $\mathcal{X}$ . Similarly, the *closure* of a subset  $\mathcal{X}' \subset \mathcal{X}$  is defined to be the smallest closed subcomplex of  $\mathcal{X}$  which contains  $\mathcal{X}'$  and is denoted by  $\overline{\mathcal{X}'}$ .

**Proposition 2.3.** [26, Theorem 3.2] *Let  $(\mathcal{X}, \kappa)$  be a complex. If  $\mathcal{A} \subset \mathcal{X}$  is closed, then both  $(\mathcal{A}, \kappa)$  and  $(\mathcal{X} \setminus \mathcal{A}, \kappa)$  are subcomplexes of  $\mathcal{X}$ .*

Observe that if  $\mathcal{A} \subset \mathcal{X}$  is closed then we can define the *relative chain complex*

$$C_*(\mathcal{X}, \mathcal{A}) := \frac{C_*(\mathcal{X})}{C_*(\mathcal{A})}$$

with the boundary operator  $\partial^{(\mathcal{X}, \mathcal{A})}$  induced by the quotient map. The associated *relative homology groups* are denoted by  $H_*(\mathcal{X}, \mathcal{A})$ . From a computational perspective, an important consequence of Proposition 2.3 is the following.

**Proposition 2.4.** [26, Theorem 3.5] *Let  $(\mathcal{X}, \kappa)$  be a complex and let  $\mathcal{A} \subset \mathcal{X}$  be closed, then*

$$H_*(\mathcal{X} \setminus \mathcal{A}) \cong H_*(\mathcal{X}, \mathcal{A}).$$

The following proposition establishes that chain maps descend to closed and open subcomplexes as expected.

**Proposition 2.5.** *Let  $\mu : C_*(\mathcal{X}) \rightarrow C_*(\mathcal{Y})$  be a chain map between complexes and assume that  $\mathcal{A} \subset \mathcal{X}$  is a closed subcomplex. Then,*

1. *the restriction  $\mu|_{C_*(\mathcal{A})}$  is a chain map from  $C_*(\mathcal{A})$  to  $C_*(\mathcal{Y})$ , and*
2. *if  $\mu(C_*(\mathcal{A})) \subset C_*(\mathcal{B})$  for some closed subcomplex  $\mathcal{B} \subset \mathcal{Y}$ , then  $\mu|_{C_*(\mathcal{X} \setminus \mathcal{A})}$  descends to a chain map from  $C_*(\mathcal{X} \setminus \mathcal{A})$  to  $C_*(\mathcal{Y} \setminus \mathcal{B})$ .*

*Proof.* If  $\mathcal{A} \subset \mathcal{X}$  is a closed subcomplex, then for any chain  $\mathbf{a} \in C_*(\mathcal{A})$  we have

$$\mu \circ \partial^{\mathcal{A}}(\mathbf{a}) = \mu \circ \partial^{\mathcal{X}}(\mathbf{a}) = \partial^{\mathcal{Y}} \circ \mu(\mathbf{a})$$

where the first equality follows from the fact that  $\mathcal{A}$  is closed and the second from the fact that  $\mu$  is a chain map from  $\mathcal{X}$  to  $\mathcal{Y}$ . On the other hand, if  $\mu(C_*(\mathcal{A})) \subset C_*(\mathcal{B})$  then  $\mu$  descends to a well-defined chain map  $\mu' : C_*(\mathcal{X}, \mathcal{A}) \rightarrow C_*(\mathcal{Y}, \mathcal{B})$  on relative homology from which the desired result follows by choosing the canonical representative chains in  $C_*(\mathcal{X} \setminus \mathcal{A})$  and  $C_*(\mathcal{Y} \setminus \mathcal{B})$  for the relative chains in the domain and range respectively.  $\square$

**Definition 2.6.** Let  $(\mathcal{X}, \kappa)$  and  $(\mathcal{Y}, \tau)$  be complexes. Their *product complex* denoted by  $(\mathcal{X} \times \mathcal{Y}, \kappa^\times)$  consists of elements  $\xi \times \eta$  where

$$\dim(\xi \times \eta) = \dim \xi + \dim \eta$$

and the *product incidence function* is given by

$$\kappa^\times(\xi \times \eta, \xi' \times \eta') := \begin{cases} \kappa(\xi, \xi') & \text{if } \eta = \eta', \\ (-1)^{\dim \xi} \tau(\eta, \eta') & \text{if } \xi = \xi', \\ 0 & \text{otherwise.} \end{cases} \quad (2)$$

As is demonstrated in [24, IV.1.1], a product complex is a complex. It is convenient to identify the associated chain complex  $C_*(\mathcal{X} \times \mathcal{Y})$  with the tensor product  $C_*(\mathcal{X}) \otimes_{\mathbf{R}} C_*(\mathcal{Y})$ . The explicit bases for this identification are given by  $\xi \times \eta \sim \xi \otimes \eta$  for  $\xi \in \mathcal{X}$  and  $\eta \in \mathcal{Y}$ . The *product boundary operator* takes the form

$$\partial^\times(\xi \otimes \eta) = \partial^{\mathcal{X}}\xi \otimes \eta + (-1)^{\dim \xi} \xi \otimes \partial^{\mathcal{Y}}\eta.$$

The canonical projection maps  $\pi_{\mathcal{X}} : C_{\mathbf{k}}(\mathcal{X} \times \mathcal{Y}) \rightarrow C_{\mathbf{k}}(\mathcal{X})$  and  $\pi_{\mathcal{Y}} : C_{\mathbf{k}}(\mathcal{X} \times \mathcal{Y}) \rightarrow C_{\mathbf{k}}(\mathcal{Y})$  are defined on the basis elements as follows and extended via  $\mathbf{R}$ -bilinearity:

$$\pi_{\mathcal{X}}(\xi \otimes \eta) := \begin{cases} \xi & \text{if } \dim \eta = 0 \\ 0 & \text{otherwise} \end{cases} \quad \text{and} \quad \pi_{\mathcal{Y}}(\xi \otimes \eta) := \begin{cases} \eta & \text{if } \dim \xi = 0 \\ 0 & \text{otherwise} \end{cases}$$

As is to be expected, we use these projection maps to relate subcomplexes of  $(\mathcal{X} \times \mathcal{Y}, \kappa^\times)$  to subcomplexes of  $(\mathcal{X}, \kappa)$  and  $(\mathcal{Y}, \tau)$ . However, to do this on the level of homology requires that the projection maps induce chain maps. The following example indicates that this need not be true for general complexes.

**Example 2.7.** Let  $\mathcal{X}$  be a complex containing a single cell  $\xi$  with  $\dim \xi = 1$ , so that the incidence function  $\kappa$  is trivial. Consider  $\mathcal{Y} = \{\eta_1, \eta_0\}$  where  $\dim \eta_i = i$  and  $\tau(\eta_1, \eta_0) = 1$ . Then,  $(\mathcal{X}, \kappa)$  and  $(\mathcal{Y}, \tau)$  are complexes. Furthermore,  $\mathcal{X} \times \mathcal{Y} = \{\xi \times \eta_1, \xi \times \eta_0\}$  and  $\kappa^\times(\xi \times \eta_1, \xi \times \eta_0) = -\tau(\eta_1, \eta_0)$ . Thus  $\partial^\times(\xi \otimes \eta_1) = -\xi \otimes \eta_0$ . Observe that

$$\pi_{\mathcal{X}} \circ \partial^\times(\xi \otimes \eta_1) = -\xi \neq 0 = \partial^\times \circ \pi_{\mathcal{X}}(\xi \otimes \eta_1).$$

Hence,  $\pi_{\mathcal{X}}: C_*(\mathcal{X} \times \mathcal{Y}) \rightarrow C_*(\mathcal{X})$  is not a chain map.

**Definition 2.8.** Let  $\epsilon: C_0(\mathcal{X}) \rightarrow \mathbf{R}$  be the standard augmentation map defined by  $\epsilon(\sum_i r_i \xi_i) = \sum_i r_i$ ,  $r_i \in \mathbf{R}$ . The complex  $\mathcal{X}$  is *augmentable* if  $\epsilon \circ \partial_1^\mathcal{X} = 0$ .

A direct computation leads to the following proposition which is our primary use of the augmentability condition.

**Proposition 2.9.** *If  $\mathcal{X}$  and  $\mathcal{Y}$  are augmentable, then the canonical projection maps  $\pi_{\mathcal{X}}$  and  $\pi_{\mathcal{Y}}$  are chain maps.*

## 2.2 Combinatorial Maps

In this section we describe the representations of maps which we employ. It should be noted that [33, 24] provide a framework for set transformations that – from the perspective of the applications that we envision – appears to be more general than necessary.

**Definition 2.10.** Let  $(\mathcal{X}, \kappa)$  and  $(\mathcal{Y}, \tau)$  be complexes. A *combinatorial map* from  $\mathcal{X}$  to  $\mathcal{Y}$  is a multi-valued map  $\mathcal{F}: \mathcal{X} \rightrightarrows \mathcal{Y}$  so that the set  $\{\xi \times \eta \mid \xi \in \mathcal{X}, \eta \in \mathcal{F}(\xi)\}$  is a closed subcomplex of the product  $\mathcal{X} \times \mathcal{Y}$ .

Our preferred method of generating combinatorial maps on complexes is to focus on transformations that are specified by their action on top-cells as defined below.

**Definition 2.11.** Given a complex  $(\mathcal{X}, \kappa)$ ,  $\xi \in \mathcal{X}$  is a *top-cell* if it is maximal with respect to  $\preceq$ . The set of top-cells is denoted by  $\mathcal{X}^+ \subset \mathcal{X}$ .

**Definition 2.12.** Let  $(\mathcal{X}, \kappa)$  and  $(\mathcal{Y}, \tau)$  be complexes. A *toplex map*  $\mathcal{T}: \mathcal{X}^+ \rightrightarrows \mathcal{Y}^+$  is a multi-valued map which takes top-cells of  $\mathcal{X}$  to nonempty sets of top-cells of  $\mathcal{Y}$ .

Our motivation for choosing to work with toplex maps comes from two sources:

1. they are compact representations for a map since they only involve top-cells, and
2. they arise naturally in problems involving numerical computations or the analysis of experimental data that involve error bounds.

The following construction generates a combinatorial map from a toplex map.

**Definition 2.13.** Given a toplex map  $\mathcal{T}: \mathcal{X}^+ \rightrightarrows \mathcal{Y}^+$ , the *associated combinatorial map*  $\mathcal{F}: \mathcal{X} \rightrightarrows \mathcal{Y}$  is defined inductively on  $\preceq$  as follows:

$$\mathcal{F}(\xi) := \begin{cases} \overline{\mathcal{T}(\xi)} & \text{if } \xi \in \mathcal{X}^+ \\ \bigcup_{\xi' \prec \xi} \mathcal{F}(\xi') & \text{otherwise.} \end{cases}$$

Many applications make use of maps on pairs. Again, we wish to encode the information via the top-cells. Given a closed subcomplex  $\mathcal{X}_0 \subset \mathcal{X}$ , we say that  $\mathcal{X}_0$  is *full* if  $\mathcal{X}_0^+ \subset \mathcal{X}^+$ , and we say that  $(\mathcal{X}, \mathcal{X}_0)$  is a *full pair* of complexes whenever  $\mathcal{X}_0$  is a full subcomplex of  $\mathcal{X}$ .

**Definition 2.14.** Let  $(\mathcal{X}, \kappa)$  and  $(\mathcal{Y}, \tau)$  be complexes with full subcomplexes  $\mathcal{X}_0 \subset \mathcal{X}$  and  $\mathcal{Y}_0 \subset \mathcal{Y}$ . A *toplex pair map*

$$\mathcal{T}: (\mathcal{X}^+, \mathcal{X}_0^+) \rightrightarrows (\mathcal{Y}^+, \mathcal{Y}_0^+)$$

is a toplex map  $\mathcal{T}: \mathcal{X}^+ \rightrightarrows \mathcal{Y}^+$  such that  $\mathcal{T}(\mathcal{X}_0^+) \subset \mathcal{Y}_0^+$ .

We may construct combinatorial pair maps from toplex pair maps.

**Definition 2.15.** Let  $\mathcal{T}: (\mathcal{X}^+, \mathcal{X}_0^+) \rightrightarrows (\mathcal{Y}^+, \mathcal{Y}_0^+)$  be a relative toplex map of full pairs. The *associated combinatorial pair map* is a pair  $(\mathcal{F}, \mathcal{F}_0)$  of combinatorial maps  $\mathcal{F}: \mathcal{X} \rightrightarrows \mathcal{Y}$ , and  $\mathcal{F}_0: \mathcal{X}_0 \rightrightarrows \mathcal{Y}_0$  which are derived from  $\mathcal{T}$  and  $\mathcal{T}|_{\mathcal{X}_0^+}$  respectively via Definition 2.13.

## 2.3 Graph Complexes

As is indicated in the Introduction, our approach to computing the induced map on homology is based on the techniques of [25]. The essential idea is as follows: given topological spaces  $X, Y$  and a continuous map  $f: X \rightarrow Y$ , let  $G \subset X \times Y$  denote the graph of  $f$ . Let  $\pi_X: G \rightarrow X$  and  $\pi_Y: G \rightarrow Y$  denote the canonical projection maps. Observe that  $f = \pi_Y \circ \pi_X^{-1}$  and hence  $f_* = \pi_{Y*} \circ \pi_{X*}^{-1}$ .

The data for our problem is fundamentally different. We work with complexes and combinatorial maps rather than topological spaces and continuous maps.<sup>3</sup> Throughout this section, we consider augmentable complexes  $(X, \kappa)$  and  $(Y, \tau)$  along with a combinatorial map  $\mathcal{F}: X \rightrightarrows Y$  derived from a topex map  $\mathcal{T}: X^+ \rightrightarrows Y^+$ . Nevertheless, we adopt a similar strategy: consider the graph generated by  $\mathcal{F}$  as a subcomplex of  $X \times Y$  and use the composition of the projection maps to determine  $\mathcal{F}_*$ .

**Definition 2.16.** The *graph complex* of  $\mathcal{F}: X \rightrightarrows Y$  consists of the cells

$$\Gamma := \{\xi \times \eta \mid \xi \in X, \eta \in \mathcal{F}(\xi)\} \subset X \times Y$$

and the *graph incidence function* is given by the restriction of the product incidence function  $\kappa^\times: (X \times Y) \times (X \times Y) \rightarrow \mathbf{R}$ .

Given  $\mathcal{A} \subset X$ , we consider the set

$$\Gamma^{\mathcal{A}} := \{\xi \times \eta \mid \xi \in \mathcal{A}, \eta \in \mathcal{F}(\xi)\} \subset \Gamma$$

with incidence function defined by the restriction of  $\kappa^\times$ . In particular, when  $\mathcal{A}$  is a single cell  $\xi \in X$ , we have

$$\Gamma^\xi := \Gamma^{\{\xi\}} = \{\xi \times \eta \mid \eta \in \mathcal{F}(\xi)\} \subset \Gamma.$$

Observe that in this case the incidence function can be viewed as either the restriction of  $\kappa^\times$  or the restriction of  $\tau$  since the restrictions are identical.

By Definition 2.10, we know that the graph  $\Gamma$  of  $\mathcal{F}$  is closed in  $X \times Y$ . Here is a stronger statement:

**Proposition 2.17.** *For each closed subcomplex  $\mathcal{A} \subset X$ , the associated fiber complex  $\Gamma^{\mathcal{A}}$  is a closed subcomplex of the product  $(X \times Y, \kappa^\times)$ .*

*Proof.* Let  $\xi \times \eta \in \text{bd}_{X \times Y} \Gamma^{\mathcal{A}}$ . Then, there exists  $\xi' \times \eta' \in \Gamma^{\mathcal{A}}$  such that either  $\xi = \xi'$  and  $\eta \prec \eta'$  in  $F(\xi) \subset Y$ , or  $\eta = \eta'$  and  $\xi \prec \xi'$  in  $X$ . In the first case, we have  $\eta \in \text{bd}_Y \mathcal{F}(\xi') \subset \mathcal{F}(\xi') = \mathcal{F}(\xi)$ , hence  $\xi \times \eta \in \Gamma^{\mathcal{A}}$  because  $\xi = \xi' \in \mathcal{A}$ . Alternately, we have  $\xi \in \text{bd}_X \xi' \subset \text{bd}_X \mathcal{A} \subset \mathcal{A}$  and  $\eta = \eta' \in \mathcal{F}(\xi') \subset \mathcal{F}(\xi)$ , hence  $\xi \times \eta \in \Gamma^{\mathcal{A}}$ .  $\square$

As a consequence of this proposition, the restriction of  $\partial^\times$  to  $\Gamma$  produces a chain complex  $(C_*(\Gamma), \partial^\times|_\Gamma)$ . Restricting further to a closed subcomplex  $\mathcal{A} \subset X$  gives rise to a chain complex  $(C_*(\Gamma^{\mathcal{A}}), \partial^\times|_{\Gamma^{\mathcal{A}}})$ .

The following proposition follows directly from Propositions 2.9 and 2.5 once we recall that the graph of a combinatorial map is closed by Definition 2.10.

**Proposition 2.18.** *Let  $\mathcal{F}: X \rightrightarrows Y$  be a combinatorial map of augmentable complexes and let  $(\Gamma, \kappa^\times)$  be its graph complex. If  $X$  and  $Y$  are augmentable, then the restriction of the canonical projection maps  $\pi_X$  and  $\pi_Y$  to  $C_*(\Gamma)$  are chain maps to  $C_*(X)$  and  $C_*(Y)$  respectively.*

To work with maps on pairs we need the following.

**Definition 2.19.** Let  $(\mathcal{F}, \mathcal{F}_0): (X, X_0) \rightrightarrows (Y, Y_0)$  be a combinatorial pair map. The associated *graph* is defined to be the subcomplex  $(\Gamma \setminus \Gamma_0, \kappa^\times)$  of  $X \times Y$  consisting of cells in the graph complex  $(\Gamma, \kappa^\times)$  of  $\mathcal{F}$  which do not lie in the graph complex  $(\Gamma_0, \kappa^\times)$  of  $\mathcal{F}_0$ .

Given a closed subcomplex  $\mathcal{A} \subset X$ , the associated graph is  $(\Gamma^{\mathcal{A}} \setminus \Gamma_0^{X_0 \cap \mathcal{A}})$  and in particular for each  $\xi \in X$  the associated *fiber* is  $(\Gamma^\xi \setminus \Gamma_0^\xi)$  where  $\Gamma_0^\xi = \emptyset$  if  $\xi \notin X_0$ . Direct applications of Propositions 2.18 and 2.5(2) give rise to the following result.

**Proposition 2.20.** *Let  $(\mathcal{F}, \mathcal{F}_0): (X, X_0) \rightrightarrows (Y, Y_0)$  be a combinatorial pair map with associated graph complex  $(\Gamma \setminus \Gamma_0)$ . If  $X$  and  $Y$  are augmentable, then the restrictions of the canonical projection maps  $\pi_X$  and  $\pi_Y$  to  $C_*(\Gamma \setminus \Gamma_0)$  are chain maps to  $C_*(X \setminus X_0)$  and  $C_*(Y \setminus Y_0)$  respectively.*

<sup>3</sup>As is shown in [25], there are relationships between these two settings.

**Definition 2.21.** A combinatorial map  $\mathcal{F} : \mathcal{X} \rightarrow \mathcal{Y}$  is *acyclic* if for each  $\xi \in \mathcal{X}$  the subcomplex  $\mathcal{F}(\xi) \subset \mathcal{Y}$  is acyclic; that is,

$$H_q(\mathcal{F}(\xi)) = \begin{cases} \mathbf{R} & \text{if } q = 0 \\ 0 & \text{otherwise.} \end{cases}$$

A relative combinatorial map  $(\mathcal{F}, \mathcal{F}_0)$  is *acyclic* if both  $\mathcal{F}$  and  $\mathcal{F}_0$  are acyclic.

Recall that we have already assumed the augmentability of  $(\mathcal{X}, \kappa)$  and  $(\mathcal{Y}, \tau)$  throughout. Henceforth we will also require that the combinatorial maps  $\mathcal{F} : \mathcal{X} \rightrightarrows \mathcal{Y}$  be acyclic. In all situations dealing with relative homology, we similarly require  $(\mathcal{F}, \mathcal{F}_0) : (\mathcal{X}, \mathcal{X}_0) \rightrightarrows (\mathcal{Y}, \mathcal{Y}_0)$  to be acyclic. For each cell  $\xi \in \mathcal{X}$ , define  $\{\xi\} \subset \mathcal{X}$  to be the complex consisting of the single cell  $\xi$ . Note that its combinatorial closure  $\{\bar{\xi}\}$  is the closed subcomplex of  $\mathcal{X}$  consisting of all  $\xi' \preceq \xi$ .

**Proposition 2.22.** *If  $\xi \in \mathcal{X}$ , then*

$$H_q(\Gamma^\xi) \cong \begin{cases} \mathbf{R} & \text{if } q = \dim \xi \\ 0 & \text{otherwise} \end{cases}$$

and  $\pi_{\mathcal{X}}$  restricted to  $C_*(\Gamma^\xi)$  induces an isomorphism from  $H_*(\Gamma^{\{\xi\}})$  to  $H_*(\{\bar{\xi}\})$ .

*Proof.* Observe that

$$H_q(\{\bar{\xi}\}) \cong \begin{cases} \mathbf{R} & \text{if } q = \dim \xi \\ 0 & \text{otherwise.} \end{cases}$$

Since  $\mathcal{F}(\xi)$  is acyclic and  $\Gamma^\xi = \{\xi\} \times \mathcal{F}(\xi)$ , the result follows from the Künneth theorem.  $\square$

**Proposition 2.23.** *Let  $(\mathcal{F}, \mathcal{F}_0) : (\mathcal{X}, \mathcal{X}_0) \rightrightarrows (\mathcal{Y}, \mathcal{Y}_0)$  be an acyclic combinatorial pair map with the associated graph  $(\Gamma \setminus \Gamma_0)$ , where  $\mathcal{X}$  and  $\mathcal{Y}$  are augmentable complexes. If  $\xi \in \mathcal{X} \setminus \mathcal{X}_0$ , then*

$$H_q(\Gamma^\xi \setminus \Gamma_0^\xi) \cong \begin{cases} \mathbf{R} & \text{if } q = \dim \xi \\ 0 & \text{otherwise.} \end{cases}$$

And for each  $\xi \in \mathcal{X}_0$

$$H_*(\Gamma^\xi \setminus \Gamma_0^\xi) = 0.$$

*Proof.* Observe that if  $\xi \in \mathcal{X} \setminus \mathcal{X}_0$ , then  $\Gamma_0^\xi = \emptyset$  and the result follows from Proposition 2.22. If  $\xi \in \mathcal{X}_0$ , then  $\Gamma_0^\xi \subset \Gamma^\xi$  is a closed subcomplex and both satisfy Proposition 2.22. Thus by the exact sequence of a pair, we have  $H_*(\Gamma^\xi, \Gamma_0^\xi) = 0$ . Using Proposition 2.4 concludes the argument.  $\square$

The goal for the remainder of this subsection is the proof of the following result which is a special case of the Vietoris-Begle Theorem [32, Theorem 6.9.15].

**Theorem 2.24.** *Let  $(\mathcal{F}, \mathcal{F}_0) : (\mathcal{X}, \mathcal{X}_0) \rightrightarrows (\mathcal{Y}, \mathcal{Y}_0)$  be an acyclic combinatorial pair map with graph  $(\Gamma \setminus \Gamma_0)$  where  $\mathcal{X}$  and  $\mathcal{Y}$  are augmentable complexes. Then, the restriction of the canonical projection map  $\pi_{\mathcal{X}}$  to  $C_*(\Gamma \setminus \Gamma_0)$  induces an isomorphism from  $H_*(\Gamma \setminus \Gamma_0)$  to  $H_*(\mathcal{X} \setminus \mathcal{X}_0)$ .*

We begin with some simple lemmas.

**Lemma 2.25.** *Let  $\mathcal{U}, \mathcal{V} \subset \mathcal{X}$  be closed subcomplexes. Assume that the restrictions of the canonical projections  $\pi_{\mathcal{X}}$  to  $C_*(\Gamma^{\mathcal{Z}})$  induce isomorphisms  $H_*(\Gamma^{\mathcal{Z}}) \simeq H_*(\mathcal{Z})$  for  $\mathcal{Z} \in \{\mathcal{U}, \mathcal{V}, \mathcal{U} \cap \mathcal{V}\}$ . Then,*

$$\pi_{\mathcal{X}}|_{C_*(\Gamma^{\mathcal{U} \cup \mathcal{V}})} : C_*(\Gamma^{\mathcal{U} \cup \mathcal{V}}) \rightarrow C_*(\mathcal{U} \cup \mathcal{V})$$

also induces an isomorphism on homology.

*Proof.* The assumption that  $\mathcal{U}$  and  $\mathcal{V}$  are closed implies that  $\mathcal{U} \cap \mathcal{V}$  and  $\mathcal{U} \cup \mathcal{V}$  are closed in the complex  $(\mathcal{X}, \kappa)$ . The proof now follows from the Five Lemma and the Mayer-Vietoris sequence.  $\square$

The following result is essentially a restatement of Proposition 2.22 once one observes that if  $\dim \xi = 0$  then  $\{\bar{\xi}\} = \{\xi\}$ .

**Lemma 2.26.** *Let  $\xi \in \mathcal{X}$  be a cell of dimension 0. Then the restriction of  $\pi_{\mathcal{X}}$  to  $C_*(\Gamma^{\overline{\{\xi\}}})$  induces an isomorphism*

$$H_*(\Gamma^{\overline{\{\xi\}}}) \simeq H_*(\{\overline{\xi}\})$$

We will prove the following proposition and theorem simultaneously.

**Proposition 2.27.** *For every  $\xi \in \mathcal{X}$ , the restriction of  $\pi_{\mathcal{X}}$  to  $C_*(\Gamma^{\overline{\{\xi\}}})$  induces an isomorphism*

$$H_*(\Gamma^{\overline{\{\xi\}}}) \simeq H_*(\{\overline{\xi}\})$$

**Theorem 2.28.** *Let  $\mathcal{F}: \mathcal{X} \rightrightarrows \mathcal{Y}$  be an acyclic combinatorial map with graph complex  $(\Gamma, \kappa^\times)$ , where  $(\mathcal{X}, \kappa)$  and  $(\mathcal{Y}, \tau)$  are augmentable complexes. If  $\mathcal{A} \subset \mathcal{X}$  is closed, then*

$$\pi_{\mathcal{X}}|_{C_*(\Gamma^{\mathcal{A}})}: C_*(\Gamma^{\mathcal{A}}) \rightarrow C_*(\mathcal{A})$$

*induces an isomorphism on homology.*

*Proof of Proposition 2.27 and Theorem 2.28.* Given  $\mathcal{A} \subset \mathcal{X}$  set  $\dim \mathcal{A} := \max\{\dim \xi \mid \xi \in \mathcal{A}\}$ . Observe that it is enough to prove that for every  $n$  the following statements are satisfied:

A(n): For every  $\xi \in \mathcal{X}$  such that  $\dim \xi \leq n$ , the restriction of  $\pi_{\mathcal{X}}$  to  $C_*(\Gamma^{\overline{\{\xi\}}})$  induces an isomorphism

$$H_*(\Gamma^{\overline{\{\xi\}}}) \simeq H_*(\{\overline{\xi}\}).$$

B(n): If  $\mathcal{A} \subset \mathcal{X}$  is closed and  $\dim \mathcal{A} \leq n$ , then

$$\pi_{\mathcal{X}}|_{C_*(\Gamma^{\mathcal{A}})}: C_*(\Gamma^{\mathcal{A}}) \rightarrow C_*(\mathcal{A})$$

induces an isomorphism on homology.

In the proof we will use the following induction scheme:

- (i) A(0) holds true,
- (ii) A(n)  $\implies$  B(n),
- (iii) B(n)  $\implies$  A(n+1).

Property (i) follows immediately from Lemma 2.26.

To prove property (ii) assume that A(n) holds. Let  $\mathcal{A}^+$  be the set of  $\preceq$ -maximal cells in  $\mathcal{A}$ . By A(n),  $\pi_{\mathcal{X}}: \Gamma^{\overline{\{\xi\}}} \rightarrow \{\overline{\xi}\}$  induces an isomorphism on homology for each  $\xi \in \mathcal{A}$ . By Lemma 2.25, if  $\zeta$  is another cell in  $\mathcal{A}^+$  then

$$\pi_{\mathcal{X}}: \Gamma^{\overline{\{\xi\}}} \cup \Gamma^{\overline{\{\zeta\}}} \rightarrow \{\overline{\xi}\} \cup \{\overline{\zeta}\}$$

induces an isomorphism on homology. Since  $\mathcal{A}$  is closed,  $\mathcal{A} = \bigcup_{\xi \in \mathcal{A}} \overline{\{\xi\}}$ , which implies that repeating this argument inductively gives the result.

To prove property (iii) assume that B(n) holds. Given  $\xi \in \mathcal{X}$  with  $\dim \xi \leq n+1$ , define

$$\mathcal{S} := \overline{\{\xi\}} \setminus \{\xi\} = \{\xi' \in \mathcal{X} \mid \xi \neq \xi' \preceq \xi\}$$

and observe that  $\mathcal{S}$  and  $\Gamma^{\mathcal{S}}$  are closed in  $\mathcal{X}$  and  $\Gamma$  respectively. By Proposition 2.3 we have the following isomorphisms

$$H_k(\Gamma^{\overline{\{\xi\}}}, \Gamma^{\mathcal{S}}) \cong H_k(\Gamma^{\{\xi\}})$$

and

$$H_k(\overline{\{\xi\}}, \mathcal{S}) \cong H_k(\{\overline{\xi}\}).$$

Combining this with the long exact sequence of a pair leads to the following commutative diagram

$$\begin{array}{ccccccc} \cdots & \longrightarrow & H_k(\Gamma^{\mathcal{S}}) & \longrightarrow & H_k(\Gamma^{\overline{\{\xi\}}}) & \longrightarrow & H_k(\Gamma^{\{\xi\}}) \longrightarrow \cdots \\ & & \downarrow \pi_{\mathcal{X}*} & & \downarrow \pi_{\mathcal{X}*} & & \downarrow \pi_{\mathcal{X}*} \\ \cdots & \longrightarrow & H_k(\mathcal{S}) & \longrightarrow & H_k(\overline{\{\xi\}}) & \longrightarrow & H_k(\{\overline{\xi}\}) \longrightarrow \cdots \end{array}$$

By Proposition 2.22,  $\pi_{\mathcal{X}*}: H_*(\Gamma^{\{\xi\}}) \rightarrow H_*(\{\overline{\xi}\})$  is an isomorphism. Furthermore, since  $\dim \mathcal{S} \leq n$ , the inductive hypothesis B(n) implies that  $\pi_{\mathcal{X}*}: H_*(\Gamma^{\mathcal{S}}) \rightarrow H_*(\mathcal{S})$  is an isomorphism. Thus by the five lemma,  $\pi_{\mathcal{X}*}: H_*(\Gamma^{\overline{\{\xi\}}}) \rightarrow H_*(\overline{\{\xi\}})$  is an isomorphism as desired.  $\square$

*Proof of Theorem 2.24.* The result follows from applying Theorem 2.28 to the exact sequence for the pair  $(\mathcal{X}, \mathcal{X}_0)$  and then using Proposition 2.4.  $\square$



## 2.4 The induced map on homology

Theorem 2.24 allows us to define the morphism on homology induced by a topex map.

**Definition 2.29.** Let  $(\mathcal{F}, \mathcal{F}_0) : (\mathcal{X}, \mathcal{X}_0) \rightrightarrows (\mathcal{Y}, \mathcal{Y}_0)$  be an acyclic combinatorial pair map and assume that  $\mathcal{X}$  and  $\mathcal{Y}$  are augmentable. Denote the associated graph by  $(\Gamma \setminus \Gamma_0)$ . The induced map on homology  $\mathcal{F}_* : H_*(\mathcal{X} \setminus \mathcal{X}_0) \rightarrow H_*(\mathcal{Y} \setminus \mathcal{Y}_0)$  is defined by

$$\mathcal{F}_* := \pi'_{\mathcal{Y}*} \circ \pi_{\mathcal{X}*}^{-1},$$

where  $\pi'_{\mathcal{Y}*}$  and  $\pi_{\mathcal{X}*}$  denote respectively the maps in homology induced by the restrictions of  $\pi_{\mathcal{Y}}$  and  $\pi_{\mathcal{X}}$  to  $(\Gamma \setminus \Gamma_0)$ .

As is indicated earlier, one of the justifications for use of topex maps is that they provide readily computable approximations for continuous functions. The following result indicates that under reasonably mild conditions different approximations induce the same map on homology.

**Proposition 2.30.** Let  $\mathcal{T}, \mathcal{T}' : \mathcal{X}^+ \rightrightarrows \mathcal{Y}^+$  be topex maps where  $\mathcal{X}$  and  $\mathcal{Y}$  are augmentable complexes. Assume that the associated relative combinatorial maps  $\mathcal{F}$  and  $\mathcal{F}'$  are acyclic. If for all  $\xi \in \mathcal{X}^+$ ,

$$\mathcal{T}(\xi) \subset \mathcal{T}'(\xi)$$

then

$$\mathcal{F}_* = \mathcal{F}'_*$$

*Proof.* Observe that  $\mathcal{F}(\xi) \subset \mathcal{F}'(\xi)$  for all  $\xi \in \mathcal{X}^+$ . Thus, if  $\Gamma$  and  $\Gamma'$  denote the respective graph complexes then  $\Gamma \subset \Gamma'$ . The conclusion follows from the commutative diagram

$$\begin{array}{ccc} & \Gamma' & \\ \swarrow & \uparrow & \searrow \\ \mathcal{X} & \Gamma & \mathcal{Y} \end{array}$$

in which the vertical arrow denotes inclusion and all other arrows denote respective projections. □

## 3 Discrete Morse Theory

As indicated in the Introduction, the goal of this section is an algorithm that constructs a Morse complex starting with an arbitrary complex  $\mathcal{X}$  satisfying Definition 2.1. In Section 3.1 we provide an algorithm for producing the cells for the desired Morse complex. The underlying structure follows the reformulation of Forman's work by Chari [6] and Kozlov [23].

Given the cells to produce a Morse complex requires construction of the boundary operator. An algorithm for this is presented in Section 3.2 along with a proof that the homology of the resulting Morse complex agrees with that of the original complex. In Section 3.3 an algorithm for computing homology groups and its generators is presented. We conclude in Section 3.4 with an application of the discrete Morse homology to computing pre-images of boundaries.

### 3.1 Acyclic Matchings via Coreduction

Let  $(\mathcal{X}, \kappa)$  be a complex over the PID  $\mathbf{R}$ .

**Definition 3.1.** A *matching* of  $(\mathcal{X}, \kappa)$  consists of a partition of  $\mathcal{X}$  into three sets  $\mathcal{A}$ ,  $\mathcal{K}$ , and  $\mathcal{Q}$  along with a bijection  $w : \mathcal{Q} \rightarrow \mathcal{K}$ , such that for each  $Q \in \mathcal{Q}$  there exists a unit  $u \in \mathbf{R}$  satisfying

$$\kappa(w(Q), Q) = u, \tag{3}$$

We denote this decomposition by  $(\mathcal{A}, w : \mathcal{Q} \rightarrow \mathcal{K})$ .

Observe that by Definition 2.1(ii) and (3) we have  $\dim w(Q) = \dim Q + 1$  for each  $Q \in \mathcal{Q}$ .

**Remark 3.2.** The definition of a matching  $(\mathcal{A}, w: \mathcal{Q} \rightarrow \mathcal{K})$  is clearly related to earlier presentations of combinatorial Morse theory. See for example the work of Forman [14], Chari [6], and in particular Kozlov [23]. The elements of  $\mathcal{A}$  are typically referred to as critical cells in analogy to classical Morse theory. The elements of  $\mathcal{K}$  and  $\mathcal{Q}$  are often not explicitly labelled since from a purely Morse theoretic perspective they are unimportant objects, it is only the pairing  $w$  that plays an essential role. However, our interest is in using combinatorial Morse theory to develop algorithms that are designed to be applied to complexes arising from experimental or numerical data sets. In particular, as is made explicit in Algorithm 3.19 we can iteratively apply the preprocessing algorithm of this paper to the resulting Morse complex. This has no analogue in the classical Morse theory and in particular the critical cells of one complex cease to be critical cells in the next iterate of the algorithm. Similarly, in our computation of the induced maps on homology in Section 4, it is essential to be able to recover homology generators in the original complex. For this we need to keep track of the individual elements of  $\mathcal{A}$ ,  $\mathcal{K}$  and  $\mathcal{Q}$  and find it useful to have different monikers for the different elements of the pairing.

Given a matching of  $\mathcal{X}$  we define by transitivity a relation  $\leq$  on  $\mathcal{Q}$  as follows. Consider distinct elements  $Q, Q' \in \mathcal{Q}$ ;

$$\text{if } \kappa(w(Q), Q') \neq 0 \text{ then } Q' < Q. \quad (4)$$

**Definition 3.3.** A matching of  $\mathcal{X}$  is *acyclic* if  $\leq$  defines a partial order on  $\mathcal{Q}$ .

We now turn to the issue of constructing an acyclic matching on a complex  $\mathcal{X}$ . This is done in an essentially sequential manner: a cell in  $\mathcal{X}$  is identified as being either in  $\mathcal{A}$ ,  $\mathcal{K}$ , or  $\mathcal{Q}$  and then excised from the complex until the complex is exhausted. Observe that the only significant constraint in the definition is the bijection  $w: \mathcal{Q} \rightarrow \mathcal{K}$  which must be compatible with (3). This leads to the following definition.

**Definition 3.4.** (comp. [26, Section 4]) Given a subcomplex  $\mathcal{X}' \subset \mathcal{X}$ , a pair of cells  $(\xi, \xi') \in \mathcal{X}' \times \mathcal{X}'$  is a *coreduction pair* in  $\mathcal{X}'$  if  $\kappa(\xi, \xi')$  is a unit in  $\mathbf{R}$  and  $\text{bd}_{\mathcal{X}'} \xi = \{\xi'\}$ .

**Definition 3.5.** Given a subcomplex  $\mathcal{X}' \subset \mathcal{X}$ , a cell  $\xi \in \mathcal{X}'$  is *free* in  $\mathcal{X}'$  if  $\text{bd}_{\mathcal{X}'} \xi = \emptyset$ .

**Algorithm 3.6. Coreduction-Based Matching**

```

Given  $\mathcal{X}$ .
 $\mathcal{X}' \leftarrow \mathcal{X}$ .
while  $\mathcal{X}'$  is not empty do
  while  $\mathcal{X}'$  admits a coreduction pair  $(K, Q)$  do
    Excise the pair  $(K, Q)$  from  $\mathcal{X}'$ .
    Insert  $K \in \mathcal{K}$ , and  $Q \in \mathcal{Q}$ .
    Set  $w(Q) := K$ .
  end while
  while  $\mathcal{X}'$  does not admit a coreduction pair do
    Excise a free cell  $A$  from  $\mathcal{X}'$ .
    Insert  $A \in \mathcal{A}$ .
  end while
end while
return  $(\mathcal{A}, w: \mathcal{Q} \rightarrow \mathcal{K})$ 

```

**Theorem 3.7.** Given a complex  $\mathcal{X}$ , Algorithm 3.6 produces an acyclic matching  $(\mathcal{A}, w: \mathcal{Q} \rightarrow \mathcal{K})$ .

*Proof.* We begin by showing that the algorithm halts. Each of the secondary while statements results in a reduction in the size of the complex  $\mathcal{X}'$ . Notice that at each stage  $\mathcal{X}'$  remains a subcomplex of  $\mathcal{X}$  because excising a coreduction pair or a free cell results in a subcomplex. Thus it is sufficient to show that as long as  $\mathcal{X}' \neq \emptyset$  one or the other body of these while statements can be carried out. If a coreduction pair exists then the first while statement is performed. Otherwise, by finiteness  $\mathcal{X}'$  contains a cell  $\xi$  of minimal dimension and hence  $\xi$  is free. Therefore the algorithm outputs a partition  $\mathcal{A}, \mathcal{K}, \mathcal{Q}$  of  $\mathcal{X}$ . This is a matching since the definition of a coreduction pair implies (3). It remains to be shown that the matching is acyclic, that is, it must be shown that (4) defines a partial order on  $\mathcal{Q}$ .

Since excising a free cell  $A$  does not affect the partial order on  $\mathcal{Q}$ , we restrict our attention to the stages where we excise a coreduction pair  $(K, Q)$ . Observe that adding a new maximal element to a partially ordered set does not invalidate the partial order property. Accordingly, we show that  $Q$  will be maximal in  $\mathcal{Q}$  immediately after  $(K, Q)$  is excised. Otherwise, there must exist a previously excised coreduction pair  $(K', Q')$  with  $Q \in \text{bd}_{\mathcal{X}'} K'$ . But this is impossible, since at that stage  $\mathcal{X}''$  the boundary  $\text{bd}_{\mathcal{X}''} K'$  of  $K'$  must have contained both  $Q$  and  $Q'$ , thus violating the coreduction pair property of  $(K', Q')$  in  $\mathcal{X}''$ .  $\square$

### 3.2 Constructing the Morse Chain Complex

Given an acyclic matching  $(\mathcal{A}, w : \mathcal{Q} \rightarrow \mathcal{K})$  of  $\mathcal{X}$  we detail the construction of a new chain complex whose basis elements are generated by the cells in  $\mathcal{A}$ . Throughout this section, we use  $\text{id}_{\mathcal{X}}$  to denote the identity map on the chains  $C_*(\mathcal{X})$  and  $\langle \cdot, \cdot \rangle$  to denote the  $\mathbf{R}$ -valued inner product on  $C_*(\mathcal{X})$  obtained by treating the cells as an orthonormal basis.

**Definition 3.8.** The  $\mathcal{K}$ -chains and canonical chains in  $C_*(\mathcal{X})$  are the submodules  $C_*(\mathcal{K})$  and  $C_*(\mathcal{A}) \oplus C_*(\mathcal{K})$ , respectively.

The following proposition is fundamental to our construction of the boundary operator for the Morse complex.

**Proposition 3.9.** *There exists a unique module homomorphism  $\gamma_* : C_*(\mathcal{X}) \rightarrow C_{*+1}(\mathcal{X})$  such that the image of  $\text{id}_{\mathcal{X}} + \partial\gamma$  is canonical and  $\gamma(\mathbf{x})$  is a  $\mathcal{K}$ -chain for each  $\mathbf{x} \in C_*(\mathcal{X})$ . Furthermore,  $\ker \gamma \subset C_*(\mathcal{X})$  is the canonical submodule  $C_*(\mathcal{A}) \oplus C_*(\mathcal{K})$ .*

The proof of this proposition follows from Algorithm 3.12 given below. Let  $\iota_{\mathcal{A}} : C_*(\mathcal{A}) \rightarrow C_*(\mathcal{X})$  and  $\pi_{\mathcal{A}} : C_*(\mathcal{X}) \rightarrow C_*(\mathcal{A})$  be the usual inclusion and projection morphisms of modules. Define  $\psi : C_*(\mathcal{X}) \rightarrow C_*(\mathcal{A})$  and  $\phi : C_*(\mathcal{A}) \rightarrow C_*(\mathcal{X})$  by

$$\psi := \pi_{\mathcal{A}} \circ (\text{id}_{\mathcal{X}} + \partial\gamma) \text{ and } \phi := (\text{id}_{\mathcal{X}} + \gamma\partial) \circ \iota_{\mathcal{A}}. \quad (5)$$

Finally, define  $\Delta : C_*(\mathcal{A}) \rightarrow C_{*-1}(\mathcal{A})$  by

$$\Delta := \psi \circ \partial \circ \phi. \quad (6)$$

**Theorem 3.10.**  $(C_*(\mathcal{A}), \Delta)$  is a chain complex. Furthermore,  $\psi : C_*(\mathcal{X}) \rightarrow C_*(\mathcal{A})$  and  $\phi : C_*(\mathcal{A}) \rightarrow C_*(\mathcal{X})$  are chain equivalences.

The following corollary is an immediate conclusion of this theorem.

**Corollary 3.11.** *The homology of the chain complex  $(C_*(\mathcal{X}), \partial)$  is isomorphic to the homology of the chain complex  $(C_*(\mathcal{A}), \Delta)$ .*

Clearly the proof of Theorem 3.10 depends on Proposition 3.9. For applications we need an explicit representation of  $\Delta$  and hence a means of constructing  $\gamma$  from an acyclic matching of the original chain complex  $(C_*(\mathcal{X}), \partial)$ . This is done using the following algorithm.

#### Algorithm 3.12. Gamma Algorithm

```

Given  $\mathbf{x}_{\text{in}} \in C_*(\mathcal{X})$ 
 $\mathbf{x} \leftarrow \mathbf{x}_{\text{in}}$ 
 $\mathbf{c} \leftarrow 0 \in C_*(\mathcal{X})$ 
while  $\mathbf{x} \notin C_*(\mathcal{A}) \oplus C_*(\mathcal{K})$  do
  Choose a  $\leq$ -maximal  $Q \in \mathcal{Q}$  so that  $\langle \mathbf{x}, Q \rangle \neq 0$ .
   $K \leftarrow w(Q)$ 
   $\omega \leftarrow -\langle \mathbf{x}, Q \rangle / \kappa(K, Q)$ 
   $\mathbf{c} \leftarrow \mathbf{c} + \omega K$ 
   $\mathbf{x} \leftarrow \mathbf{x} + \omega \partial K$ 
end while
return  $\mathbf{c}$ 

```

**Proposition 3.13.** *If Algorithm 3.12 is called with an arbitrary chain  $\mathbf{x} \in C_*(\mathcal{X})$ , then it returns a  $\mathcal{K}$ -chain  $\mathbf{c}$  such that  $\mathbf{x} + \partial\mathbf{c}$  is canonical. Furthermore, if  $\mathbf{x}$  is canonical then  $\mathbf{c} = 0$ .*

*Proof.* Let  $\{\mathbf{x}^i\}$  and  $\{\mathbf{c}^i\}$  be the sequences of values of  $\mathbf{x}$  and  $\mathbf{c}$  produced by Algorithm 3.12. An inductive argument shows that

$$\mathbf{x}^i = \mathbf{x}_{\text{in}} + \partial\mathbf{c}^i.$$

We now show that the algorithm halts. Let  $Q_i$  denote the maximal element chosen from  $\mathbf{x}^i$ . More precisely, let  $Q_i$  be a  $\leq$ -maximal element from the set  $\{Q \in \mathcal{Q} \mid \langle \mathbf{x}^i, Q \rangle \neq 0\}$ . A direct calculation shows that  $\langle \mathbf{x}^{i+1}, Q_i \rangle = 0$ . Furthermore, the maximality of  $Q_i$  implies that  $\langle \mathbf{x}^{i+j}, Q_i \rangle = 0$ , for all  $j \geq 1$ .

Since the set  $\{Q \in \mathcal{Q} \mid \langle \mathbf{x}, Q \rangle \neq 0\}$  is finite, there exists  $j$  such that

$$\{Q \in \mathcal{Q} \mid \langle \mathbf{x}^j, Q \rangle \neq 0\} = \emptyset,$$

in which case  $\mathbf{x}^j$  is canonical and the algorithm halts.  $\square$

The following result is used to show that  $\gamma$  is uniquely defined.

**Proposition 3.14.** *Consider a  $\mathcal{K}$ -chain  $\mathbf{x}$ . If  $\partial\mathbf{x}$  is canonical, then  $\mathbf{x} = 0$ .*

*Proof.* Assume  $\mathbf{x} \neq 0$ . Choose  $K \in \mathcal{K}$  such that  $Q = w^{-1}(K)$  is  $\leq$ -maximal in the set

$$\{w^{-1}(K') \in \mathcal{Q} \mid \langle \mathbf{x}, K' \rangle \neq 0\}.$$

By (3),  $\kappa(K, Q) \neq 0$ . However, by assumption  $\partial\mathbf{x}$  is canonical and so  $\langle \partial\mathbf{x}, Q \rangle = 0$ . Since the coefficient ring  $\mathbf{R}$  has no zero divisors, there exists  $K' \in \mathcal{K}$  such that  $K \neq K'$ ,  $\langle \mathbf{x}, K' \rangle \neq 0$  and  $\kappa(K', Q) \neq 0$ . But now  $Q \leq w^{-1}(K')$ , contradicting the maximality of  $Q$ .  $\square$

*Proof of Proposition 3.9.* Proposition 3.13 proves the existence of  $\gamma$ . It remains to be shown that  $\gamma$  is unique. So assume that given  $\mathbf{x} \in C_*(\mathcal{X})$  there exist  $\mathcal{K}$ -chains  $\mathbf{c}_1, \mathbf{c}_2$  such that  $\mathbf{x} + \partial\mathbf{c}_i$  is canonical for  $i \in \{1, 2\}$ . Then,  $\mathbf{c}_1 - \mathbf{c}_2$  is a  $\mathcal{K}$ -chain and  $\partial(\mathbf{c}_1 - \mathbf{c}_2)$  is canonical, and hence by Proposition 3.14  $\mathbf{c}_1 = \mathbf{c}_2$ .  $\square$

Before turning the proof of Theorem 3.10 we state an important property of  $\gamma$ .

**Proposition 3.15.** *The following sequence is exact:*

$$C_*(\mathcal{X}) \xrightarrow{\text{id}_X + \partial\gamma} C_*(\mathcal{X}) \xrightarrow{\gamma} C_{*+1}(\mathcal{X}) \xrightarrow{\text{id}_X + \gamma\partial} C_{*+1}(\mathcal{X})$$

*In particular,  $\ker \gamma = \text{img}(\text{id}_X + \partial\gamma)$  is the canonical submodule and  $\text{img} \gamma = \ker(\text{id}_X + \gamma\partial)$  is the submodule of  $\mathcal{K}$ -chains.*

*Proof.* Both  $\ker \gamma$  and  $\text{img}(\text{id}_X + \partial\gamma)$  consist precisely of canonical chains by Proposition 3.9, and so it suffices to check that  $\ker(\text{id}_X + \gamma\partial) = \text{img} \gamma$  is the set of  $\mathcal{K}$ -chains. Let  $\mathbf{x}$  be a  $\mathcal{K}$ -chain. By Proposition 3.9,  $\gamma(\partial\mathbf{x})$  is the unique  $\mathcal{K}$ -chain such that

$$\partial\mathbf{x} + \partial\gamma(\partial\mathbf{x}) \text{ is canonical.}$$

Since 0 is a canonical chain and  $-\mathbf{x}$  is a  $\mathcal{K}$ -chain, it follows that  $\gamma(\partial\mathbf{x}) = -\mathbf{x}$ . There are two immediate consequences: first,  $\mathbf{x}$  lies in the image of  $\gamma$ , and second  $(\text{id}_X + \gamma\partial)\mathbf{x} = 0$  as desired. Conversely, if  $\mathbf{x} \in \ker(\text{id}_X + \gamma\partial)$ , then  $\mathbf{x} + \mathbf{k} = 0$  for some  $\mathcal{K}$ -chain  $\mathbf{k}$  and thus  $\mathbf{x}$  is a  $\mathcal{K}$ -chain.  $\square$

**Lemma 3.16.**  $\phi \circ \psi = \text{id}_X + \partial\gamma + \gamma\partial$  on  $C_*(\mathcal{X})$ .

*Proof.* By definition, we have  $\phi \circ \psi = (\text{id}_X + \gamma\partial) \circ \iota_A \circ \pi_A \circ (\text{id}_X + \partial\gamma)$ . Since  $\text{img}(\text{id}_X + \partial\gamma)$  is canonical by Proposition 3.15,  $\iota_A \circ \pi_A \circ (\text{id}_X + \partial\gamma)$  differs from  $(\text{id}_X + \partial\gamma)$  only by a  $\mathcal{K}$ -chain  $\mathbf{k}$  which gets projected to 0 by  $\pi_A$ . However, since  $\mathbf{k}$  lies in  $\ker(\text{id}_X + \gamma\partial)$  again by Proposition 3.15, we have:

$$\phi \circ \psi = (\text{id}_X + \gamma\partial) \circ (\text{id}_X + \partial\gamma),$$

from which the desired result follows by using  $\partial \circ \partial \equiv 0$ .  $\square$

**Corollary 3.17.**  $(C_*(\mathcal{A}), \Delta)$  is a chain complex.

*Proof.* It suffices to show that  $\Delta \circ \Delta \equiv 0$  on  $C_*(\mathcal{A})$ . By definition,

$$\Delta \circ \Delta = (\psi \circ \partial \circ \phi) \circ (\psi \circ \partial \circ \phi)$$

But by Lemma 3.16, the inner composition  $\phi \circ \psi$  equals  $(\text{id}_X + \partial\gamma + \gamma\partial)$ . Using  $\partial \circ \partial \equiv 0$ , we get  $\partial \circ (\phi \circ \psi) \circ \partial \equiv 0$  as desired.  $\square$

**Proposition 3.18.** *The morphisms  $\psi : C_*(\mathcal{X}) \rightarrow C_*(\mathcal{A})$  and  $\phi : C_*(\mathcal{A}) \rightarrow C_*(\mathcal{X})$  are chain maps.*

*Proof.* To see that  $\psi$  is a chain map, consider  $\Delta \circ \psi = (\psi \circ \partial \circ \phi) \circ \psi$ . By Lemma 3.16, this composition equals  $\psi \circ \partial \circ (\text{id}_X + \partial\gamma + \gamma\partial)$ . Using  $\partial \circ \partial \equiv 0$ , we have

$$\Delta \circ \psi = \psi \circ \partial \circ (\text{id}_X + \gamma\partial) = \psi \circ (\text{id}_X + \partial\gamma) \circ \partial.$$

Since  $\psi = \pi_A \circ (\text{id}_X + \partial\gamma)$  by definition, we have  $\Delta \circ \psi = \pi_A \circ (\text{id}_X + \partial\gamma) \circ (\text{id}_X + \partial\gamma) \circ \partial$ . By Proposition 3.15,  $\gamma \circ (\text{id}_X + \partial\gamma)$  is trivial, and so  $\Delta \circ \psi = \pi_A \circ (\text{id}_X + \partial\gamma) \circ \partial = \psi \circ \partial$  as desired. A very similar argument establishes that  $\phi \circ \Delta = \partial \circ \phi$ ; we leave this to the reader.  $\square$

Finally, we are ready to prove Theorem 3.10.

*Proof of Theorem 3.10.* It suffices to check that  $\psi$  and  $\phi$  are chain equivalences. On one hand,  $\gamma$  provides a chain homotopy on  $C_*(\mathcal{X})$  between  $\phi \circ \psi$  and the identity  $1_{\mathcal{X}}$  by Lemma 3.16. On the other hand, we claim that  $\psi \circ \phi$  equals the identity map  $1_{\mathcal{A}}$  on  $C_*(\mathcal{A})$ . To see this, observe that

$$\psi \circ \phi = \pi_{\mathcal{A}} \circ (\text{id}_{\mathcal{X}} + \partial\gamma) \circ (\text{id}_{\mathcal{X}} + \gamma\partial) \circ \iota_{\mathcal{M}}.$$

Since  $\text{img } \gamma = C_*(\mathcal{K}) \subset C_*(\mathcal{K}) \oplus C_*(\mathcal{A}) = \ker \gamma$  by Propositions 3.15 and 3.9, we have  $\gamma \circ \gamma = 0$ ,  $\pi_{\mathcal{A}} \circ \gamma = 0$  and  $\gamma \circ \iota_{\mathcal{A}} = 0$ . Therefore,

$$\psi \circ \phi = \pi_{\mathcal{A}} \circ \text{id}_{\mathcal{X}} \circ \iota_{\mathcal{A}} = \text{id}_{\mathcal{A}}$$

as desired.  $\square$

### 3.3 Computing Homology of Complexes

We conclude this section by presenting Algorithm 3.19 which computes both the homology of a complex and a set of generators of the homology.

We begin with the observation that with regard to the presentation in this paper there are two natural inputs: the complex and the incidence number  $(\mathcal{X}, \kappa)$ , or the complex and the boundary operator  $(\mathcal{X}, \partial)$ . The particular choice of input should be thought of as being problem dependent. Recall that in Definition 2.1 the boundary operator is explicitly given in terms of the incidence number. Similarly, if the boundary operator is defined in terms of the basis  $\mathcal{X}$  of  $C_*(\mathcal{X})$ , then the incidence number can be derived explicitly from the boundary operator. For simplicity of presentation we choose the input  $(\mathcal{X}, \partial)$ .

#### Algorithm 3.19. Homology Generators Algorithm

**given**  $(\mathcal{X}, \partial)$

Using Algorithm 3.6 produce an acyclic matching  $(\mathcal{A}, w: \mathcal{Q} \rightarrow \mathcal{K})$ .

**for all**  $A \in \mathcal{A}$  **do**

compute and store  $\Delta A$  using Algorithm 3.12 and definition of  $\Delta$ .

**end for**

**if**  $\mathcal{A} = \mathcal{X}$  **then**

Use Smith Normal Form to compute homology generators  $\{\mathbf{g}_i\}$  of  $(C_*(\mathcal{A}), \Delta)$ .

**end if**

**return**  $\{\phi(\mathbf{g}_i)\}$  (or optionally  $\{\mathbf{g}_i\}$ ,  $(\mathcal{A}, \Delta)$ ,  $\phi$ , and  $\psi$ .)

Note that by Theorem 3.10 the output  $(\mathcal{A}, \Delta)$  is a complex. Thus we may call this routine iteratively with the output of a previous iteration becoming the input to a current iteration until eventually the size of the complex stabilizes.

**Theorem 3.20.** *If Algorithm 3.19 is called with a chain complex  $(\mathcal{X}, \partial)$ , then it returns  $H_*(\mathcal{X})$  and a set of cycles  $\{\phi(\mathbf{g}_i)\} \subset Z_*(\mathcal{X})$  which generates  $H_*(\mathcal{X})$ .*

*Proof.* The proof follows from Theorem 3.7, Proposition 3.9 and Theorem 3.10. The only question is whether it halts. Observe that if  $\mathcal{A} = \mathcal{X}$ , then a Smith Normal Form algorithm is called and produces the desired output. If  $\mathcal{A} \neq \mathcal{X}$ , then by Algorithm 3.6,  $\mathcal{A} \subset \mathcal{X}$ . Since  $\mathcal{X}$  is finite, eventually  $\mathcal{A} = \mathcal{X}$ .  $\square$

If one applies Algorithm 3.19 to a complex  $(\mathcal{X}, \partial)$ , then at the end of the **for all** loop one can extract an *associated Morse complex* which consists of the cell complex  $(\mathcal{M}, \Delta)$  and the homomorphisms  $\{\gamma, \psi, \phi\}$ .

### 3.4 Computing Preboundaries

Let  $(\mathcal{X}, \kappa)$  be a complex with associated boundary operator  $\partial$ . Recall that  $B_*(\mathcal{X}) = \text{img } \partial$  is the submodule of  $C_*(\mathcal{X})$  consisting of boundaries. In this section we indicate how the Morse preprocessing methods can be used to solve the following problem. Given  $\mathbf{b} \in B_*(\mathcal{X})$  find its preboundary  $\mathbf{c}$ , that is, solve

$$\partial \mathbf{c} = \mathbf{b}. \tag{7}$$

It is a classical result that Smith Normal Form can be used to find a solution (see [20, Algorithm 3.54] for an explicit algorithm). In the same spirit as the rest of the paper, we expect that it is more efficient to apply the Smith Normal Form algorithm on the smaller Morse complex. To be more precise, assume an

algorithm (such as [20, Algorithm 3.54]) has been chosen to find preboundaries. This implies the existence of a well defined map

$$P_{\mathcal{X}}: B_*(\mathcal{X}) \rightarrow C_{*+1}(\mathcal{X}), \quad (8)$$

such that  $\partial P_{\mathcal{X}}(\mathbf{c}) = \mathbf{b}$ .

**Theorem 3.21.** *Let  $(\mathcal{X}, \partial)$  be a complex with an associated Morse complex  $(\mathcal{A}, \Delta)$  and homomorphisms  $\{\gamma, \psi, \phi\}$ . Let  $\mathbf{b} \in B_*(\mathcal{X})$ . If*

$$\mathbf{c} := \phi \circ P_{\mathcal{A}} \circ \psi(\mathbf{b}) - \gamma(\mathbf{b}),$$

*then  $\partial \mathbf{c} = \mathbf{b}$ .*

*Proof.* This can be shown by a direct computation. Define  $\mathbf{c}$  as above. Then

$$\partial \mathbf{c} = \partial [\phi \circ P_{\mathcal{A}} \circ \psi(\mathbf{b}) - \gamma(\mathbf{b})].$$

Since  $\phi$  is a chain map, we may write  $\partial \circ \phi = \phi \circ \Delta$ , and so we have

$$\partial \mathbf{c} = \phi \circ \Delta \circ P_{\mathcal{A}} \circ \psi(\mathbf{b}) - \partial \circ \gamma(\mathbf{b}).$$

Since  $\psi$  is a chain map,  $\psi(\mathbf{b})$  is again a boundary, and we have  $\Delta \circ P_{\mathcal{A}} \circ \psi(\mathbf{b}) = \psi(\mathbf{b})$ . Hence, one obtains

$$\partial \mathbf{c} = \phi \circ \psi(\mathbf{b}) - \partial \circ \gamma(\mathbf{b}).$$

Since  $\gamma$  is a chain homotopy (by Lemma 3.16), we have  $\partial \circ \gamma = \phi \circ \psi - 1_{\mathcal{X}} - \gamma \circ \partial$ .

$$\partial \mathbf{c} = \phi \circ \psi(\mathbf{b}) - [\phi \circ \psi - 1_{\mathcal{X}} - \gamma \circ \partial](\mathbf{b}) = \mathbf{b} + \gamma \circ \partial(\mathbf{b}).$$

Finally, since  $\mathbf{b} \in B_*(\mathcal{X})$  we have  $\partial \mathbf{b} = 0$ . Thus,

$$\partial \mathbf{c} = \mathbf{b}.$$

□

**Remark 3.22.** As in the computation of homology, one can attempt to minimize the cost of computing  $P_{\mathcal{M}}$  by using this formula iteratively, that is, one can iterate the application of Algorithm 3.19 to produce a Morse complex of minimal size.

## 4 Computing the Induced Map on Homology

Throughout this section,  $(\mathcal{F}, \mathcal{F}_0): (\mathcal{X}, \mathcal{X}_0) \rightrightarrows (\mathcal{Y}, \mathcal{Y}_0)$  denotes an acyclic combinatorial pair map where  $(\mathcal{X}, \kappa)$  and  $(\mathcal{Y}, \tau)$  are augmentable complexes. This section provides an algorithm for computing the induced map on relative homology

$$\mathcal{F}_*: H_*(\mathcal{X}, \mathcal{X}_0) \rightarrow H_*(\mathcal{Y}, \mathcal{Y}_0).$$

The computation makes use of the graph complex  $(\Gamma \setminus \Gamma_0)$  of  $(\mathcal{F}, \mathcal{F}_0)$  and therefore in what follows we assume that  $(\mathcal{F}, \mathcal{F}_0)$  is acyclic. We use  $\mathfrak{d}$  to denote the boundary operator on the chains  $C_*(\Gamma \setminus \Gamma_0)$  of the graph. Note that we have the relation

$$\mathfrak{d} = \pi_{\Gamma \setminus \Gamma_0} \circ \partial^\times \circ \iota_{\Gamma \setminus \Gamma_0},$$

where  $\pi_{\Gamma \setminus \Gamma_0}$  and  $\iota_{\Gamma \setminus \Gamma_0}$  are the projection and inclusion maps between  $C_*((\mathcal{X} \times \mathcal{Y}) \setminus (\mathcal{X}_0 \times \mathcal{Y}_0))$  and  $C_*(\Gamma \setminus \Gamma_0)$  and  $\partial^\times$  is the boundary operator on the product complex. As usual, the fiber  $(\Gamma^\xi \setminus \Gamma_0^\xi)$  associated to each cell  $\xi \in \mathcal{X}$  is a subcomplex of the graph, and we will denote boundary operator of this complex by  $\mathfrak{d}^\xi$  throughout this section. Note that projection maps of the type  $\pi_{\Gamma \setminus \Gamma_0}^\xi: C_*(\Gamma \setminus \Gamma_0) \rightarrow C_*(\Gamma^\xi \setminus \Gamma_0^\xi)$  are *not* chain maps in general, since  $\Gamma^\xi$  need not be a closed subcomplex of  $\Gamma$ . However, for each  $\mathbf{g} \in C_*(\Gamma^\xi \setminus \Gamma_0^\xi)$  we have

$$\mathfrak{d}^\xi(\mathbf{g}) = \pi_{\Gamma \setminus \Gamma_0}^\xi \circ \mathfrak{d}(\mathbf{g}).$$

## 4.1 Lifting Cycles to the Graph

**Definition 4.1.** Given a cycle  $\mathbf{x} \in Z_*(\mathcal{X} \setminus \mathcal{X}_0)$ , we say that  $\mathbf{g} \in C_k(\Gamma \setminus \Gamma_0)$  is a *graph cycle* corresponding to  $\mathbf{x}$  if  $\mathbf{g} \in Z_*(\Gamma \setminus \Gamma_0)$  and  $\pi_{\mathcal{X}}(\mathbf{g}) = \mathbf{x}$ .

To make use of Definition 2.29, for each cycle  $\mathbf{x} \in C_*(\mathcal{X} \setminus \mathcal{X}_0)$  we need to construct a corresponding graph cycle  $\mathbf{g}$ . Since  $\pi_{\mathcal{X}}$  is an isomorphism, the requirement that  $\pi_{\mathcal{X}}(\mathbf{g})$  equal  $\mathbf{x}$  suffices to establish the well-definedness on homology of such a construction. Thus, the remainder of this section is devoted to building a graph cycle via Algorithms 4.2 and 4.5 which are based on the acyclicity of the combinatorial pair map  $(\mathcal{F}, \mathcal{F}_0)$ .

### Algorithm 4.2. Chain Lift

**Given**  $\mathbf{x} \in C_k(\mathcal{X} \setminus \mathcal{X}_0)$ ,  $(\mathcal{F}, \mathcal{F}_0): (\mathcal{X}, \mathcal{X}_0) \rightarrow (\mathcal{Y}, \mathcal{Y}_0)$   
 $\mathbf{g}_0 \leftarrow 0$   
**for all**  $\xi \in \mathcal{X} \setminus \mathcal{X}_0$  with  $\dim \xi = k$  **do**  
  **if**  $\langle \mathbf{x}, \xi \rangle \neq 0$  **then**  
    Choose  $\eta \in \mathcal{F}(\xi)$  such that  $\dim \eta = 0$   
     $\mathbf{g}_0 \leftarrow \mathbf{g}_0 + \langle \mathbf{x}, \xi \rangle \xi \otimes \eta$   
  **end if**  
**end for**  
**return**  $\mathbf{g}_0$

**Proposition 4.3.** If Algorithm 4.2 is applied to  $\mathbf{x} \in C_k(\mathcal{X} \setminus \mathcal{X}_0)$ , then the output is of the form

$$\mathbf{g}_0 = \sum_{\xi_j \in \mathcal{X}^k} r_j (\xi_j \otimes \eta_j) \in C_k(\Gamma \setminus \Gamma_0),$$

where  $\xi_j \in \mathcal{X} \setminus \mathcal{X}_0$  have dimension  $k$ ,  $\eta_j \in \mathcal{Y} \setminus \mathcal{Y}_0$  have dimension zero, and  $r_j \in \mathbf{R}$ . Furthermore,

$$\pi_{\mathcal{X}}(\mathbf{g}_0) = \mathbf{x}.$$

*Proof.* Since  $\mathbf{x} \in C_k(\mathcal{X} \setminus \mathcal{X}_0)$  there is a unique representation  $\mathbf{x} = \sum r_j \xi_j$  where  $\xi \in \mathcal{X} \setminus \mathcal{X}_0$  and  $r_j \in \mathbf{R}$ . By Proposition 2.23 each  $(\Gamma^{\xi_j} \setminus \Gamma_0^{\xi_j})$  has the homology of a  $k$ -cell, and hence there exists some  $\eta_j \in \mathcal{F}(\xi_j) \setminus \mathcal{F}_0(\xi_j)$  of dimension zero. Clearly,  $\mathbf{g}_0 = \sum r_j (\xi_j \otimes \eta_j)$  projects to  $\mathbf{x}$  under  $\pi_{\mathcal{X}}$ .  $\square$

While applying Algorithm 4.2 to a cycle  $\mathbf{x} \in Z_*(\mathcal{X} \setminus \mathcal{X}_0)$  will produce a chain  $\mathbf{g}_0 \in C_*(\Gamma \setminus \Gamma_0)$  projecting to  $\mathbf{x}$  via  $\pi_{\mathcal{X}}$ , in general  $\mathbf{g}_0 \notin Z_*(\Gamma \setminus \Gamma_0)$  so  $\mathbf{g}_0$  is not a graph cycle corresponding to  $\mathbf{x}$ . The following algorithm modifies  $\mathbf{g}_0$  on the fibers of  $(\mathcal{F}, \mathcal{F}_0)$  to produce a graph cycle.

**Remark 4.4.** As is suggested above and made explicit in the following algorithm our approach to computing the induced map on homology requires that we solve the preboundary problem (7) on fibers. Thus the particular implementation, e.g. (8), Theorem 3.21 combined with (8), or an alternative geometric method such as that of Allili and Kaczynski [2], will have a significant effect on run time depending on the structure and/or size of  $(\Gamma \setminus \Gamma_0)$ . With this in mind, we do not include a specific algorithm for solving the preboundary problem in the following algorithm.

### Algorithm 4.5. Cycle Lift

**Given**  $\mathbf{x} \in Z_*(\mathcal{X} \setminus \mathcal{X}_0)$ ,  $(\mathcal{F}, \mathcal{F}_0): (\mathcal{X}, \mathcal{X}_0) \rightrightarrows (\mathcal{Y}, \mathcal{Y}_0)$   
 $k \leftarrow \dim \mathbf{x}$   
 $\mathbf{g}_0 \leftarrow$  Apply Algorithm 4.2 to  $\mathbf{x}$ .  
**for**  $i = 0$  to  $k - 1$  **do**  
   $j \leftarrow k - i - 1$   
  **for all** cells  $\xi \in \mathcal{X}$  with  $\dim \xi = j$  **do**  
     $\mathbf{b}_i^\xi \leftarrow \pi_{\Gamma \setminus \Gamma_0}^\xi \circ \mathfrak{d}(\mathbf{g}_i)$   
    Choose  $\mathbf{c}_i^\xi \in C_k(\Gamma^\xi \setminus \Gamma_0^\xi)$  so that  $\mathfrak{d}^\xi(\mathbf{c}_i^\xi) = \mathbf{b}_i^\xi$  (i.e. solve the *preboundary problem*.)  
  **end for**  
   $\mathbf{g}_{i+1} \leftarrow \mathbf{g}_i - \sum_{\dim \xi = j} \mathbf{c}_i^\xi$   
**end for**  
**return**  $\mathbf{g}_k$

## 4.2 Verification

The goal of this section is to prove the following theorem:

**Theorem 4.6.** *If Algorithm 4.5 is applied to a cycle  $\mathbf{x} \in Z_*(\mathcal{X} \setminus \mathcal{X}_0)$ , then the output  $\mathbf{g}_k$  is a graph cycle corresponding to  $\mathbf{x}$ .*

In order to prove Theorem 4.6 we must first show that Algorithm 4.12 executes, and the only non-trivial part lies in the preboundary step. That is, we must establish that each  $\mathbf{b}_i^\xi$  is a boundary in the fiber  $\Gamma^\xi \setminus \Gamma_0^\xi$ . To achieve this, we modify slightly Algorithm 4.5 without changing its behaviour. First, we assume that the algorithm stops immediately and declares failure whenever the preboundary problem has no solution. With this change, our goal may be reformulated as proving that the algorithm never declares failure. Moreover, we let the outer loop range over  $i \in \{0, 1, \dots, k\}$ , noting that the final  $k$ -th iteration does nothing of value. Also, we will let the inner loop range over all  $\xi \in \mathcal{X}$ , rather than just those  $\xi$  with  $\dim \xi = j(i)$ . Again, this does nothing of value since all the extra  $\mathbf{b}_i^\xi$  and  $\mathbf{c}_i^\xi$  variables calculated will all be zero, as we shall see. Finally, we split the inner loop into two loops: one evaluating all the  $\mathbf{b}_i^\xi$  for all  $\xi \in \mathcal{X}$  and the other solving the respective preboundary problems. Our strategy is to show that for each  $i \in \{0, 1, \dots, k\}$  the following three propositions are true:

$P(i)$ : The algorithm does not fail before entering the  $i$ th pass of the outer loop, and for all  $\xi \in \mathcal{X}$  such that  $\dim \xi \neq j(i)$ ,

$$\mathbf{b}_i^\xi = 0.$$

$Q(i)$ : The algorithm does not fail before entering  $i$ th pass of the outer loop, and

$$\mathfrak{d}(\mathbf{g}_i) = \sum_{\dim \xi = j(i)} \mathbf{b}_i^\xi.$$

$R(i)$ : The algorithm does not fail before entering  $i$ th pass of the outer loop, and for all  $\xi \in \mathcal{X}$ , we have

$$\mathbf{b}_i^\xi \in B_{k-1}(\Gamma^\xi \setminus \Gamma_0^\xi).$$

The proposition  $R(i)$  is of particular importance as it could easily be rewritten as:

$R(i)$ : *The algorithm does not fail before completing the  $i$ th pass of the outer loop.*

This is because the only possible obstacle to executing Step  $i$  is the failure of a preboundary to exist, but  $R(i)$  guarantees that it does.

We will prove these statements in a round-robin inductive fashion over  $i \in \{0, \dots, k\}$  as follows. As a base step, we establish that  $P(0)$  holds in Proposition 4.7. For the inductive step, we have

$$\begin{cases} R(i-1) \text{ and } P(i-1) \text{ imply } P(i) & \text{Proposition 4.9} \\ P(i) \text{ implies } Q(i) & \text{Proposition 4.10, and} \\ Q(i) \text{ implies } R(i) & \text{Proposition 4.11.} \end{cases}$$

We postpone the proofs of these Propositions. Assuming their truth for the moment, we prove the main result of this section.

*Proof of Theorem 4.6.* From  $R(k)$ , we conclude that Algorithm 4.5 indeed executes and produces  $\mathbf{g}_k$ . By  $Q(k)$ , we have

$$\mathfrak{d}(\mathbf{g}_k) = \sum_{\dim \xi = -1} \mathbf{b}_k^\xi = 0,$$

which proves that the output  $\mathbf{g}_k$  is indeed a cycle in the graph  $\Gamma \setminus \Gamma_0$ . It remains only to show that  $\mathbf{g}_k$  is a graph cycle corresponding to  $\mathbf{x}$ . That is, we must show that  $\pi_{\mathcal{X}}(\mathbf{g}_k) = \mathbf{x}$ . We prove by induction on  $i$  that  $\pi_{\mathcal{X}}(\mathbf{g}_i) = \mathbf{x}$ . As a base step,  $\pi_{\mathcal{X}}(\mathbf{g}_0) = \mathbf{x}$  by Proposition 4.3. For the inductive step, consider the hypothesis true for some  $i > 0$  and observe that

$$\pi_{\mathcal{X}}(\mathbf{g}_{i+1}) = \pi_{\mathcal{X}}(\mathbf{g}_i) - \sum_{\dim \xi = j(i)} \pi_{\mathcal{X}}(\mathbf{c}_i^\xi).$$

But each summand  $\pi_{\mathcal{X}}(\mathbf{c}_i^\xi)$  on the right side vanishes because  $\dim \xi = j(i) = k - i - 1 \neq k = \dim \mathbf{c}_i^\xi$ .

$$\pi_{\mathcal{X}}(\mathbf{g}_{i+1}) = \pi_{\mathcal{X}}(\mathbf{g}_i) = \mathbf{x},$$

where the second equality comes from the inductive hypothesis. This concludes the proof, modulo proofs of Propositions 4.7, 4.9, 4.10, and 4.11.  $\square$



The following proposition provides the base case for the inductive process outlined above.

**Proposition 4.7.**  $P(0)$  holds. That is,  $\mathbf{b}_0^\xi = 0$  for all cells  $\xi \in \mathcal{X}$  with  $\dim \xi \neq j(0) = k - 1$ .

*Proof.* Note by Proposition 4.3 that  $\mathbf{g}_0$  is an  $\mathbf{R}$ -linear combination of cells of the type  $\zeta \otimes \eta$  where  $\dim \zeta = k$  and  $\dim \eta = 0$ . Therefore, the boundary  $\partial(\mathbf{g}_0)$  in the graph is a combination of cells of the type  $\zeta' \otimes \eta$  where  $\zeta'$  has dimension  $k - 1$  and  $\eta$  has dimension 0. Since  $\dim \xi \neq k - 1$  by assumption,  $\langle \xi, \zeta' \rangle = 0$  for all such  $\zeta'$ . Thus,  $\mathbf{b}_0^\xi = \pi_{\Gamma \setminus \Gamma_0}^\xi \circ \partial(\mathbf{g}_0)$  is trivial as desired.  $\square$

The following lemma facilitates proofs of the three subsequent propositions which comprise the inductive step of our argument.

**Lemma 4.8.** Let  $\xi, \zeta \in \mathcal{X}$  be distinct cells such that  $\xi \not\prec \zeta$ , i.e.  $\kappa(\zeta, \xi) = 0$ . Then

$$\pi_{\Gamma \setminus \Gamma_0}^\xi \circ \partial \equiv 0 \text{ on } C_*(\Gamma^\zeta \setminus \Gamma_0^\zeta).$$

*Proof.* Without loss of generality  $\Gamma^\zeta \setminus \Gamma_0^\zeta$  is non-empty, so pick any  $\eta \in \mathcal{F}(\zeta) \setminus \mathcal{F}_0(\zeta)$ . We will show that  $\pi_{\Gamma \setminus \Gamma_0}^\xi \circ \partial(\zeta \otimes \eta)$  is trivial and extend bilinearly to chains in  $C_*(\Gamma^\zeta \setminus \Gamma_0^\zeta)$ . By definition,

$$\partial(\zeta \otimes \eta) = \pi_{\Gamma \setminus \Gamma_0}(\partial^x \zeta \otimes \eta + (-1)^{\dim \zeta} \zeta \otimes \partial^y \eta).$$

Projecting to the fiber of  $\xi$ , we have

$$\pi_{\Gamma \setminus \Gamma_0}^\xi \circ \partial(\zeta \otimes \eta) = \pi_{\Gamma \setminus \Gamma_0}^\xi \circ \pi_{\Gamma \setminus \Gamma_0}(\partial^x \zeta \otimes \eta + (-1)^{\dim \zeta} \zeta \otimes \partial^y \eta).$$

Since the fiber  $\Gamma^\xi \setminus \Gamma_0^\xi$  is a subcomplex of the graph  $\Gamma \setminus \Gamma_0$ , the projections satisfy  $\pi_{\Gamma \setminus \Gamma_0}^\xi \circ \pi_{\Gamma \setminus \Gamma_0} \equiv \pi_{\Gamma \setminus \Gamma_0}^\xi$  on the chains of the product complex  $(\mathcal{X} \times \mathcal{Y}) \setminus (\mathcal{X}_0 \times \mathcal{Y}_0)$ . Therefore,

$$\begin{aligned} \pi_{\Gamma \setminus \Gamma_0}^\xi \circ \partial(\zeta \otimes \eta) &= \pi_{\Gamma \setminus \Gamma_0}^\xi (\partial^x \zeta \otimes \eta + (-1)^{\dim \zeta} \zeta \otimes \partial^y \eta) \\ &= \langle \xi, \partial^x \zeta \rangle \xi \otimes \eta + (-1)^{\dim \zeta} \langle \xi, \zeta \rangle \xi \otimes \partial^y \eta, \end{aligned}$$

since  $\xi \neq \zeta$  the second term vanishes, and since  $\xi \not\prec \zeta$  the first term also vanishes.  $\square$

An immediate consequence of this lemma is the following: if  $\dim \zeta - \dim \xi \neq 1$  for cells  $\zeta, \xi \in \mathcal{X}$  then  $\pi_{\Gamma \setminus \Gamma_0}^\xi \circ \partial \equiv 0$  on  $C_*(\Gamma^\zeta \setminus \Gamma_0^\zeta)$ . We use this fact frequently in the following proposition.

**Proposition 4.9.** For each  $i \in \{1, \dots, k\}$ ,  $R(i - 1)$  and  $P(i - 1)$  imply  $P(i)$ .

*Proof.* For some  $i > 0$ , assume  $R(i - 1)$  and  $P(i - 1)$ . From  $R(i - 1)$  we know that the algorithm executes Step  $(i - 1)$ . Let  $\xi \in \mathcal{X}$  satisfy  $\dim \xi \neq j(i)$ . We show  $\mathbf{b}_i^\xi = 0$ . By the definition of  $\mathbf{b}_i^\xi$ , we need to show  $\pi_{\Gamma \setminus \Gamma_0}^\xi \circ \partial(\mathbf{g}_i) = 0$ . From the definition of  $\mathbf{g}_i$ , we obtain

$$\pi_{\Gamma \setminus \Gamma_0}^\xi \circ \partial(\mathbf{g}_i) = \pi_{\Gamma \setminus \Gamma_0}^\xi \circ \partial(\mathbf{g}_{i-1}) - \sum_{\dim \zeta = j(i) + 1} \pi_{\Gamma \setminus \Gamma_0}^\xi \circ \partial(\mathbf{c}_{i-1}^\zeta). \quad (9)$$

To show  $P(i)$ , we must show that the right side of Equation 9 vanishes. To this end we consider two cases which exhaust the possibilities, and show in both cases that the right side of Equation 9 vanishes.

1.  $\dim \xi < j(i)$  or  $\dim \xi > j(i) + 1$ .

In this case,  $\pi_{\Gamma \setminus \Gamma_0}^\xi \circ \partial(\mathbf{g}_{i-1}) = 0$  by the inductive hypothesis  $P(i - 1)$  since  $\dim \xi \neq j(i - 1) = j(i) + 1$ . In addition, each summand  $\pi_{\Gamma \setminus \Gamma_0}^\xi \circ \partial(\mathbf{c}_{i-1}^\zeta)$  vanishes by Lemma 4.8, as  $\mathbf{c}_{i-1}^\zeta \in C_*(\Gamma^\zeta \setminus \Gamma_0^\zeta)$  and  $\dim \zeta - \dim \xi \neq 1$ .

2.  $\dim \xi = j(i) + 1$ .

By Lemma 4.8, each summand of the type  $\pi_{\Gamma \setminus \Gamma_0}^\xi \circ \partial(\mathbf{c}_{i-1}^\zeta)$  on the right side of (9) vanishes when  $\zeta \neq \xi$ . The single surviving term of the sum is  $\pi_{\Gamma \setminus \Gamma_0}^\xi \circ \partial(\mathbf{c}_{i-1}^\xi) = \partial^\xi(\mathbf{c}_{i-1}^\xi) = \mathbf{b}_{i-1}^\xi$ , where the last equality follows from  $R(i - 1)$  and the definition of  $\mathbf{c}_{i-1}^\xi$ .

Thus, we may rewrite the right hand side of (9) as:

$$\pi_{\Gamma \setminus \Gamma_0}^\xi \circ \partial(\mathbf{g}_{i-1}) - \mathbf{b}_{i-1}^\xi.$$

This vanishes by the definition of  $\mathbf{b}_{i-1}^\xi$ .

□

**Proposition 4.10.** *For each  $i \in \{0, \dots, k\}$ ,  $P(i)$  implies  $Q(i)$ .*

*Proof.* Let  $i \in \{0, \dots, k\}$  and assume  $P(i)$ . We show  $Q(i)$ . By  $P(i)$ , we know the algorithm must execute Step  $(i - 1)$ . Since the fibers  $\{\Gamma^\xi \setminus \Gamma_0^\xi \mid \xi \in \mathcal{X}\}$  partition the graph  $\Gamma \setminus \Gamma_0$ , we have

$$\mathfrak{d}(\mathbf{g}_i) = \sum_{\xi \in \mathcal{X}} \pi_{\Gamma \setminus \Gamma_0}^\xi \circ \mathfrak{d}(\mathbf{g}_i) = \sum_{\xi \in \mathcal{X}} \mathbf{b}_i^\xi.$$

By Proposition 4.9, only those  $\xi \in \mathcal{X}$  which satisfy  $\dim \xi = j(i)$  make a non-trivial contribution to this sum. Accordingly,

$$\mathfrak{d}(\mathbf{g}_i) = \sum_{\dim \xi = j(i)} \mathbf{b}_i^\xi,$$

and we have shown  $Q(i)$ . □

**Proposition 4.11.** *For each  $i \in \{0, \dots, k\}$ ,  $Q(i)$  implies  $R(i)$ .*

*Proof.* Let  $i \in \{0, 1, \dots, k\}$ . From  $Q(i)$  we know the algorithm must execute Step  $(i - 1)$ . Let  $\xi \in \mathcal{X}$  have dimension  $k(i)$ . We show  $\mathbf{b}_i^\xi \in B_{k-1}(\Gamma^\xi \setminus \Gamma_0^\xi)$ .

To this end, we first show  $\mathfrak{d}^\xi(\mathbf{b}_i^\xi) = 0$ , i.e., that  $\mathbf{b}_i^\xi$  is a cycle in the fiber of  $\xi$ . From  $Q(i)$  we may obtain

$$\mathfrak{d}^\xi \circ \mathfrak{d}(\mathbf{g}_i) = \sum_{\dim \zeta = j(i)} \mathfrak{d}^\xi(\mathbf{b}_i^\zeta).$$

The left side of this expression is zero, since  $\mathfrak{d}^\xi \circ \mathfrak{d} \equiv \pi_{\Gamma \setminus \Gamma_0}^\xi \circ \mathfrak{d} \circ \mathfrak{d}$  and  $\mathfrak{d}$  is a boundary operator. On the other hand, each summand on the right side equals  $\pi_{\Gamma \setminus \Gamma_0}^\xi \circ \mathfrak{d}(\mathbf{b}_i^\zeta)$  but by Lemma 4.8 this quantity is trivial when  $\xi \neq \zeta$ . Thus, the right side equals  $\mathfrak{d}^\xi(\mathbf{b}_i^\xi)$  and so  $\mathbf{b}_i^\xi$  is a cycle in the fiber of  $\xi$  as desired.

By Proposition 2.23, the homology group  $H_{k-1}(\Gamma^\xi \setminus \Gamma_0^\xi)$  is trivial whenever  $\xi \in \mathcal{X}_0$  or if  $\dim \xi \neq k - 1$ . In this case every cycle in the associated fiber  $\Gamma^\xi \setminus \Gamma_0^\xi$  is a boundary, so without loss of generality we may assume  $\xi \in \mathcal{X} \setminus \mathcal{X}_0$  and  $\dim \xi = k - 1$ , i.e., the case when  $i = 0$ . Note that  $\Gamma_0^\xi = \emptyset$ , so Proposition 2.22 applies and  $H_{k-1}(\Gamma^\xi \setminus \Gamma_0^\xi)$  possesses a unique non-trivial homology class, any representative of which is isomorphically projected to a representative of the unique non-trivial homology class of  $H_{k-1}(\{\xi\})$ . Thus, to show that the cycle  $\mathbf{b}_0^\xi$  is in fact a boundary in the fiber, it suffices to show that  $\pi_{\mathcal{X}}(\mathbf{b}_0^\xi) = 0$ .

By definition,  $\mathbf{b}_0^\xi = \pi_{\Gamma \setminus \Gamma_0}^\xi \circ \mathfrak{d}(\mathbf{g}_0)$  and by Proposition 4.3,  $\pi_{\mathcal{X}}(\mathbf{g}_0) = \mathbf{x}$ . Denote by  $p_\xi$  the canonical projection map from  $C_*(\mathcal{X} \setminus \mathcal{X}_0)$  to  $C_*(\{\xi\})$ :

$$p_\xi(\mathbf{c}) := \langle \xi, \mathbf{c} \rangle \xi.$$

It is straightforward to verify  $\pi_{\mathcal{X}} \circ \pi_{\Gamma \setminus \Gamma_0}^\xi \equiv p_\xi \circ \pi_{\mathcal{X}}$  on  $C_*(\Gamma \setminus \Gamma_0)$ . Thus,

$$\pi_{\mathcal{X}}(\mathbf{b}_0^\xi) = \pi_{\mathcal{X}} \circ \pi_{\Gamma \setminus \Gamma_0}^\xi \circ \mathfrak{d}(\mathbf{g}_0) = p_\xi \circ \pi_{\mathcal{X}} \circ \mathfrak{d}(\mathbf{g}_0).$$

Since  $\pi_{\mathcal{X}}$  is a chain map and  $\pi_{\mathcal{X}}(\mathbf{g}_0) = \mathbf{x}$ , we have  $\pi_{\mathcal{X}}(\mathbf{b}_0^\xi) = p_\xi \circ \partial^{\mathcal{X}} \circ \pi_{\mathcal{X}}(\mathbf{g}_0) = p_\xi \circ \partial^{\mathcal{X}}(\mathbf{x})$ . But since  $\mathbf{x}$  is a cycle by assumption, this expression equals zero as desired. □

### 4.3 Algorithm for Combinatorial Maps

The following algorithm computes the induced map on homology beginning with an acyclic combinatorial map pair.

#### Algorithm 4.12. Map Homology Algorithm

**given**  $(\mathcal{F}, \mathcal{F}_0): (\mathcal{X}, \mathcal{X}_0) \rightrightarrows (\mathcal{Y}, \mathcal{Y}_0)$   
 $\{\mathbf{x}_i\} \leftarrow$  Apply Algorithm 3.19 to  $(\mathcal{X} \setminus \mathcal{X}_0)$ .  
**for all** generators  $\mathbf{x}_i$  **do**  
     $\mathbf{g}_i \leftarrow$  Apply Algorithm 4.5 to  $\mathbf{x}_i, (\mathcal{F}, \mathcal{F}_0)$ .  
**end for**  
 $\{\gamma', \psi', \phi'\}, \{\mathbf{y}_i\} \leftarrow$  Apply Algorithm 3.19 to  $(\mathcal{Y} \setminus \mathcal{Y}_0)$ .  
Store the generators  $\{\psi'(\mathbf{y}_i)\}$  columnwise in a matrix  $G$ .

Store the cycles  $\{\psi' \circ \pi_y(\mathbf{g}_i)\}$  columnwise in a matrix  $H$ .  
Solve the matrix equation  $GF = H$  for  $F$ .  
**return**  $F$

Another possible means of computing map homology is to directly compute representative cycles of homology classes in the relative graph complex and project them to both domain and codomain. However, Algorithm 4.12, despite its added complexity, can offer significant advantages in both space and time over the direct computation of the homology of the relative graph complex. We discuss space and time complexities in the next section.

## 5 Complexity Analysis and Concluding Remarks

In this section we provide complexity estimates for Algorithms 3.6 and 4.12.

### 5.1 Complexity of Computing Morse Complexes

In order to provide parameters in terms of which we state the complexity bounds, we first measure the size of input complexes. Next, we make some assumptions about the complexity of primitive operations.

**Complexity Parameters.** In graph theory complexity, results are usually stated in terms of vertex count  $|V|$  and edge count  $|E|$ . Though it would be perfectly valid to replace  $|E|$  with  $|V|^2$ , the bounds would not be as good. Indeed, they would be very misleading when applied to graphs where the number of edges was nearly linear compared to the number of vertices. In a similar vein, we do not wish to state our complexity results solely in terms of the *cell count*  $\mathbf{n}$  of a complex  $(\mathcal{X}, \kappa)$ , but also in terms of what we call the *complex mass*:

$$\mathbf{m} := \text{card} \{(\sigma, \tau) \in \mathcal{X}^2 : \kappa(\sigma, \tau) \neq 0\}.$$

Complex mass tells us the number of pairs of cells which are related by the incidence numbers  $\kappa$ . There is a clear analogy between the cells of a complex and the vertices in a graph, as well as the incidence numbers in a complex and the edges of a graph. Accordingly, we will state complexity results in terms of the cell count  $\mathbf{n}$  and the complex mass  $\mathbf{m}$ .

To make the analogy between graphs and cell complexes even clearer define the *boundary mass* of a cell  $\xi$  to be

$$\mathbf{m}_\xi := \text{card} \{\zeta \in \mathcal{X} : \kappa(\xi, \zeta) \neq 0\},$$

and the *coboundary mass* to be

$$\tilde{\mathbf{m}}_\xi := \text{card} \{\zeta \in \mathcal{X} : \kappa(\zeta, \xi) \neq 0\}.$$

Boundary mass and coboundary mass of a cell are analogous to the number of out-edges and number of in-edges of a vertex in a directed graph, respectively. Considering this analogy, it is straightforward to see that we have

$$\sum_{\xi} \mathbf{m}_\xi = \sum_{\xi} \tilde{\mathbf{m}}_\xi = \mathbf{m}.$$

**Complexity Assumptions.** For what follows, we will consider a complex  $(\mathcal{X}, \kappa)$  with cell count  $\mathbf{n}$  and complex mass  $\mathbf{m}$  giving rise to a chain complex  $(\mathcal{C}, \partial)$ . We assume that the underlying ring  $\mathbf{R}$  is a finite field, so we can represent chain coefficients in  $O(1)$  space and perform ring arithmetic in  $O(1)$  time. We will assume we can represent any chain with  $\mathbf{t}$  terms in  $O(\mathbf{t})$  space, for a maximum of  $O(\mathbf{n})$  space. We will assume that the time complexity of computing  $\partial(\mathbf{c})$  for a chain  $\mathbf{c}$  is

$$O\left(\sum_{\langle \xi, \mathbf{c} \rangle \neq 0} \mathbf{m}_\xi\right),$$

and the space complexity is  $O(\mathbf{n})$ . (In fact, we could give a better space complexity bound, but it isn't necessary for the result we wish to give.) In particular this means that evaluating  $\partial$  on any chain can be done in  $O(\mathbf{m})$  time and  $O(\mathbf{n})$  space. Similarly, we assume we can compute the coboundary  $\partial^T$  of a chain  $\mathbf{c}$  in time bounds

$$O\left(\sum_{\langle \xi, \mathbf{c} \rangle \neq 0} \tilde{\mathbf{m}}_\xi\right).$$

Observe that these assumptions allow us to evaluate  $\partial$  or  $\partial^T$  on each distinct cell in the complex in succession in  $O(\mathbf{m})$  time.

**Proposition 5.1.** *The following complexity bounds hold:*

1. *Algorithm 3.6 applied to  $(\mathcal{X}, \kappa)$  executes in  $O(m)$  time and requires  $O(n)$  memory.*
2. *The chain equivalences  $\phi$  and  $\psi$  can be evaluated on any chain in  $O(m)$  time and  $O(n)$  space.*

*Proof.* We show (1). A straightforward implementation of Algorithm 3.6 keeps track of the number of boundaries of each cell – in particular, updating them upon excision. Initializing this structure requires computing the boundary of each cell, which takes  $O(\sum_{\xi} m_{\xi}) = O(m)$  time. The boundary-counting structure makes it fairly straightforward to manage to control structure of the algorithm – in particular, when a cell has its boundary count lowered to one, it and its boundary cell become candidates for a  $\mathcal{K} - \mathcal{Q}$  pair, to be placed in a queue for processing. We omit further details on how to implement the control structures, pointing out only that the bottleneck cost is updating the boundary-count structure upon excision. There are  $O(n)$  excisions which may take place, and each one requires the computation of the coboundary to know which cells need their “number of boundaries” count deducted. Since a cell may be excised at most once, we require  $O(\sum_{\xi} \tilde{m}_{\xi}) = O(m)$  time. Thus the algorithm is linear time in the number of non-zero incidence numbers, that is, it is  $O(m)$ . Now we show (2). Since  $i$ ,  $j$  and  $\partial$  can be computed in  $O(m)$  time, then for both  $\psi$  and  $\phi$  the burden is to establish that  $\gamma$  can be evaluated on an input in  $O(m)$  time. Inspection of Algorithm 3.12 provides this after the following observations: 1) the algorithm is based on a sequence of deformations based on the boundary of a cell, and (2) the deformations are distinct, so in the computations of the boundary each incidence number is used at most  $O(1)$  times. It follows that the time complexity is at most  $O(m)$ . Since at each stage in Algorithm 3.12 we only need to store a chain, the memory usage is  $O(n)$ . □

**Perfect Morse Complexes.** It is often the case, in fact, that when we apply discrete Morse theory we obtain a Morse complex that is *perfect*; it has zero mass. In other words, the Morse boundary  $\Delta$  is identically zero. If this happens, there is no algebra required to know the homology of the Morse complex; the Betti numbers are just the cell counts of each dimension. Another way to say this is that a perfect Morse complex does not possess non-trivial boundary chains.

**Proposition 5.2.** *Let  $(\mathcal{X}, \kappa)$  be a complex with cell count  $n$  and complex mass  $m$ . Suppose the application of discrete Morse theory yields  $k$  critical cells. Then computing the set of all homology generators in  $\mathcal{X}$  has worst-case time complexity  $O(km + k^3)$ . Moreover, if the Morse complex has  $\Delta = 0$ , then we can give an improved worst-case time complexity bound of only  $O(km)$ .*

*Proof.* The cost of creating the Morse complex is  $O(km)$ , since we need to determine  $\Delta$  on each of the  $k$  critical cells, each which takes worst-case  $O(m)$  time. The cost of computing the homology generators in the Morse complex (for the case of fields) boils down to the cost of some algebra – in particular Gaussian elimination suffices for the case of fields. That Gaussian elimination can be done in cubic time is well known – hence we get the  $O(k^3)$  term. If, on the other hand, the Morse complex is perfect, then no algebra is necessary and we may omit the  $O(k^3)$ . Once we obtain these homology generators, we lift them (and there are at most  $k$ ) to the original complex via the chain equivalence  $\phi$ . This again takes  $O(km)$  time, leaving us with the stated worst-case time complexity bound. □

**Worst-Case versus Empirically Observed Times.** Worst-case analyses can be extremely misleading in this context. Worst-case analysis of matrix algorithms on sparse matrices tend to assume we get maximal fill-in on matrix algebra problems, thus leading to cubic estimates. Worst-case analysis of the flow step ( $\gamma$  evaluation) in discrete Morse theory assumes a chain must be deformed throughout the entire complex before it becomes canonical. Indeed, these worst-case scenarios are unrealistic for many practical problems, for which we empirically observe  $O(m)$  time rather than  $O(km + k^3)$  time. It would seem in order to find refinements to the method or else prove that these empirical times are guaranteed would require breakthroughs with significant repercussions for other branches of mathematics, particularly graph theory and efficient matrix multiplication. Our point of view is that the reason we see better empirical times is because our examples are topologically much simpler than the pathological worst-cases. We have observed that discrete Morse theory is efficient at solving the simple, essentially linear part, and reduces the problem of computing homology to a small algebraic core. We have pursued this method because we have been motivated by practical performance on test problems [17], rather than by theoretical asymptotic estimates.

## 5.2 Complexity of Computing Relative Map Homology

Though we are unable to state we have improved the complexity of computing homology by applying discrete Morse theory, we can, at least for an important class of problems, assert that we have improved the complexity bounds on the homology of maps problem.

The time complexity of the relative map homology algorithm can be broken into two parts. The first is the computation of chain representatives of the homology generators  $H_*(\mathcal{X} \setminus \mathcal{X}_0)$  and  $H_*(\mathcal{Y} \setminus \mathcal{Y}_0)$ . Also, there is the need to express homology generators in a preferred basis.

The second part analyzes the relative combinatorial map  $(\mathcal{F}, \mathcal{F}_0) : (\mathcal{X}, \mathcal{X}_0) \rightarrow (\mathcal{Y}, \mathcal{Y}_0)$  – in particular, we must perform the *Cycle Lift Algorithm*, Algorithm 4.5.

We have already discussed the time complexity of the first part (the preferred basis part consists of algebra which may be done in the Morse complex via Gaussian Elimination, which was already considered). It turns out that for the class of maps arising from applications in [3], we can achieve linear time in the mass of the relative graph complex for the second part. Unfortunately, this would be absorbed into the first term. Yet the first part has, as already discussed, a very misleading worst-case behavior. For this reason we believe it is more appropriate to consider the worst case time for the first and second parts separately, since we do not see the worst-case behavior attained in practice for the first part.

**Proposition 5.3.** *If Algorithm 3.6 applied to every one the relative graph fibers yields a perfect Morse function, then Algorithm 4.5 can be executed in  $O(N)$  time, where  $N$  is the number of cells in the relative graph complex.*

*Proof.* By inspection, every step of Algorithm 4.5 can be done in  $O(N)$  time except possibly the step of finding preboundaries in the relative graph fibers. Using the discrete Morse theoretic preboundary formula, we have

$$\mathcal{P} = \phi \circ \mathcal{P}_{\mathcal{A}} \circ \psi - \gamma.$$

Since the Morse complexes for the fibers are perfect, the first term drops away. This is because there are no non-trivial boundary chains in a perfect Morse complex, so  $\psi(\mathbf{b}) = 0$  for all boundaries  $\mathbf{b}$ . We are left with  $-\gamma$  furnishing the solution directly. Since  $\gamma$  executes in time linear to the mass of the relative fiber, and the sum of the masses of the relative fibers is bounded by the mass of the relative graph complex, the result follows.  $\square$

To analyze the space complexity of the homology of maps algorithm, we again break it into two parts and ignore the first for largely the same reasons. Thus we are only concerned with the space required to execute Algorithm 4.5. The space usage comes in the form of storing the chain  $\mathbf{g}_i$  at each stage, as well as represent the fiber complexes in which we must perform the preboundary-finding step. Since there are  $O(N)$  different terms possible in  $\mathbf{g}_i$ , it appears difficult to obtain a worst-case upper bound any better than  $O(N)$ . Yet, intuitively, we'd expect that most cycles occupy only a small fraction of fibers, and within the fibers only a small fraction of cells. This intuition is what motivated the fiber-wise algorithm, as we felt it would be quite space-efficient in practice. Indeed, a brute-force approach which computes the homology of the relative graph which is represented in memory has a lower bound space usage of  $\omega(N)$ ; whereas the minimum requirements of Algorithm 4.5 depend on the size of the domain and codomain, not the size of the graph. We consider this a significant advantage, since in many important problems the size of the chains  $\mathbf{g}_i$  are not dominant.

## 5.3 Concluding Remarks

We have given a presentation of discrete Morse theory geared towards using it in algorithms. We think it is difficult, if not impossible without some breakthrough, to improve on worst-case time bounds for the core algebraic algorithms at the heart of homology computations. Accordingly, we have not attempted to do this. Instead, we have focussed on discrete Morse theory – which might be considered as “geometrically inspired algebra” – as a preprocessing step to attempt to efficiently reduce the size of the problem. This approach has met with success computationally [17].

We have given a new algorithm for the homology of maps. Crucially, this algorithm is based on a fiber-wise approach on the graph and the primary step asks us to compute a preboundary. We have discussed how discrete Morse theory may be used to find preboundaries, and in particular, that this may be done in linear time for an important class of examples which arise in practice. Because of this, we can present for the first time a map homology algorithm capable of addressing Conley Index computations as in [3] which has a linear time dependence on the size complexity of the mapping itself. We expect, though our upper

bounds on worst-case space usage are too weak to prove this claim, that our fiber-based treatment of the map homology problem leads to space savings.

**Acknowledgements.** The work of S.H., K.M., and V.N. was partially supported by NSF grants DMS-0915019, DMS-1125174, and CBI-0835621 and by contracts from DARPA and AFOSR. The work of M.M. was partially supported by NCN of Poland, grant N N201 419639.

## References

- [1] M. Allili and T. Kaczynski. An algorithmic approach to the construction of homomorphisms induced by maps in homology. *Trans. Amer. Math. Soc.*, 352(5):2261–2281, 2000.
- [2] M. Allili and T. Kaczynski. Geometric construction of a coboundary of a cycle. *Discrete Comput. Geom.*, 25(1):125–140, 2001.
- [3] Z. Arai, W. Kalies, H. Kokubu, K. Mischaikow, H. Oka, and P. Pilarczyk. A database schema for the analysis of global dynamics of multiparameter systems. *SIAM J. Applied Dyn Syst*, 8:757–789, 2009.
- [4] CAPD::RedHom. <http://redhom.ii.uj.edu.pl>.
- [5] G. Carlsson. Topology and data. *Bull. Amer. Math. Soc. (N.S.)*, 46(2):255–308, 2009.
- [6] M. K. Chari. On discrete Morse functions and combinatorial decompositions. *Discrete Math.*, 217(1-3):101–113, 2000. Formal power series and algebraic combinatorics (Vienna, 1997).
- [7] CHomP. <http://chomp.rutgers.edu>.
- [8] C. J. A. Delfinado and H. Edelsbrunner. An incremental algorithm for Betti numbers of simplicial complexes on the 3-sphere. *Comput. Aided Geom. Design*, 12(7):771–784, 1995. Grid generation, finite elements, and geometric design.
- [9] P. Dłotko, R. Ghrist, M. Juda, and M. Mrozek. Distributed computation of coverage in sensor networks by homological methods. *Applicable Algebra in Engineering, Communication and Computing*, 23:29–58, 2012.
- [10] P. Dłotko, T. Kaczynski, M. Mrozek, and T. Wanner. Coreduction homology algorithm for regular CW-complexes. *Discrete and Computational Geometry*, 46:361–388, 2011.
- [11] J-G Dumas, F. Heckenbach, D. Saunders, and V. Welker. Computing simplicial homology based on efficient smith normal form algorithms. In *M. Joswig and N. Takayama, editors, Algebra, Geometry, and Software Systems*, pages 177–206, 2003.
- [12] H. Edelsbrunner and J. Harer. Persistent homology—a survey. In *Surveys on discrete and computational geometry*, volume 453 of *Contemp. Math.*, pages 257–282. Amer. Math. Soc., Providence, RI, 2008.
- [13] H. Edelsbrunner and J. L. Harer. *Computational topology*. American Mathematical Society, Providence, RI, 2010. An introduction.
- [14] R. Forman. Morse theory for cell complexes. *Advances in Mathematics*, 134:90–145, 1998.
- [15] R. Ghrist. Barcodes: the persistent topology of data. *Bull. Amer. Math. Soc. (N.S.)*, 45(1):61–75, 2008.
- [16] R. Ghrist. Three examples of applied and computational homology. *Nieuw Arch. Wiskd. (5)*, 9(2):122–125, 2008.
- [17] S. Harker, K. Mischaikow, M. Mrozek, V. Nanda, H. Wagner, M. Juda, and P. Dłotko. The efficiency of a homology algorithm based on discrete Morse theory and coreductions. *Proceedings of the 3rd International Workshop on Computational Topology in Image Context*, 1:41–47, 2010.
- [18] Allen Hatcher. *Algebraic Topology*. Cambridge University Press, 2002.
- [19] T. Kaczynski, K. Mischaikow, and M. Mrozek. Computing homology. *Homology Homotopy Appl.*, 5(2):233–256, 2003. Algebraic topological methods in computer science (Stanford, CA, 2001).
- [20] T. Kaczynski, K. Mischaikow, and M. Mrozek. *Computational Homology. Applied Mathematical Sciences 157*. Springer-Verlag, 2004.
- [21] T. Kaczynski, M. Mrozek, and M. Ślusarek. Homology computation by reduction of chain complexes. *Computers & Mathematics with Applications*, 35(4):59–70, 1998.
- [22] W. D. Kalies, K. Mischaikow, and G. Watson. Cubical approximation and computation of homology. In *Conley index theory (Warsaw, 1997)*, volume 47 of *Banach Center Publ.*, pages 115–131. Polish Acad. Sci., Warsaw, 1999.

- [23] D. Kozlov. *Combinatorial Algebraic Topology*, volume 21 of *Algorithms and Computation in Mathematics*. Springer, 2008.
- [24] S. Lefschetz. *Algebraic Topology*. American Mathematical Society Colloquium Publications, v. 27. American Mathematical Society, New York, 1942.
- [25] K. Mischaikow, M. Mrozek, and P. Pilarczyk. Graph approach to the computation of the homology of continuous maps. *Found. Comput. Math.*, 5(2):199–229, 2005.
- [26] M. Mrozek and B. Batko. The coreduction homology algorithm. *Discrete and Computational Geometry*, 41:96–118, 2009.
- [27] M. Mrozek, P. Pilarczyk, and N. Żelazna. Homology algorithm based on acyclic subspace. *Computers and Mathematics with Applications*, 55:2395–2412, 2008.
- [28] M. Mrozek and T. Wanner. Coreduction homology algorithm for inclusions and persistent homology. *Computers and Mathematics with Applications*, 60(10):2812–2833, 2010.
- [29] M. Mrozek, M. Żelawski, A. Gryglewski, S. Han, and A. Krajniak. Homological methods for extraction and analysis of linear features in multidimensional images. *Pattern Recognition*, 45:285–298, 2012.
- [30] Perseus. <http://www.math.rutgers.edu/~vidit/perseus.html>.
- [31] B. D. Saunders and Z. Wan. Smith normal form of dense integer matrices, fast algorithms into practice. *Internat. Symp. Symbolic Algebraic Comput.*, pages 274–281, 2004.
- [32] E. H. Spanier. *Algebraic topology*. McGraw-Hill Book Co., New York, 1966.
- [33] A. W. Tucker. Cell spaces. *Ann. of Math. (2)*, 37(1):92–100, 1936.