

Extending the Notion of AT-Model for Integer Homology Computation*

Rocio Gonzalez-Diaz, María José Jiménez, Belén Medrano**, and Pedro Real

Applied Math Department, University of Sevilla, Seville, Spain
{rogodi,majiro,belenmg,real}@us.es
<http://alojamientos.us.es/gtocoma>

Abstract. When the ground ring is a field, the notion of algebraic topological model (AT-model) is a useful tool for computing (co)homology, representative (co)cycles of (co)homology generators and the cup product on cohomology of nD digital images as well as for controlling topological information when the image suffers local changes [6,7,9]. In this paper, we formalize the notion of λ -AT-model (λ being an integer) which extends the one of AT-model and allows the computation of homological information in the integer domain without computing the Smith Normal Form of the boundary matrices. We present an algorithm for computing such a model, obtaining Betti numbers, the prime numbers p involved in the invariant factors (corresponding to the torsion subgroup of the homology), the amount of invariant factors that are a power of p and a set of representative cycles of the generators of homology mod p , for such p .

1 Introduction

There are many tasks in Vision and Image Processing that involve computing certain topological characteristics of objects in a given image such as, for example, connectivity and the number of holes and cavities. We focus here on homology groups (connectivity and the number of holes and cavities can be obtained from them), which are known to be computable in finite dimensions. The classical algorithm for computing integer homology is based on performing row and column operations on the boundary matrices in order to reduce them to the Smith Normal Form (SNF). The integer homology groups can be then determined from this canonical form (see, for example, [13]). However, explicit examples can be given for which this algorithm has a worst-case computational complexity which grows exponentially in both space and time [4].

Our aim is the computation of integer homology information avoiding the computation of the SNF in the integer domain. In fact, our approach allows the computation of Betti numbers, the prime numbers p involved in the invariant

* Partially supported by Junta de Andalucía (FQM-296 and TIC-02268) and Spanish Ministry for Science and Education (MTM-2006-03722).

** Fellow associated to University of Seville under a Junta de Andalucía research grant.

factors (corresponding to the torsion subgroup of the homology), the amount of invariant factors that are a power of p and a set of “moduli” representative cycles of the generators of homology in polynomial time. Moreover, our method is not only valid for simplicial complexes but also for other combinatorial objects such as cubical complexes or simplicial complexes since we deal with the group structures.

In the first part of the paper, we recall classical definitions from Algebraic Topology. We also present previous tools for computing topological information: AT-models and AM-models, and we recall the main properties of these structures. Furthermore, we define the notion of λ -AT-model, study its properties, give an algorithm for computing it and study its complexity. Finally, we describe how to obtain homology information in the integer domain from a λ -AT-model. The last section is devoted to conclusions and future works.

2 Definitions and Prior Work

This section introduces the background needed throughout the paper which is essentially extracted from Munkres’ book [13]. We also recall briefly the concepts of AT-model and AM-model and their properties.

A *chain complex* \mathcal{C} is a sequence $\{C_q, d_q\}$ of abelian groups C_q and homomorphisms $d_q : C_{q+1} \rightarrow C_q$,

$$\dots \xrightarrow{d_3} C_2 \xrightarrow{d_2} C_1 \xrightarrow{d_1} C_0 \xrightarrow{d_0} 0,$$

such that, for all q , $d_q d_{q+1} = 0$. The set of all the homomorphisms d_q ($q \geq 0$) is called the *differential* of \mathcal{C} . The chain complex \mathcal{C} is *free* if C_q is a free abelian group for each q . It is *finite* if there exists an integer $n > 0$ such that $C_q = 0$ for $q > n$ and each abelian group C_q is finitely generated. In this case, if $C_n \neq 0$, we say that \dim of \mathcal{C} is n , and then, \mathcal{C} can be encoded as a pair (C, d) , where $C = \bigcup_{q=0}^n B_q$, being B_q a basis of C_q and d the matrix corresponding to the differential of \mathcal{C} with respect to the basis C . Suppose that $B_q = \{a_1, \dots, a_{m_q}\}$. A q -chain $a \in \mathcal{C}$ is a formal sum of elements of B_q , $a = \sum_{i=1}^{m_q} \lambda_i a_i$, where $\lambda_i \in \mathbf{Z}$ and $a_i \in C_q$. In this case, $\dim a = q$ and $c_a(a_i)$ denotes the coefficient λ_i .

Since our goal is the computation of homology information of “finite” objects (for example, objects explicitly represented within a computer), all chain complexes are finite and free.

Example 1. Shapes are classically modelled with a cellular subdivision. Several combinatorial structures may represent such subdivision. Simplicial complexes have proven to be a useful tool to model a geometric object. Roughly speaking, they are collections of simplices (convex hulls of a set of affinely independent points) that fit together in a natural way to form the object. For every simplicial complex K , one can define a chain complex $C(K)$ canonically associated to it. The homology of K is then defined as the homology of $C(K)$ [13]. Another way to extract combinatorial information from a geometric structure arising naturally, for example, from tomography, numerical computations and graphics,

is by means of cubical grids, which subdivide the space into cubes with vertices in an integer lattice. This approach, that can be generalized to an arbitrary dimension, is a cubical complex. The homology of a given cubical complex is the homology of the cubical chain complex associated to it [11]. Finally, simplicial sets [1] include simplicial complexes and cubical complexes as particular cases. They can be used for representing “hybrid” grids coming from finite element methods. In [15], a free chain complex is associated to a simplicial set and the homology of the simplicial set is defined as the homology of the associated chain complex.

We base all formulas and algorithms in this paper on an ordered basis of the chain complex \mathcal{C} where each prefix of the ordering contains the basis of a subcomplex. We call such an ordering a *filter*. In other words, given a chain complex \mathcal{C} , $C = \{a_1, \dots, a_m\}$ is a filter if it is a basis of the chain complex \mathcal{C} and for each j (where $1 \leq j \leq m$), $C = \{a_1, \dots, a_j\}$ is a basis of a subcomplex of \mathcal{C} . For instance, given a chain complex $\mathcal{D} = (D, d)$, a reordering $D' = \{c'_1, \dots, c'_m\}$ of D such that $\dim c'_i \leq \dim c'_j$ when $i < j$, is always a filter of \mathcal{D} .

Example 2. Consider the simplicial complex S derived from the triangulation of the Klein bottle given in Figure 1 and the chain complex $\mathcal{C}(S)$ associated to S . Then,

$$C(S) = \{ a, d, ad, f, af, df, adf, b, ab, bf, abf, c, bc, cf, bcf, g, cg, fg, cfg, ac, ag, acg, e, ae, eg, aeg, de, ef, def, h, fh, eh, efh, gh, fgh, i, gi, hi, ghi, ei, egi, di, dei, ah, aeh, bh, abh, bi, bhi, ci, bci, ai, aci, adi \},$$

where $v_0 \cdots v_n$ denotes the simplex spanned by the vertices v_0, \dots, v_n , is a filter of $\mathcal{C}(S)$.

The chain a is a q -cycle if $a \in \text{Ker } d_q$. If $a \in \text{Im } d_{q+1}$ then a is called a q -boundary. Denote the groups of q -cycles and q -boundaries by Z_q and B_q respectively. Define the integer q th homology group to be the quotient group Z_q/B_q , denoted

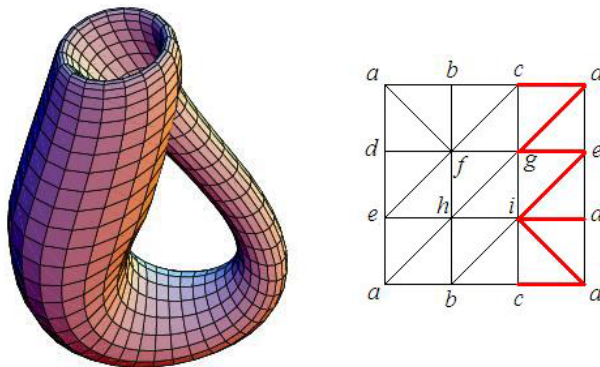


Fig. 1. The Klein bottle and a triangulation of it

by $H_q(\mathcal{C}; \mathbf{Z})$. We say that a is a *representative q -cycle* of the homology generator $a + B_q$ (denoted by $[a]$). For each q , the integer q th homology group $H_q(\mathcal{C}; \mathbf{Z})$ is a finitely generated abelian group. Then $H_q(\mathcal{C}; \mathbf{Z})$ is isomorphic to $F_q \oplus T_q$ where

$$F_q = \mathbf{Z} \oplus \cdots \oplus \mathbf{Z} \quad \text{and} \quad T_q = (\mathbf{Z}/\alpha_{(q,1)}) \oplus \cdots \oplus (\mathbf{Z}/\alpha_{(q,s)})$$

are the *free subgroup* and the *torsion subgroup* of $H_q(\mathcal{C}; \mathbf{Z})$, respectively. The rank of F_q , denoted by β_q , is called the q th *Betti number* of \mathcal{C} . Each $\alpha_{(q,i)}$ is a power of a prime, $\alpha_{(q,i)} = p_i^{t_{(q,p_i)}}$. They are called the *invariant factors* of $H_q(\mathcal{C}; \mathbf{Z})$. The numbers β_q and $\alpha_{(q,i)}$ are uniquely determined by $H_q(\mathcal{C}; \mathbf{Z})$ (up to a rearrangement). Therefore, this representation is in some sense a “canonical form” for $H_q(\mathcal{C}; \mathbf{Z})$.

The q th homology group of \mathcal{C} with coefficients in \mathbf{Z}/p for p a prime, denoted by $H_q(\mathcal{C}; \mathbf{Z}/p)$, is a vector space. Its rank, denoted by $\beta_{(q,p)}$, depends on the prime p . Universal Coefficient Theorem for Homology [13, p. 332] implies that for each prime p ,

$$T_{(0,p)} = \beta_{(0,p)} - \beta_0 \quad \text{and} \quad T_{(q,p)} = \beta_{(q,p)} - \beta_q - T_{(q-1,p)} \quad \text{for } q > 0;$$

where $T_{(i,p)}$ is the number of invariant factors of $H_i(\mathcal{C}; \mathbf{Z})$ that are a power of p .

Let $\mathcal{C} = \{C_q, d_q\}$ and $\mathcal{C}' = \{C'_q, d'_q\}$ be two chain complexes. A *chain map* $f : \mathcal{C} \rightarrow \mathcal{C}'$ is a family of homomorphisms $\{f_q : C_q \rightarrow C'_q\}$ such that $d'_q f_q = f_{q-1} d_q$ for all $q \geq 0$. A chain map $f : \mathcal{C} \rightarrow \mathcal{C}'$ induces a homomorphism $f_* : H(\mathcal{C}; \mathbf{Z}) \rightarrow H(\mathcal{C}'; \mathbf{Z})$ where $f_*[a] = [f(a)]$ for $[a] \in H(\mathcal{C}; \mathbf{Z})$. If $f, g : \mathcal{C} \rightarrow \mathcal{C}'$ are chain maps, then a *chain homotopy* $\phi : \mathcal{C} \rightarrow \mathcal{C}'$ of f to g is a family of homomorphisms $\{\phi_q : C_q \rightarrow C'_{q+1}\}$ such that $f_q - g_q = d'_{q+1} \phi_q + \phi_{q-1} d_q$.

A *chain contraction* of a chain complex \mathcal{C} to another chain complex \mathcal{C}' is a set of three homomorphisms (f, g, ϕ) such that: $f : \mathcal{C} \rightarrow \mathcal{C}'$ and $g : \mathcal{C}' \rightarrow \mathcal{C}$ are chain maps; fg is the identity map of \mathcal{C}' and $\phi : \mathcal{C} \rightarrow \mathcal{C}$ is a chain homotopy of the identity map of \mathcal{C} to gf , that is, $\phi d + d \phi = id_{\mathcal{C}} - gf$. Important properties of chain contractions are: \mathcal{C}' has fewer or the same number of generators than \mathcal{C} ; and \mathcal{C} and \mathcal{C}' have isomorphic homology groups.

An *AT-model* [6,7,9] for a chain complex $\mathcal{C} = (C, d)$ is a chain contraction of \mathcal{C} to a chain complex \mathcal{H} with null differential. An AT-model can be stored as a set $((C, d), H, f, g, \phi)$, where C and H are basis of \mathcal{C} and \mathcal{H} , and f, g and ϕ are the matrices corresponding to the homomorphisms that defines the chain contraction of \mathcal{C} to \mathcal{H} . Observe that the homology of \mathcal{C} is isomorphic to \mathcal{H} . If the ground ring is \mathbf{Z}/p , being p a prime, the following algorithm computes an AT-model for a given chain complex. This algorithm is a straightforward modification of that in [9].

Algorithm 1. *Computing an AT-model for a chain complex \mathcal{C} over \mathbf{Z}/p .*

INPUT: a filter $C = \{a_0, \dots, a_m\}$ of the chain complex \mathcal{C} ,
 and the matrix of the differential d for the basis C .

$H := \{\}$, $f := 0$, $g := 0$, $\phi := 0$.

```

For  $i = 0$  to  $m$  do
  If  $fd(a_i) = 0$ , then
     $H := H \cup \{a_i\}$ ,  $f(a_i) := a_i$ ,  $\phi(a_i) := 0$ ,  $g(a_i) := a_i - \phi d(a_i)$ .
  If  $fd(a_i) \neq 0$ , then:
     $\mu_i := \min \{c_{fd(a_i)}(a_j), j = 0, \dots, i - 1\}$ ,
     $k := \max \{j \text{ such that } c_{fd(a_i)}(a_j) = \mu_i, j = 0, \dots, i - 1\}$ ,
     $H := H \setminus \{a_k\}$ ,  $f(a_i) := 0$ ,  $\phi(a_i) := 0$ .
    For  $j = 0$  to  $i - 1$  do,
       $\lambda_{a_j} := c_{f(a_j)}(a_k)$ ,
       $f(a_j) := f(a_j) - \mu_i^{-1} \lambda_{a_j} fd(a_i)$ ,
       $\phi(a_j) := \phi(a_j) + \mu_i^{-1} \lambda_{a_j} (a_i - \phi d(a_i))$ .
OUTPUT: the set  $((C, d), H, f, g, \phi)$ .

```

The key idea of this algorithm is the same as in [3]: in the i th step, the element a_i of the filter C is added and then, a homology class is created or destroyed. The algorithm runs in time at most $\mathcal{O}(m^3)$, where m is the number of elements of C . Recall that the notion of AT-model is an useful tool for computing (co)homology, representative cycles of (co)homology generators and the cup product on cohomology of nD digital images as well as for controlling topological information when the image suffers local changes [6,7,9]. The main problem of the computation of AT-models over \mathbf{Z}/p is that if the object under study contains torsion, then the Betti numbers $\beta_{(q,p)}$ can change when p varies.

Example 3. The Betti numbers of the simplicial complex S (see Figure 1) computed over the field \mathbf{Z}/p , for $p = 2, 3, 29$.

	$\beta_{(0,p)}$	$\beta_{(1,p)}$	$\beta_{(2,p)}$
$\mathbf{Z}/2$	1	2	2
$\mathbf{Z}/3$	1	1	0
$\mathbf{Z}/29$	1	1	0

An AM-model [5] for a chain complex $\mathcal{C} = (C, d)$ is a chain contraction (f, g, ϕ) of \mathcal{C} to $\mathcal{M} = (M, d')$ such that the matrix A of the differential d' coincides with its Smith normal form and satisfies that any non-null entry of A is greater than 1. Working with coefficients in the integer domain, an AM-model for \mathcal{C} can always be computed. Moreover, the integer (co)homology of \mathcal{C} and representative cycles of (co)homology generators can be directly obtained from \mathcal{M} [8].

The algorithm for computing AM-models given in [8] needs to reduce the matrix of the differential to its Smith Normal Form (SNF). Explicit examples can be given for which the computation of SNF has a worst-case computational complexity which grows exponentially in both space and time [4]. Many algorithms have been devised to improve this complexity bound [10,17,2,14].

Our aim in this paper is the computation of integer homology information avoiding the computation of the SNF in the integer domain.

3 Extending the Notion of Algebraic Topological Model

In this section we will consider that the ground ring is \mathbf{Z} . We first define the notion of λ -AT-model which is a generalization of the one of AT-model. We study its properties, give an algorithm for computing it and study its complexity.

Definition 1. Let $\mathcal{C} = (C, d)$ be a chain complex, λ a non-null integer and $\mathcal{H} = (H, d')$ a chain complex with null differential (that is, $d' = 0$). Let $f : C \rightarrow H$, $g : H \rightarrow C$ and $\phi : C \rightarrow C$ be three homomorphisms. Then the set $((C, d), H, f, g, \phi, \lambda)$ is a λ -AT-model if f and g are chain maps, $fg = \lambda \cdot id_{\mathcal{H}}$ and ϕ is a chain homotopy of $\lambda \cdot id_{\mathcal{C}}$ to gf , that is, $\lambda \cdot id_{\mathcal{C}} - gf = \phi d + d\phi$.

Proposition 1. Given a λ -AT-model, a rational AT-model (i.e., an AT-model over \mathbf{Q}) can directly be obtained as well as rational (co)homology and representative cycles of (co)homology generators. Concretely, if $((C, d), H, f, g, \phi, \lambda)$ is a λ -AT-model for a chain complex \mathcal{C} , then $((C, d), H, \frac{1}{\lambda}f, g, \frac{1}{\lambda}\phi)$ is an AT-model for \mathcal{C} over \mathbf{Q} and $\{g(h) : h \in H\}$ is a set of representative cycles of the generators of $H(\mathcal{C}; \mathbf{Q})$.

Corollary 1. Let $((C, d), H, f, g, \phi, \lambda)$ be a λ -AT-model. Then \mathcal{H} (the chain complex generated by H with null differential) is isomorphic to the free subgroup of $H(\mathcal{C}; \mathbf{Z})$. Moreover, the set $\{g(h) : h \in H\}$ is a set of independent non-boundary cycles of \mathcal{C} over \mathbf{Z} .

Proposition 2. Given a λ -AT model $((C, d), H, f, g, \phi, \lambda)$ and a prime p such that p does not divide λ , then $((C, d_p), H, f_p, g_p, \phi_p)$, where $d_p = d \bmod p$, $f_p = \lambda^{-1}f \bmod p$, $g_p = g \bmod p$ and $\phi_p = \lambda^{-1}\phi \bmod p$, is an AT-model for \mathcal{C} over \mathbf{Z}/p and $\{g_p(h) : h \in H\}$ is a set of representative cycles of the generators of $H(\mathcal{C}; \mathbf{Z}/p)$.

Proposition 3. Let $((C, d), H, f, g, \phi, \lambda)$ be a λ -AT-model. Let $a \in \mathcal{C}$ such that $d(a) = 0$. If there exists $b \in \mathcal{C}$ such that $d(b) = \alpha a$ where $\alpha \in \mathbf{Z}$ and $\alpha \neq 0$, then $f(a) = 0$.

Proof. Suppose that $b \in \mathcal{C}$ such that $d(b) = \alpha a$ where $\alpha \in \mathbf{Z}$ and $\alpha \neq 0$, and $f(a) \neq 0$. Then $\alpha f(a) \neq 0$ (since the ground ring is \mathbf{Z}). On the other hand, $\alpha f(a) = f(\alpha a) = f(d(b)) = 0$, a contradiction. \square

Corollary 2. Let $((C, d), H, f, g, \phi, \lambda)$ be a λ -AT-model. Let $a \in \mathcal{C}$ such that $d(a) = 0$. If there exists $b \in \mathcal{C}$ such that $d(b) = \alpha a$ where $\alpha \in \mathbf{Z}$, $\alpha \neq 0$, and for each β , where $0 < \beta < \alpha$, $\beta a \notin \text{Im } d$. then α divides λ .

Proof. By Proposition 3, we have $f(a) = 0$. Since $\lambda a - gf(a) = \phi d(a) + d\phi(a)$, then $\lambda a = d\phi(a)$. Suppose that α does not divide λ . There exists $c, r \in \mathbf{Z}$, such that $0 < r < \alpha$ and $\lambda = c\alpha + r$. On one hand, $ra \notin \text{Im } d$; on the other hand, $ra = (\lambda - c\beta)a = d(\phi(a) - cb) \in \text{Im } d$, a contradiction. We conclude that α divides λ . \square

Corollary 3. *Let $((C, d), H, f, g, \phi, \lambda)$ be a λ -AT-model. If $\alpha = p^{t(a,p)}$ is an invariant factor of $H_q(C; \mathbf{Z})$, then p divides λ .*

Algorithm 2. *Algorithm for computing a λ -AT-model for a chain complex C .*

INPUT: a filter $C = \{a_0, \dots, a_m\}$ of the chain complex C ,
 and the matrix of the differential d for the basis C .

$H := \{ \}, \lambda := 1, f := 0, g := 0, \phi := 0$.

For $i = 0$ to m do

 If $fd(a_i) = 0$, then

$H := H \cup \{a_i\}, f(a_i) := a_i, \phi(a_i) := 0, g(a_i) := \lambda a_i - \phi d(a_i)$.

 If $fd(a_i) \neq 0$, then

$\mu_i := \min \{|c_{fd(a_i)}(a_j)|, j = 0, \dots, i - 1\}$,

$k := \max \{j \text{ such that } |c_{fd(a_i)}(a_j)| = \mu_i, j = 0, \dots, i - 1\}$,

$\lambda_k := c_{fd(a_i)}(a_k)$,

$H := H \setminus \{a_k\}, f(a_i) := 0, \phi(a_i) := 0$.

 For $j = 0$ to $i - 1$,

$\lambda_{a_j} := c_{f(a_j)}(a_k)$,

$f(a_j) := \lambda_k f(a_j) - \lambda_{a_j} fd(a_i)$,

$\phi(a_j) := \lambda_k \phi(a_j) + \lambda_{a_j} (\lambda a_i - \phi d(a_i))$,

$\lambda := \lambda \lambda_k$.

OUTPUT: The set $((C, d), H, f, g, \phi, \lambda)$.

Theorem 3. *The set $((C, d), H, f, g, \phi, \lambda)$ obtained applying Algorithm 2 defines a λ -AT-model for the chain complex $C = (C, d)$.*

Proof. Assume that $((C_{i-1}, d), H_{i-1}, f_{i-1}, g_{i-1}, \phi_{i-1}, \lambda_{i-1})$ is the λ_{i-1} -AT-model obtained using the algorithm above for the filter $C_{i-1} = \{a_0, \dots, a_{i-1}\}$. Assume that the annihilation properties $f_{i-1}\phi_{i-1} = 0, \phi_{i-1}g_{i-1} = 0$ and $\phi_{i-1}\phi_{i-1} = 0$ hold. We will prove that the set $(C_i, d, f_i, g_i, \phi_i, \lambda_i)$ obtained after adding a_i to the filter C_{i-1} is a λ_i -AT-model. More concretely, we will prove that $f_i d = 0, dg_i = 0, f_i g_i = \lambda_i \cdot id, \lambda_i \cdot id - g_i f_i = \phi_i d + d\phi_i, f_i \phi_i = 0, \phi_i g_i = 0$ and $\phi_i \phi_i = 0$. We deal only with the case $f_{i-1}d(a_i) \neq 0$; the other case is left to the reader. First, $f_i d(a_i) = \lambda_k f_{i-1}d(a_i) - \lambda_k f_{i-1}d(a_i) = 0$. Second, $\phi_i d(a_i) + d\phi_i(a_i) = \lambda_k \phi_{i-1}d(a_i) + \lambda_k (\lambda_{i-1} a_i - \phi_{i-1}d(a_i)) = \lambda_i a_i = \lambda_i a_i - g_i f_i(a_i)$. Finally, it is clear that $f_i \phi_i(a_i) = 0$ and $\phi_i \phi_i(a_i) = 0$. Now, let $a_j \in C_{i-1}$, then $f_i d(a_j) = \lambda_k f_{i-1}d(a_j)$ which is null by induction; $\phi_i d(a_i) + d\phi_i(a_i) = \lambda_k \phi_{i-1}d(a_j) + \lambda_{d(a_j)} (\lambda_{i-1} a_i - \phi_{i-1}d(a_i)) + \lambda_k d\phi_{i-1}(a_j) + \lambda_{a_j} (\lambda_{i-1} d(a_i) - d\phi_{i-1}d(a_i)) = \lambda_k (\lambda_{i-1} a_j - g_{i-1} f_{i-1}(a_j)) + \lambda_{a_j} g_{i-1} f_{i-1}d(a_i) = \lambda_i a_j - g_i f_i(a_j)$. Moreover, $f_i \phi_i(a_j) = f_i (\lambda_k \phi_{i-1}(a_j) + \lambda_{a_j} (\lambda_{i-1} a_i - \phi_{i-1}d(a_i))) = 0; \phi_i \phi_i(a_j) = \phi_i (\lambda_k \phi_{i-1}(a_j) + \lambda_{a_j} (\lambda_{i-1} a_i - \phi_{i-1}d(a_i))) = 0$. If $a_j \in H_i$, then $dg_i(a_j) = dg_{i-1}(a_j) = 0$, by induction; $f_i g_i(a_j) = \lambda_k f_{i-1} g_{i-1}(a_j) - \lambda_{g_{i-1}(a_j)} f_{i-1} d(a_j) = \lambda_k \lambda_{i-1} a_j = \lambda_i a_j$. Finally, it is easy to see that $\phi_i g_i(a_j) = 0$. \square

To study the complexity, fix the dimension of the complex, n , and count the number of elementary operations involved in the algorithm. In the i th step, we have to evaluate $f_{i-1}d(a_i)$. The numbers of elements of C involved in $d(a_i)$

and $f_{i-1}(a_j)$ for $1 \leq j < i$ is at most n and m , respectively. Therefore, the evaluation of $f_{i-1}d(a_i)$ costs $O(nm) = O(m)$. If $f_{i-1}d(a_i) \neq 0$, we have to update $f_{i-1}(a_j)$ and $\phi_{i-1}(a_j)$ for $1 \leq j < i$. The total cost of these operations is $O(m^2)$. Therefore, the total algorithm runs in time at most $O(m^3)$.

The following proposition shows that AT-models over \mathbf{Z}/p , p being any prime, can also be computed using Algorithm 2.

Proposition 4. *If the output of Algorithm 2, working with coefficients in \mathbf{Z}/p , p being any prime, is $(C, d), H, f, g, \phi, \lambda$, then $((C, d), H, \lambda^{-1}f, g, \lambda^{-1}\phi)$ is an AT-model over \mathbf{Z}/p . Furthermore, $\{g(h) : h \in H\}$ is a set of representative cycles of the generators of $H(C; \mathbf{Z}/p)$.*

Example 4. Consider the simplicial complex S derived from the triangulation of the Klein bottle given in Figure 1 and the chain complex $\mathcal{C}(S)$ associated to S . Let $C(S)$ be the filter of $\mathcal{C}(S)$ given in Example 1. Running the algorithm above, we obtain a 2-AT-model of $\mathcal{C}(S)$, $(C(S), H_S, f_S, g_S, \phi_S, 2)$, where $H_S = \{a, ac\}$. The value of f_S on each vertex of S is $2a$. The value of f_S on each edge marked in red in Figure 1, is $2ac$. The value of f_S on the rest of the simplices of S is zero. For the map g_S , we obtain that $g_S(a) = a$ and $g_S(ac) = ac - bc - ab$. On each vertex of S , ϕ_S gives a path connecting this vertex with a , multiplied by 2; for example, $\phi_S(g) = 2(ab + bc + cg)$. On the edges of S , the key idea is the same, that is, on each edge of S , ϕ_S gives a “path” connecting this edge with ac , multiplied by 2; for example, $\phi_S(gh) = 2(fgh - cfg + bcf + abf)$. On each triangle of S , the value of ϕ_S is zero.

Summing up, given a filter C of a chain complex \mathcal{C} , it is possible to compute a λ -AT-model, $\lambda AT = (C, d), H, f, g, \phi, \lambda$, in $O(m^3)$ if C has m elements. The Betti numbers and a set of independent non-boundary cycles of \mathcal{C} over \mathbf{Z} can directly be obtained from λAT . Moreover, the integer λ provides the prime numbers involved in the invariant factors of the torsion subgroup of $H(C; \mathbf{Z})$. This last information will be essential in the next section for designing an algorithm for computing “moduli” representative cycles of the generators of the free and the torsion subgroups of $H(C; \mathbf{Z})$.

4 Computing Integer Homology Information

As we have said before, a λ -AT-model for a given chain complex \mathcal{C} provides information of the free subgroup of $H(C; \mathbf{Z})$ as well as the prime numbers involved in the invariant factors of $H(C; \mathbf{Z})$. For obtaining “moduli” representative cycles of the generators of the free and the torsion subgroups of $H(C; \mathbf{Z})$ we only have to compute AT-models for \mathcal{C} over \mathbf{Z}/p , for each prime p dividing λ . Observe that for this last task, since we work with coefficients in \mathbf{Z}/p , we can use either Algorithm 1 or Algorithm 2.

Algorithm 4. *Computing integer homology information and “moduli” representative cycles of homology generators of a chain complex \mathcal{C} .*

INPUT: a filter $C = \{a_0, \dots, a_m\}$ of the chain complex \mathcal{C} of dim n ,
 and the matrix of the differential d for the basis C .

Apply Algorithm 2 with coefficients in \mathbf{Z} for computing a
 λ -AT-model for (C, d) , $((C, d), H, f, g, \phi, \lambda)$;

$\beta_q :=$ number of elements of H of dim q , for $q = 0$ to n ;

$G := \{g(h) : h \in H\}$.

For each prime p dividing λ do

Apply Algorithm 2 with coefficients in \mathbf{Z}/p for computing an
 AT-model for \mathcal{C} over \mathbf{Z}/p , $((C, d_p), H_p, f_p, g_p, \phi_p)$;

$T_{(0,p)} = \beta_{(0,p)} - \beta_0$;

$T_{(q,p)} = \beta_{(q,p)} - \beta_q - T_{(q-1,p)}$, for $q = 1$ to n ;

$G_p = \{g_p(h_p) : h_p \in H_p\}$.

OUTPUT: The sets G , $\{G_p : p \text{ being a prime dividing } \lambda\}$, $\{\beta_1, \dots, \beta_n\}$,
 and $\{T_{(q,p)} : 0 \leq q \leq n \text{ and } p \text{ being a prime dividing } \lambda\}$.

Summing up, after computing a λ -AT-model and an AT-model over \mathbf{Z}/p , for each p dividing λ , for a given chain complex \mathcal{C} , we obtain:

- the Betti numbers β_q for $0 \leq q \leq n$, and a set G of independent non-boundary cycles of \mathcal{C} over \mathbf{Z} (in fact, G is also a set of generators of $H(\mathcal{C}; \mathbf{Q})$);
- the prime numbers p involved in the invariant factors corresponding to the torsion subgroup of $H(\mathcal{C}, \mathbf{Z})$, the amount of invariant factors in each dimension q that are a power of p , $T_{(q,p)}$, and a set G_p of representative cycles of the generators of $H(\mathcal{C}; \mathbf{Z}/p)$ for each prime p dividing λ .

Example 5. In Example 1, we applied the Algorithm 2 and computed a 2-AT-model for $\mathcal{C}(S)$ and the Betti numbers of \mathcal{C} ; $\beta_0 = 1$, $\beta_1 = 1$ and $\beta_2 = 0$. Now, we apply Algorithm 2 for compute an AT-model for (C, d) with coefficients in $\mathbf{Z}/2$ to obtain $((C, d), H_2, f_2, g_2, \phi_2)$ where $H_2 = \{a, ac, de, adi\}$. Then, $\beta_{(0,2)} = 1$, $\beta_{(1,2)} = 2$, $\beta_{(2,2)} = 1$ and $G_2 = \{a, ac + bc + ab, ad + de + ae, adf + abf + bcf + cfg + acg + aeg + def + efh + fgh + ghi + egi + dei + aeh + abh + bhi + bci + aci + adi\}$. Therefore, $t_{(0,2)} = 0$, $t_{(1,2)} = 1$ and $t_{(2,2)} = 0$. We conclude that $H_0(S) = \mathbf{Z}$ and $H_1(S) = \mathbf{Z} \oplus \mathbf{Z}/2$.

5 Conclusions and Future Work

A λ -AT-model for a chain complex \mathcal{C} can be computed in cubic time. It provides information of the free subgroup of $H(\mathcal{C}; \mathbf{Z})$ and also the primes p that are candidates to be involved in an invariant factor of $H(\mathcal{C}; \mathbf{Z})$. For obtaining the amount of invariant factors that are a power of p and “moduli” representative cycles of homology generators, we compute an AT-model with coefficients in \mathbf{Z}/p for such primes p .

A future work is to study if it is possible to obtain generators with integer coefficients of the torsion subgroup of $H(\mathcal{C}; \mathbf{Z})$. Another task is to study if cohomology features can be computed over \mathbf{Z} from a λ -AT-model.

Concerning to the complexity, Algorithm 4 runs in time $O(m^3\psi(\lambda))$ in the worst case, ψ being the Euler function. Therefore, one important question is to

bound the coefficient λ . In order to improve the complexity we might first compute a chain contraction to obtain a smaller chain complex with same homology in the integer domain and apply Algorithm 4 to a “thinned” complex.

References

1. Dahmen, W., Micchelli, C.A.: On the Linear Independence of Multivariate b-splines I. Triangulation of simploids. *SIAM J. Numer. Anal.*, vol. 19 (1982)
2. Dumas, J.G., Saunders, B., Villard, G.: On Efficient Sparse Integer Matrix Smith Normal Form Computations. *J. of Symbolic Computation* (2001)
3. Edelsbrunner, H., Letscher, D., Zomorodian, A.: Topological Persistence and Simplification. In: *Proc. 41st Symp. on Foundations of Computer Science*, pp. 454–463 (2000)
4. Gui, X., Havas, G.: On the Worst-case Complexity of Integer Gaussian Elimination. In: *Proc. of ISSAC 1997*, pp. 28–31 (1997)
5. González-Díaz, R., Real, P.: Computation of Cohomology Operations on Finite Simplicial Complexes. *Homology, Homotopy and Applications* 5(2), 83–93 (2003)
6. González-Díaz, R., Real, P.: Towards Digital Cohomology. In: Nyström, I., Sanniti di Baja, G., Svensson, S. (eds.) *DGCI 2003. LNCS*, vol. 2886, pp. 92–101. Springer, Heidelberg (2003)
7. González-Díaz, R., Real, P.: On the Cohomology of 3D Digital Images. *Discrete Applied Math.* 147, 245–263 (2005)
8. Gonzalez-Diaz, R., Medrano, B., Real, P., Sánchez-Peláez, J.: Reusing Integer Homology Information of Binary Digital Images. In: Kuba, A., Nyúl, L.G., Palágyi, K. (eds.) *DGCI 2006. LNCS*, vol. 4245, pp. 199–210. Springer, Heidelberg (2006)
9. Gonzalez-Diaz, R., Medrano, B., Real, P., Sánchez-Peláez, J.: Simplicial Perturbation Technique and Effective Homology. In: Ganzha, V.G., Mayr, E.W., Vorozhtsov, E.V. (eds.) *CASC 2006. LNCS*, vol. 4194, pp. 166–177. Springer, Heidelberg (2006)
10. Iliopoulos, O.S.: Worst-case Complexity Bounds on Algorithms for Computing the Canonical Structure of Finite Abelian Groups and the Hermite and Smith Normal Forms of an Integer Matrix. *SIAM J. Comput.* 18, 658–669 (1989)
11. Massey, W.M.: *A Basic Course in Algebraic Topology*. New York (1991)
12. Kaczynski, T., Mischaikow, K., Mrozek, M.: *Computational Homology*. Applied Mathematical Sciences, vol. 157. Springer-Verlag, Heidelberg (2004)
13. Munkres, J.R.: *Elements of Algebraic Topology*. Addison–Wesley Co., London (1984)
14. Peltier, S., Alayrangués, S., Fuchs, L., Lachaud, J.: Computation of Homology Groups and Generators. In: Andrès, É., Damiand, G., Lienhardt, P. (eds.) *DGCI 2005. LNCS*, vol. 3429, pp. 195–205. Springer, Heidelberg (2005)
15. Peltier, S., Fuchs, L., Lienhardt, P.: Homology of Simplicial Set. In: Kuba, A., Nyúl, L.G., Palágyi, K. (eds.) *DGCI 2006. LNCS*, vol. 4245, pp. 235–246. Springer, Heidelberg (2006)
16. Sergeraert, F.: Homologie effective. I, II. *C. R. Acad. Sci. Paris Sér. I Math.* 304, vol. 11, pp. 279–282, vol. 12, pp. 319–321 (1987)
17. Storjohann, A.: Near Optimal Algorithms for Computing Smith Normal Forms of Integral Matrices. *Proc. of ISSAC 1996*, pp. 267–274 (1996)
18. Zomorodian, A., Carlsson, G.: Computing Persistent Homology. In: *Proc. of the 20th annual Symposium on Computational Geometry*, pp. 347–356 (2004)