Homology Computation by Reduction of Chain Complexes [1]

Tomasz Kaczyński[2], Marian Mrozek[3] and Maciej Ślusarek[4]

Abstract

A new algorithm for computing the homology module of a finitely generated chain complex is given. It is based on local one-step reductions of the size of the initial chain complex and it has a clear geometrical interpretation. The complexity of the algorithm is discussed in special cases.

Keywords: *chain complex, homology, algorithm, computational complexity.*

---

[1]the first author was supported by grants from NSERC of Canada and FCAR of Québec, the second and third author were supported by KBN, Grant No. 2 P03A 029 12

[2]Département de mathématiques et d'informatique, Université de Sherbrooke (Québec), Canada; e-mail: kaczyn@dmi.usherb.ca

[3]School of Mathematics, Georgia Institute of Technology, Atlanta, Georgia, on leave from Institute of Computer Science, Jagiellonian University, Kraków, Poland; e-mail: mrozek@ii.uj.edu.pl

[4]Institute of Computer Science, Jagiellonian University, Kraków, Poland; e-mail: slusarek@ii.uj.edu.pl

1 Introduction

The increasing number of computer assisted proofs in mathematics motivates the search of effective combinatorial algorithms for mathematical constructions previously known only on a very abstract level (see [Mro], [Zei], and their references).

In particular, the applications of the Conley Index theory to discrete dynamics ([Mis–Mro1], [Mis–Mro2]) require effective computing of homology modules and homology endomorphisms of polyhedra consisting of a very large number of simplices or n-dimensional cubes (cubical grids usually are more convenient in numerical computations).

Unfortunately all known homology computing algorithms have a very poor running time. The classical approach by Munkres [Mun] yields exponential bounds. So far, the best worst-case complexity for complexes of arbitrary dimension is achieved by Illiopoulos [Ill]. If n is the number of all simplices and the dimension of the complex is fixed, his algorithm runs in time $O(n^5)$. However it turns out that when the so called "sparse" triangulations are considered the expected running time of the classical algorithm is only $O(n^2)$. This probabilistic analysis was recently done by Chang and Donald ([Cha-Don]). The worst–case complexity is significantly improved, in the case of a finite simplicial complex embedded in $\mathcal{R}^3$, by Delfinado and Edelsbrunner ([D-E]) who compute Betti numbers in that case in an almost linear time. Their method, however, does not apply to complexes in $\mathcal{R}^n$ with $n \geq 4$.

In this paper we propose an algorithm for computing homology of a finitely generated chain complex based on reducing the size of the complex to a minimum while homology is preserved in each step of the reduction. In the case of homology with coefficients in a field, we achieve a chain complex whose boundary map is trivial. Hence, the resulting complex is equal to its homology complex and there is nothing left to compute. Moreover, our reduction naturally extends to a chain map between two chain complexes thus allowing the simultaneous computation of the homomorphism induced in homology by a chain map. To our best knowledge this is the first efficient algorithm addressing this problem.

The reduction is purely combinatorial which constitutes its main strength though it results from topological considerations and it has a nice geometric flavor. However from the algorithmic point of view, the topological layer gives an unnecessary balast which complicates the algorithm.

It seems that our construction could be possibly placed in an abstract algebra framework of minimal models [Roi] or Tate resolutions [Tat], [Joz] but, since this paper is concerned with algorithms rather than an abstract theory, we leave consulting the above references to the reader interested in that topic.

In the case of a chain complex of a planar polyhedron the running time achieved by our algorithm is linear. In that case, the method of exterior face collapsing (described in Section 3) has been simultaneously explored by Szymczak who actually implemented it in a computer program which is presented in [Mi-Mro-Sz].

Although we presently can only estimate the complexity in some special cases, we expect that the running time for fine grids of low-dimensional spaces could be shorter than for previous algorithms which were directly applying the definition of homology module to the initial chain complex, since our algorithm permits localizing the computation to small areas: each one-step reduction only requires information on the neighbouring simplices and not the whole chain complex.

In Section 2, we present the general reduction from the topological point of view and we prove the correctness of the algorithm. In Section 3 geometric interpretation and examples are given. In Section 4 the algorithm is described and its complexity is discussed.

Acknowledgements. The thought to localize the homology computation via collapsing of simplices is due to Konstantin Mischaikow and we gratefully acknowledge this fact. The idea to get rid of the topological balast came to us during a discussion in June 1995 with our colleauge Gilles Fournier who unexpectedly passed away two months later.

2 One-step reduction of chain complexes

Let (C, ∂) be a finitely generated free chain complex with coefficients in a ring $\mathcal{R}$. Let $\{C_q\}_{q\in\mathbb{Z}}$ be the gradation of C and $\{\partial_q : C_q \to C_{q-1}\}_{q\in\mathbb{Z}}$ the gradation of ∂. For simplicity of notation we write ∂v for $\partial_q v$ if

$v \in C_q$ for a given q. Assume that a fixed base E_q of C_q is given for each q, and let $\langle \cdot, \cdot \rangle$ denote the associated scalar product, i.e. the bilinear form $C_q \times C_q \to \mathcal{R}$ defined on generators by

$$\langle e, e' \rangle := \begin{cases} 1 & \text{if } e = e' \\ 0 & \text{otherwise} \end{cases}, e, e' \in E_q.$$

Let $m \in \mathbb{Z}$ be a fixed number. Assume that $a \in E_{m-1}$ and $b \in E_m$ are two fixed elements such that

$$\partial b = \lambda a + r, \tag{1.1}$$

where $\langle a, r \rangle = 0$ and $\lambda \in \mathcal{R}$ is invertible in $\mathcal{R}$.

For $q \in \mathbb{Z}$, define

$$\overline{C}_q := \begin{cases} C_q & \text{if } q \notin \{m-1, m\} \\ \{v \in C_q \mid \langle v, a \rangle = 0\} & \text{if } q = m-1 \\ \{v \in C_q \mid \langle v, b \rangle = 0\} & \text{if } q = m \end{cases} \tag{1.2}$$

and for $v \in \overline{C}_q$ define

$$\overline{\partial} v := \begin{cases} \partial v & \text{if } q \notin \{m, m+1\} \\ \partial v - \lambda^{-1} \langle \partial v, a \rangle \partial b & \text{if } q = m \\ \partial v - \langle \partial v, b \rangle b & \text{if } q = m+1. \end{cases} \tag{1.3}$$

Obviously, $\overline{C}_q$ is a free submodule of C_q with a base

$$\overline{E}_q := \begin{cases} E_q & \text{if } q \notin \{m-1, m\} \\ E_q - \{a\} & \text{if } q = m-1 \\ E_q - \{b\} & \text{if } q = m. \end{cases}$$

Theorem 1. $(\overline{C}, \overline{\partial}) := \left(\left\{ \overline{C}_q \right\}_{q \in \mathbb{Z}}, \left\{ \overline{\partial}_q \right\}_{q \in \mathbb{Z}} \right)$ *is a free chain complex over* $\mathcal{R}$.

Proof. One easily verifies that $\overline{\partial}_q(v) \in \overline{C}_{q-1}$ for $v \in \overline{C}_q$ and $q \in \mathbb{Z}$. It remains to show that $\overline{\partial}_q \overline{\partial}_{q+1} = 0$ for all q. This is obvious if $q \notin \{m-1, m, m+1\}$. Let $v \in \overline{C}_{q+1}$ and assume that $q = m-1$. We have

$$\overline{\partial}^2 v = \partial(\partial v - \lambda^{-1}\langle \partial v, a\rangle \partial b) = -\lambda^{-1}\langle \partial v, b\rangle \partial^2 b = 0.$$

Assume in turn that $q = m$. Then

$$\overline{\partial}^2 v = \overline{\partial}(\partial v - \langle \partial v, b\rangle b) = \partial(\partial v - \langle \partial v, b\rangle b) - \lambda^{-1}\langle \partial^2 v - \langle \partial v, b\rangle \partial b, a\rangle \partial b = 0.$$

Finally, consider the case $q = m+1$. We get

$$\overline{\partial}^2 v = \overline{\partial}(\partial v) = \partial(\partial v) - \langle \partial(\partial v), b\rangle b = 0. \qquad \blacksquare$$

Now we define linear maps $p_q : C_q \to \overline{C}_q$, $j_q : \overline{C}_q \to C_q$, for any $q \in \mathbb{Z}$ as follows.

$$p_q v := \begin{cases} v & \text{if } q \notin \{m-1, m\} \\ v - \lambda^{-1}\langle v, a\rangle \partial b & \text{if } q = m-1 \\ v - \langle v, b\rangle b & \text{if } q = m \end{cases}, v \in C_q.$$

$$j_q v := \begin{cases} v & \text{if } q \neq m \\ v - \lambda^{-1}\langle \partial v, a\rangle b & \text{if } q = m \end{cases}, v \in \overline{C}_q.$$

Let also $i_q : \overline{C}_q \to C_q$ be the inclusion map. The following is verified instantly:

Proposition 1.
a) $\overline{\partial}_q = p_{q-1}\partial_q i_q$, *for any* $q \in \mathbb{Z}$*;*
b) $\partial_q = \partial_q i_q p_q$, *for all* $q \neq m$. $\blacksquare$

Proposition 2. *For any* $q \in \mathbb{Z}$ *and* $v \in C_q$ *we have*

$$j_q p_q v := \begin{cases} v & \text{if } q \notin \{m-1, m\} \\ v - \lambda^{-1}\langle v, a\rangle \partial b & \text{if } q = m-1 \\ v - \lambda^{-1}\langle \partial v, a\rangle b & \text{if } q = m \end{cases}.$$

Proof. The assertion is obvious for $q \neq m$. Assume that $q = m$ and $v \in C_q$. Then

$$\begin{aligned} j_q p_q v &= v - \langle v, b\rangle b - \lambda^{-1}\langle \partial(v - \langle v, b\rangle b), a\rangle b \\ &= v - \langle v, b\rangle b - \lambda^{-1}\langle \partial v, a\rangle b + \lambda^{-1}\langle v, b\rangle\langle a + \lambda r, a\rangle b \\ &= v - \lambda^{-1}\langle \partial v, a\rangle b. \qquad \blacksquare \end{aligned}$$

Proposition 3. For any $q \in \mathbb{Z}$, $p_q j_q = id$.

Proof. The assertion is obvious for $q \notin \{m-1, m\}$. Assume $q = m-1$ and $v \in \overline{C}_{m-1}$. Then

$$p_{m-1} j_{m-1} v = p_m v = v - \lambda^{-1}\langle v, a\rangle \partial b = v,$$

since $\langle v, a\rangle = 0$. Assume $q = m$ and $v \in \overline{C}_m$. Then

$$p_m j_m v = p_m(v - \lambda^{-1}\langle \partial v, a\rangle b) = p_m v = v - \langle v, b\rangle b = v,$$

since $p_m b = 0$ and $\langle v, b\rangle = 0$.

Lemma 1. *The maps* $p : C \to \overline{C}$, $p = \{p_q\}_{q\in\mathbb{Z}}$ *and* $j : \overline{C} \to C$, $j = \{j_q\}_{q\in\mathbb{Z}}$ *are chain maps.*

Proof. a) $p_{q-1}\partial_q = \overline{\partial}_q p_q$ for all q:
The assertion follows immediately from Proposition 1 for $q \neq m$.
Assume $q = m$ and $v \in C_m$. Then

$$\begin{aligned} \overline{\partial} p_m v &= \overline{\partial}(v - \langle v, b\rangle b) = \partial(v - \langle v, b\rangle b) - \lambda^{-1}\langle \partial(v - \langle v, b\rangle b, a\rangle \partial b \\ &= \partial v - \langle v, b\rangle \partial b - \lambda^{-1}(\langle \partial v, a\rangle - \langle v, b\rangle\langle \partial b, a\rangle)\partial b \\ &= \partial v - \lambda^{-1}\langle \partial v, a\rangle \partial b = p_{m-1}\partial v. \end{aligned}$$

b) $j_{q-1}\overline{\partial}_q = \partial_q j_q$ for all q:
The assertion is obvious for $q \notin \{m, m+1\}$. Assume $q = m$ and $v \in \overline{C}_m$. By Propositions 2 and 1 we have

$$\begin{aligned} \partial j_m v &= \partial(v - \lambda^{-1}\langle \partial v, a\rangle b) = \partial v - \lambda^{-1}\langle \partial v, a\rangle \partial b \\ &= j_{m-1} p_{m-1} \partial v = j_{m-1}\overline{\partial} v. \end{aligned}$$

Assume $q = m+1$ and $v \in \overline{C}_{m+1}$. By Proposition 2 we have

$$\partial j_{m+1} v = \partial v = \partial v - \lambda^{-1}\langle \partial(\partial v), a\rangle b = j_m p_m \partial v = j_m \overline{\partial} v. \qquad \blacksquare$$

Theorem 2. $H(\overline{C}) = H(C)$.

Proof. By Lemma 1, $p : C \to \overline{C}$ and $j : \overline{C} \to C$ are chain maps, and by Proposition 3, $pj : \overline{C} \to \overline{C}$ is the identity. It is sufficient to show that $jp : C \to C$ is chain homotopic to the identity.

We define the chain homotopy $\varphi = \{\varphi_q\}$, $\varphi_q : C_q \to C_{q+1}$ by

$$\varphi_q v = \begin{cases} \lambda^{-1}\langle v, a\rangle b & \text{if } q = m-1 \\ 0 & \text{otherwise} \end{cases}$$

It remains to verify the condition

$$\mathrm{id} - j_q p_q = \partial_{q+1}\varphi_q + \varphi_{q-1}\partial_q. \tag{1.4}$$

for all q. This is obvious if $q \notin \{m-1, m\}$. Assume $q = m-1$ and $v \in C_{m-1}$. The left-hand side of (1.4) is $\lambda^{-1}\langle v, a\rangle \partial b$. The right-hand side is

$$\partial\varphi_{m-1} v = \partial(\lambda^{-1}\langle v, a\rangle b) + 0 = \lambda^{-1}\langle v, a\rangle \partial b.$$

Assume $q = m$ and $v \in C_m$. The left-hand side of (1.4) is $\lambda^{-1}\langle \partial v, a\rangle b$. The right-hand side is $\varphi_{m-1}\partial v = \lambda^{-1}\langle \partial v, a\rangle b$. ■

Let now (C^0, ∂^0), (C^1, ∂^1), $(C^2, \partial^2), \ldots$ be a sequence of chain complexes obtained from (C, ∂) by iterating construction (1.2) – (1.3) for fixed $m \in \mathbb{Z}$ as long as the choice of a, b in (1.1) is possible. Thus $\{(C^k, \partial^k)\}$ has the property

$$(C^0, \partial^0) = (C, \partial),\ (C^{k+1}, \partial^{k+1}) = (\overline{C}^k, \overline{\partial}^k),\ \text{for } k = 0, 1, 2, \ldots \tag{1.5}$$

Denote $E^{k+1}_q = \overline{E}^k_q$, for $q \in \mathbb{Z}$, and $N(k) = \sum_q \operatorname{card} E^k_q$, for $k = 0, 1, 2, \ldots$ Since (C, ∂) is finitely generated, $N(k) < \infty$ and strictly decreases, therefore, there exists a final element of that sequence denoted by (C^{fm}, ∂^{fm}), beyond which the construction cannot be extended.

Theorem 3. If $\mathcal{R}$ is a field, then $\partial^{fm}_m = 0$ and

$$H(C) = H(C^{fm}).$$

Proof. The identity $H(C) = H(C^{fm})$ follows from Theorem 2 by recurrence. If $\mathcal{R}$ is a field then λ in (1.1) is invertible if and only if

$$\langle \partial b, a\rangle \neq 0. \tag{1.6}$$

Therefore the construction can be iterated as long as there exist $a \in E^k_{m-1}$ and $b \in E^k_m$ satisfying (1.6), i.e. as long as $\partial^k_m \neq 0$. ■

Let $d(C)$ denote the largest $q \in \mathbb{Z}$ such that (C, ∂) has a nonempty base E_q. We perform the sequence of iterations (1.5) first for $m = d(C)$ and we obtain $(C^{f_{d(C)}}, \partial^{f_{d(C)}})$ as the final element of (1.5). Next we do the same for $m = d(C) - 1$, this time with $(C^{f_{d(C)}}, \partial^{f_{d(C)}})$ as the first item in the sequence (1.5). Again, the final complex $C^{f_{d(C)-1}}$ serves as the initial one for computing (1.5) with $m = d(C) - 2$ and so on. The last sequence of iterations (1.5) is performed for $m = 1$. Denote $(C^f, \partial^f) = (C^{f_1}, \partial^{f_1})$, i.e. the final element of the last sequence. Theorem 3 implies immediately the following corollary.

Corollary 1. If R is a field then $\partial^f = 0$ and, consequently,

$$H(C) = H(C^f) = C^f.$$

Remark 1.

Let us observe that once the computation of C^{f^m} is performed with $m = d(C)$ all m-generators of $C^{f^m}_m$ can be removed from the data structure since their boundaries are trivial. Therefore the further computation can be restricted to the $(m-1)$-skeleton of C^{f^m} and so on.

Remark 2.

Given a triple (C, D, φ) where C, D are finite chain complexes with coefficients in R and $\varphi : C \to D$ is a chain map, the reduction performed either on C or D induces the reduction on the whole triple.

Indeed, if $\overline{C}$ is obtained from C by a one–step reduction, we define $\overline{\varphi} : \overline{C} \to D$ by $\overline{\varphi} := \varphi j$, where $j : \overline{C} \to C$ is the chain map given by Lemma 1. If $\overline{D}$ is obtained from D, we define $\overline{\varphi} : C \to \overline{D}$ by $\overline{\varphi} := p\varphi$ where $p : D \to \overline{D}$ is given for D as in Lemma 1. Since we showed in course of the proof of Theorem 2 that j and p induce isomorphisms in homology, this procedure can be iterated so that in the final stage we get a chain map $\varphi^f : C^f \to D^f$ such that the diagram

$$\begin{array}{ccc} C & \xrightarrow{\varphi} & D \\ \uparrow{\scriptstyle j^f} & & \downarrow{\scriptstyle p^f} \\ C^f & \xrightarrow{\varphi^f} & D^f \end{array}$$

commutes and the vertical arrows induce isomorphisms in homology. Thus Corollary 1 and the above discussion imply the following

Corollary 2. If R is a field then the triple (C^f, D^f, φ^f) is isomorphic to the triple $(H(C), H(D), \varphi_*)$ in the sense that the vertical arrows in the following commuting diagram are isomorphisms:

$$\begin{array}{ccc} H(C) & \xrightarrow{\varphi_*} & H(D) \\ \uparrow{\scriptstyle j^f_*} & & \downarrow{\scriptstyle p^f_*} \\ C^f & \xrightarrow{\varphi^f} & D^f \end{array}$$

3 Geometric interpretation of the reduction

To fix ideas, let C be a finite simplicial complex and E_q the set of its q-simplices (closed) for each q. Then the terms appearing in (1.1) are an m-simplex b, one of its $(m-1)$-faces a, and $\lambda = \pm 1$. Denote by $[C]$ the topological realization of C.

The complex $\overline{C}$ is obtained from C by collapsing b so that the face a is glued to the remaining $(m-1)$-faces of b. They become new faces of those m-simplices (other than b) which had a as an $(m-1)$-face. The so-obtained complex might no longer be a simplicial complex but its topological realization $[\overline{C}]$ has the structure of a CW-complex (see, e.q. [Dold]).

Here are three particular cases which are easier to visualize.

1) **Interior face reduction**

Suppose that a is a common $(m-1)$-face of exactly two m-simplices b and c. Then $\overline{C}$ is obtained from C by deleting a and replacing the

m-simplices b and c by a cell which is still denoted by $c \in \overline{E}_m$ but, geometrically, it represents the union $b \cup c$. Its boundary is $\partial b + \partial c$ (a is reduced since it appears with opposite signs in ∂b and ∂c). $\overline{C}$ is not a simplicial complex, it is a cell complex, but $[\overline{C}] = [C]$.

2) **Exterior face collapsing**

Suppose that a is an exterior face of a polyhedron of C, more precisely, b is not a face of any $(m+1)$-simplex and it is the only m-simplex having a as a face. Then $\overline{C}$ is a simplicial complex obtained by collapsing b so that a is projected onto the remaining part of the boundary of b. Thus $[\overline{C}]$ is a strong deformation retract of $[C]$. This case of our construction is inspired by [Cohen].

3) **Cell-to-point contraction**

If e is any given q-simplex of C (more generally, a closed cell with the trivial reduced homology) then the quotient space $X = [C]/e$ has the same homology as $[C]$. X can be obtained by iterating the one-step reduction for e and all its k-faces, $k = q-1, \ldots, 0$. This type of geometrical reduction corresponds to a sequence of algebraic reductions presented in Sect. 2.

Example. Let C be the boundary of a triangle with vertices **A**, **B**, **C**. More precisely $C_1 = \{a, b, c\}$, $C_0 = \{\mathbf{A}, \mathbf{B}, \mathbf{C}\}$, $C_q = 0$ for $q \notin \{0, 1\}$, and

$$\partial(a) = \mathbf{B} - \mathbf{A}, \ \partial(b) = \mathbf{C} - \mathbf{B}, \ \partial(c) = \mathbf{A} - \mathbf{C}$$

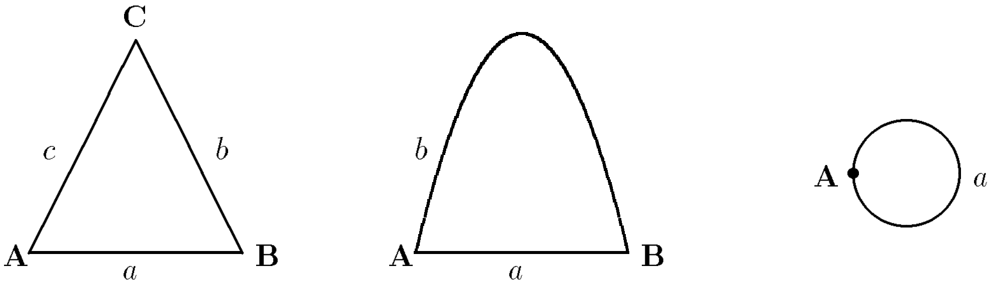
The reduction of c and **C** gives $\overline{E}_1 = \{a, b\}$, $\overline{E}_0 = \{\mathbf{A}, \mathbf{B}\}$,

$$\overline{\partial}(a) = \partial(a) = \mathbf{B} - \mathbf{A}, \ \overline{\partial}(b) = \partial(b + c) = \mathbf{A} - \mathbf{B}$$

The reduction of b and **B** gives $\overline{\overline{E}}_1 = \{a\}$, $\overline{\overline{E}}_0 = \{\mathbf{A}\}$, and $\overline{\overline{\partial}}(a) = \overline{\partial}(a + b) = 0$. Therefore

$$\overline{\overline{C}} = H\left(\overline{\overline{C}}\right) \simeq H(S^1) \simeq \begin{cases} \mathrm{R} & \text{if } q = 1, 0 \\ 0 & \text{otherwise} \end{cases}.$$

The two reductions are illustrated below.



C $\qquad\qquad$ $\overline{C}$ $\qquad\qquad$ $\overline{\overline{C}}$

Let now C be a simplicial complex of a finite planar polyhedron P. Then any edge, i.e. any 1-simplex, is a face of at most two 2-simplices, and if it is a face of exactly two, then it is in the interior of P (except possibly for its vertices). We also have the folllowing

Proposition 4. *If $C_2 \neq 0$ then there exists an edge a in E_1 which is a face of exactly one 2-simplex b in E_2.*

Proof. Let P' be the closed union of all 2-simplices in P. It is sufficient to prove the claim for the simplicial complex of P', since the edges in $P' - P$ do not bound any 2-simplex. The topological boundary $\partial P'$ of P' separates the interior of any 2-simplex in P' from $\mathcal{R}^2 - P'$, therefore it contains at least one edge. Obviously, any edge of $\partial P'$ has the required property. ■

By the above proposition, the exterior face collapsing described in Section 2 can be performed in the planar case until all 2-simplices disappear, so C^{f_2} is a simplicial complex of a graph in $\mathcal{R}^2$.

4 Description of the algorithm

4.1 Overall structure

In this chapter we show how to implement efficiently the contraction method in order to compute the homology of a finitely generated chain complex over a field $\mathcal{R}$. We assume that a single arithmetic operation in R has constant cost. We propose a suitable data structure for this task, specify the algorithm, and partially analyze its computational complexity. A straightforward implementation of a classical homology computation method by Munkres ([Mun]), when applied to computations over the field $\mathbb{Z}_2$ yields an $O(n^3)$ algorithm. For some special cases we achieve significant improvement of that bound.

Let a finitely generated chain complex (C, ∂) with coefficients in R be given together with its fixed bases E_q in each dimension $q = 0, \ldots, d(C)$ (recall that $d(C)$ denotes the maximal dimension in which (C, ∂) has a nonempty base). For a q-generator S, by a face of S we mean any $(q-1)$-generator e such that $\langle \partial S, e \rangle \neq 0$ in the sense of Section 2.

The general idea of the algorithm HOMOLOGY is to reduce one by one the number of generators with nontrivial boundaries. In general, for a fixed dimension $m \geq 1$ the reduction step goes as follows:

1. choose an m-generator S and one of its faces e,
2. for each m-generator S' such that $S' \neq S$ and e is a face of S', compute a new expression for $\partial S'$, according to definition (1.3), case $q = m$,
3. for each $(m+1)$-generator S'' such that S' is a face of S'', remove S' from $\partial S''$, that is compute a new expression for $\partial S''$ according to definition (1.3), case $q = m + 1$,
4. remove S and e from the structures associated with dimensions $m+1$, m and $m-1$.

Such reduction steps can be performed in any order and repeated until there is no object with a nonempty boundary. Here we perform these steps starting from the highest dimension $d(C)$ and repeat until there is no possibility of reduction at this dimension. Then we move to dimension $d(C) - 1$, and so on. Observe that if we perform reductions in this order step 3 is never executed since the boundary of any $(m+1)$-generator is zero.

For a fixed dimension the process is controlled by procedure COLLAPSE and a single reduction step is executed by procedure REDUCE, described further in more detail.

```
Algorithm HOMOLOGY:

input: finitely generated chain complex C
output: nonnegative integers k(0), ..., k(d(C))
```

```
for q=d(C) downto 1
do
   COLLAPSE (q) ;
   k(p) = card E(p) ;
end.

procedure COLLAPSE (q) ;
   while E(q) nonempty and
         there exist S in E(q) and e in E(q-1) s.t.
         e is a face of S
   do
         REDUCE (S,e) ;
end.
```

Proposition 5. *After applying procedure COLLAPSE to a finitely generated chain complex (C, ∂) and a given dimension m, we obtain a new chain complex $(\overline{C}, \overline{\partial})$ s.t. $\overline{\partial}_m = 0$.*

Proof. Apply Theorem 3. ■

Proposition 6. *The q-dimensional homology of a finitely generated chain complex C is equal to $R^{k(q)}$, for each $q = 0, \dots, d(C)$, where $k(0), \dots, k(d(C))$ are the numbers computed by the algorithm HOMOLOGY.*

Proof. Follows from Corollary 1. ■

4.2 Complexity analysis

We now proceed with the description of procedure REDUCE. Assume that C has n_q generators in dimension q, and let $n = \max\{n_q : q = 0, \dots, d(C)\}$. Fix m in the range $1, \dots, d(C)$ and let $e \in E_{m-1}$. By $\deg(e)$ we mean the cardinality of the co-boundary of e, i.e

$$\deg(e) = \mathrm{card}\{S \in E_m : \langle \partial S, e\rangle \neq 0\}$$

Note that for $m = 1$ $\deg(e)$ is the degree of a node of a (multi)graph which is a widely used notation.

Procedure REDUCE operates on the following data structures:

1. An array T_m of all m-generators,
2. An array T_{m-1} containing all faces of m-generators,
3. For each m-generator S a list $F(S)$ of its faces with their coefficients in ∂S, sorted by faces' names.

```
procedure REDUCE(S,e) ;

for each S' in Tm do
  if e appears in F(S') then
    compute new list F(S') by means of synchronous scan
    (i.e. merge) of F(S) and F(S')

end Reduce.
```

Complexity analysis of algorithm HOMOLOGY in the above form is simple. In procedure COLLAPSE finding a pair (S, e) for the reduction costs $O(n)$. Since the upper bound for $\deg(e)$ is $O(n)$ and similarly for the length of $F(S)$ and $F(S')$ then there are $O(n)$ merges of lists of lenght $O(n)$. Therefore a single execution of REDUCE costs $O(n^2)$ and assuming that $d(C)$ is fixed we have:

Proposition 7. *Homology groups of a finitely generated chain complex with coefficients in a field R can be computed in $O(n^3)$ arithmetical operations.* ∎

In a few cases we can give much better estimation for the complexity of algorithm HOMOLOGY. One such situation is when we consider a simplicial complex C of a finite polyhedron in a euclidean space. In such the case for the highest dimension $d(C)$ the assumptions of the following proposition are satisfied.

Proposition 8. *Assume that a finitely generated chain complex C is given. If for each $(d(C) - 1)$-face e of C it holds that $deg(e) \leq 2$ then* *procedure COLLAPSE computes $d(C)$-dimensional homologies of C in time $O(n^2)$.*

Proof. Observe that the maximal dimensional homology coincides with the group of maximal dimensional cycles. However, since the procedure COLLAPSE replaces C by $\overline{C}$ with the same homologies and $\overline{\partial} = 0$, $H_{d(C)}(C) = \overline{C}_{d(C)}$. Thus in this case a single run of procedure COLLAPSE suffices for computing the maximal dimensional homology. It remains to establish the complexity.

For this end, observe that the condition $\deg(e) \leq 2$ once satisfied for all faces, becomes an invariant for all faces during the whole algorithm execution. Hence there occurs only one merge operation in procedure REDUCE which costs $O(n)$. This gives the requested time bound. ∎

Another interesting situation concerns 1-dimensional complexes, i.e. graphs. Observe that 0-dimensional homologies can be computed by finding the number of connected components of the graph which can be performed in linear time by clasical graph scan methods like depth-first search. What concerns 1-dimensional homologies, they can be specified by computing the number of fundamental cycles in the graph. However for a connected graph this needs no computation at all: this number is equal to $n_1 - n_0 + 1$, as is easily seen from the Euler characteristic formula.

In the light of the above let us change to the case of polyhedron triangulation assuming the polyhedron is embedded in the plane $\mathcal{R}^2$.

Proposition 9. *Assume that a triangulation C in the plane $\mathcal{R}^2$ is given. Then the homology of C can be computed in linear time.*

Proof. By Proposition 4 initially there exists an edge e which is a face of exactly one triangle S. Apply procedure REDUCE to (S, e). Observe that after that either at least one of remaining two edges of S has its co-boundary size equal to 1 or there is another face of different triangle with this property. Therefore at each step we can choose an edge of degree 1 for the reduction. However, reduction of such an edge does not increase the boundary of any triangle hence each reduction step takes constant time. Additionally for each edge we keep a list (of length 2 maximum) of its co-boundary and modify it while merging boundaries. This allows for finding a suitable pair (S, e) for the reduction in constant time and completion of the whole COLLAPSE in

$O(n)$ time, for 2-dimensional homologies. By Theorem 3 the homology of C coincides with the homology of the resulting 1-dimensional complex and we are done by the preceding discussion. ∎

4.3 Practical considerations

Let (S, e) be the pair chosen for the reduction. Observe that $\deg(e)$, the co-boundary size of e, is a crucial factor influencing the amount of computation. Therefore it is extremely useful to choose a face with the minimum co-boundary. However, if we are given such a face e we should be able to find quickly any generator S belonging to the co-boundary of e. Hence the following modification of the data structure is proposed:

1. Keep the list $F(S)$ of faces of each S (together with their coefficients in ∂S) in a form of a balanced tree, thus making possible search, insert, delete and update of any element in $O(\log n)$ time [Cor].
2. Introduce a balanced tree $A(e)$ for the co-boundary of each edge separately
3. Maintain $\deg(e)$ for each face e, for example in array T_{m-1}.

In this version of our algorithm procedure COLLAPSE first searches for an edge e with lowest $\deg(e)$ and takes any S from $A(e)$. The suitably modified procedure REDUCE has the following form.

```
procedure REDUCE(S,e) ;  // version 2

for each S' in A(e) do
  for each f in F(S) do
    search for f in F(S') and compute
      new coefficient for f in F(S') ;
    if f not found
    then
         add f to F(S') ;
         add S' to A(f) ;
         increase deg(f) by 1
    else
         update coefficient for f in F(S') ;
         if new coefficient = 0
```

```
        then remove f from F(S') ;
             remove S' from A(f) ;
             decrease deg(f) by 1

end Reduce.
```

Assume that $\deg(e) = K$ and the number of generators in ∂S is B. Then a single call of REDUCE costs $O(KB \log n)$ plus $O(n)$ for finding a face e with the lowest $\deg(e)$, performed in procedure COLLAPSE. It is expected that in most applications $\deg(e)$, where e is chosen optimally as above, will remain constant for a fixed dimension. This would give an upper bound of $O(n \log n)$ for every single REDUCE call and $O(n^2 \log n)$ for the whole homology computation.

A very special situation occurs when the boundaries are short (even after a long sequence of reductions). It can be the case, for example, with large complexes built of many relatively small subcomplexes separated in high dimensions and connected only, for example, by 1-dimensional edges. Formally this situatiuon is interesting if $KB = o(n/\log n)$. Then it is useful to introduce a data structure for faster search for minimal co-boundary size face e in procedure COLLAPSE. This can be provided by a priority queue PQ for faces, $\deg(e)$ being the priority, the smallest the first. It is possible to implement PQ (e.g. by a heap – see [Cor]) in such a way that its creation takes linear time and deleting minimum and changing priority costs $O(\log n)$ each. Then creating PQ would be performed before each call of COLLAPSE, deleting minimum before each call of REDUCE, and changing priority would accompany each modification of the co-boundary size of any face.

With this last refinement of the algorithm we would get rid of the $O(n)$ term in the complexity for REDUCE, getting $O(KB \log n)$ bound which is sublinear in this case.

4.4 Homology of a simplicial map

Assume that C, D are finitely generated chain complexes and $\varphi : C \to D$ is a simplicial map. For each $q \in \mathbb{Z}$ let A_q be the matrix of $\varphi_q : C_q \to D_q$ with respect to fixed bases E_q and F_q in C_q and D_q, respectively.

1. Given the reduction of elements a and b in C_{m-1} and C_m we define the matrix $\overline{A}_q$ of $\overline{\varphi}_q : \overline{C}_q \to D_q$, for each q, by specifying its columns, using the definition of the map j from Lemma 1. If $q \neq m$ then $\overline{A}_q = A_q$. Assume then $q = m$. Let Col(e) be the column of A_q which corresponds to $e \in E_m$, let $\lambda = \langle \partial b, a \rangle \neq 0$, and let $c_e = \langle \partial e, a \rangle$, $e \in \overline{E}_m = E_m - \{b\}$.

If $c_e \neq 0$ then we put $\overline{\mathrm{Col}}_e = \mathrm{Col}_e$, otherwise $\overline{\mathrm{Col}}_e = \mathrm{Col}_e - c_e \lambda^{-1} \mathrm{Col}_b$.

2. Given the reduction of elements a and b in D_{m-1}, D_m we define the matrix $\overline{A}_q$ of $\overline{\varphi}_q : C_q \to \overline{D}_q$, for each q, by specifying its rows, using the definition of the map p from Lemma 1.

If $q \neq m-1, m$ then $\overline{A}_q = A_q$. If $q = m$ we obtain $\overline{A}_q$ by deleting the row Row_b which corresponds to b in A_q. Assume then $q = m-1$ and let $f \in \overline{F}_{m-1} = F_{m-1} - \{a\}$. Let Row_f be the row of A_{m-1} which corresponds to f. Let Row_a be the row which corresponds to a and let $d_f = \langle \partial b, f \rangle$. It is easy to check that the row in the matrix $\overline{A}_{m-1}$ which corresponds to f is given by

$$\overline{\mathrm{Row}}_f = \mathrm{Row}_f - \lambda^{-1} d_f \mathrm{Row}_a$$

For a fixed q the computation of $\overline{\varphi}_q : \overline{C} \to \overline{D}$ cosists then in two iterated transformations of the matrix A_q, each one taking $O(n^2)$ time. According to the remark given in the end of Section 2 we can iterate the above matrix transformations until there is no possibility for a one-step reduction, finally obtaining the homomorphism $\varphi^* : H(C) \to H(D)$ corresponding to the simplicial map $\varphi : C \to D$. Since the dimension of the complexes is fixed we get an algorithm for computing the homology of a chain map of complexity $O(n^3)$.

5 References

[Cha-Don] D.R.Chang and B.R.Donald, *On computing the homology type of a triangulation*, preprint 1994.

[Cohen] M.M.Cohen, *A Course in Single Homotopy Theory*, Springer 1973.

[Cor] T.H.Cormen, Ch.E.Leiserson, and R.L.Rivest, *Introduction to Algorithms*, McGraw-Hill 1990.

[D-E] C.J.A.Delfinado and H.Edelsbrunner, An incremental algorithm for Betti numbers of simplicial complexes on the 3-sphere, *Computer Aided Geometric Design*, **12** (1995), 771–784.

[Dold] A.Dold, *Lectures on Algebraic Topology*, Springer 1972.

[Ill] C.S.Illiopoulos, Worst-case complexity bounds on algorithms for computing the canonical structure of finite abelian groups and Hermite and Smith normal form of an integer matrix, *SIAM J.Computing*, **18** (1989), 658-669.

[Joz] T.Józefiak, Tate resolutions for commutative graded algebras over a local ring, *Fund. Math.* **74** (1972), 209-231.

[Mis-Mro1] K.Mischaikow and M.Mrozek, Chaos in Lorenz equations: a computer assisted proof, *Bull. Amer. Math. Soc. (N.S.)* **32** (1995), 66-72.

[Mis-Mro2] K.Mischaikow and M.Mrozek, Chaos in Lorenz equations: a computer assisted proof, part II. Details, *Mathematics of Computation*, to appear.

[Mis-Mro-Sz] K.Mischaikow, M.Mrozek, and A.Szymczak, Chaos in the Lorenz equations: A computer assisted proof; Part III: Classical parameter values, preprint.

[Mro] M.Mrozek, Topological Invariants, Multivalued Maps and Computer Assisted proofs in Dynamics, *Computers and Mathematics with Appl.*, **32**(1996), 83-104.

[Mun] J.R.Munkres, *Elements of Algebraic Topology*, Addison-Wesley 1984.

[Roi] A.Roig, Minimal resolutions and other minimal models, *Publ. Math.* **37** (1993), 285-303.

[Tat] J.Tate, Homology of noetherian rings and local rings, *Ill. J. of Math.* **1**, (1957), 14-27.

[Zei] D.Zeilberger, Theorems for a price: tomorrow's semi-rigrous mathematical culture, *The Mathematical Intelligencer* **16** (1994), 11-14, 76.