

Computation of homology groups and generators

Samuel Peltier¹, Sylvie Alayrangués², Laurent Fuchs¹, and Jacques-Olivier Lachaud²

¹ SIC (FRE 2731 CNRS), Université de Poitiers, Boulevard Marie et Pierre Curie,
86962 Futuroscope Chasseneuil Cedex, France
`{peltier,fuchs}@sic.univ-poitiers.fr`

² LaBRI, Université Bordeaux 1, 351 cours de la Libération, 33405 Talence Cedex,
France
`{alayrang,lachaud}@labri.fr`

Abstract. Topological invariants are extremely useful in many applications related to digital imaging and geometric modelling, and homology is a classical one. We present an algorithm that computes the whole homology of an object of arbitrary dimension: Betti numbers, torsion coefficients and generators. Results on classical shapes in algebraic topology are presented and discussed.

1 Introduction

In digital image analysis, shape invariants are useful for classification, indexation, or, more recently, shape description [ACZ04]. They can be used in object simplification and object thinning. In solid modeling, shape invariants ensure the consistency of constructive operations. Computing topological invariants of objects has thus a significant impact in these domains. The fundamental group is an invariant that carries most of the topological information about an object. It has been studied by many authors [Kon89,Box99,Mal01,ADFQ03] in the image analysis field. But the comparison of such groups is highly related to undecidable problems [Mal01]. Many authors have proposed algorithms to compute the Euler characteristic (some of them summarized in [KR89]), but it is a simpler and less expressive topological invariant. Other approaches compute the Betti numbers [DE95] of embedded objects.

We focus here on homology groups, which are known to be computable in finite dimensions, and which have a good topological characterization power at least in low dimensions. We not only compute these groups but also their generators, to delineate the topological holes on the shapes. For instance, the generators of the homology group of dimension 1 are connectivity lines of the shape: cutting along such lines does not divide the shape into two parts. The contributions of this work are: (i) we report recent works in computational group theory and bring these results to the imagery community, (ii) we combine these works to classical results in homology theory to compute the homology groups (Betti numbers and torsion) and their generators, (iii) we effectively implement these algorithms with numerous optimizations.

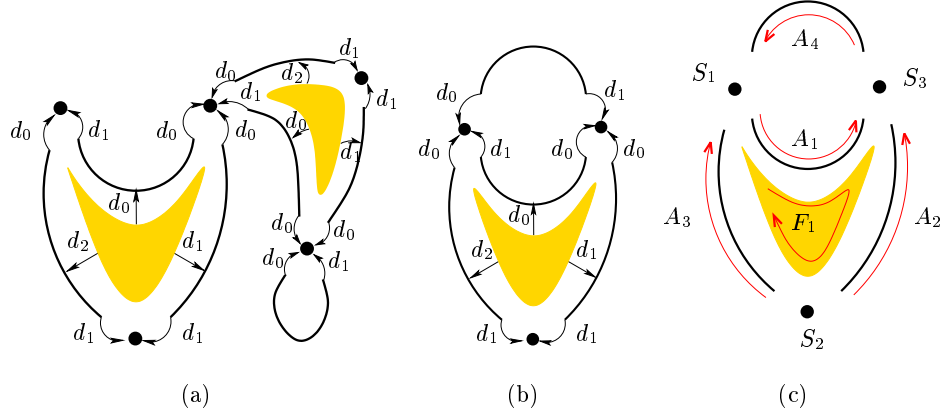


Fig. 1. On (a) and (b), examples of semi-simplicial sets. On (c), positive orientation of the simplices of (b).

In the first part of the paper we recall classical definitions in homology theory. We choose here simplicial homology since it is widely used in geometric modeling and is straightforwardly applicable to digital objects. We then present related works. After that, we present our approach for computing homology groups: Smith Normal Form (SNF) of the boundary homomorphisms, modified SNF to compute generators, integer computations performed with a modulo. Lastly, we show some experiments and list some perspectives to this work.

2 Simplicial homology

Semi-simplicial set. Shapes are classically modeled with a cellular subdivision. Several combinatorial structures may represent such a subdivision. We choose here semi-simplicial sets, which can represent indifferently manifold or non-manifold objects. This structure is a subclass of simplicial sets, a structure studied in algebraic topology [May67, Cur71].

Definition 1. [May67] A semi-simplicial set $S = (K, (d_i^q))$ is a graded family of sets $K = (K^q)_{q \in \mathbb{N}}$ together with maps $d_i^q : K^q \rightarrow K^{q-1}$ for $i = 0, \dots, q$, which satisfy the following identity: $\forall \sigma \in K^q, d_j^{q-1}(d_i^q(\sigma)) = d_{i-1}^{q-1}(d_j^q(\sigma))$ if $j < i$.

The elements of K^q are called q -simplices. The d_i^q are called *boundary operators* (the subscripts q will generally be dropped later for clarity). Simplices are glued together consistently with these operators (see Fig 1a-b for two examples).

Semi-simplicial sets are clearly adapted to the constructive operations of solid modeling [LL95]. They are also well suited to digital imagery [DG03]. To determine a semi-simplicial set that represents a given digital object, the first step is to construct a simplicial analog. One method is proposed in [GDR03] (see

Section 3). The second step is to number the vertices of the simplicial analog; the boundary maps follow directly [LE93].

We can now introduce homology groups in an intuitive way. All objects are assumed to be finite. Note that the homology theory is applicable on most combinatorial structures.

Chain, boundary homomorphism, chain complex. In a first step, we define group structures on semi-simplicial sets. A p -chain in K^p is a linear combination of p -simplices with integer coefficients. More formally, any p -chain is written uniquely as a finite sum $\sum_{i=1}^{n_p} \alpha_i^p \sigma_i^p$, where n_p is the cardinal of $K^p = \{\sigma_1^p, \dots, \sigma_{n_p}^p\}$, and for all i , α_i^p is an integer. The addition over p -chains is defined simply by adding coefficients simplex by simplex. The resulting groups are denoted by C_p . For all p , K^p forms a basis for C_p (see [Mun84] p.28).

A p -chain is a purely formal construction. The coefficients α_i have generally not a geometric interpretation, except for the coefficients 1 and -1 . In this case $1 \cdot \sigma$ means that we consider the simplex σ with its orientation and $-1 \cdot \sigma$ means that we consider the simplex σ with its opposite orientation. This is consistent with the fact that simplices can be equipped with two orientations, one considered positive and the other negative. Fig. 1(c) displays the positive orientations of the simplices of Fig. 1(b). A formal definition of simplex orientation is available in classical algebraic topology books [Mun84,Hat02].

In a second step, we relate chain groups of successive dimension with homomorphisms called boundary operators.

Definition 2. For all $p > 0$, the boundary of a p -simplex σ^p , denoted by $\partial_p(\sigma^p)$, is the $(p-1)$ -chain $\sum_{i=0}^p (-1)^i d_i(\sigma)$. 0-simplices have an empty boundary. The boundary is extended as an homomorphism from C_p to C_{p-1} , meaning for any p -chain $c = \sum_{i=1}^{n_p} \alpha_i^p \sigma_i^p$, its boundary $\partial_p(c)$ is equal to $\sum_{i=1}^{n_p} \alpha_i^p \partial_p(\sigma_i^p)$.

Usually, when no confusion may arise, we simply write $\partial(c)$ for the boundary of a p -chain c . For example, on Fig. 1c, we have $\partial(F) = A_1 - A_2 + A_3$ and we can verify that $\partial(\partial(F)) = \partial(A_1 - A_2 + A_3) = 0$.

We have just constructed a sequence of chain groups C_p together with homomorphisms $\partial_p, C_n \xrightarrow{\partial_n} C_{n-1} \xrightarrow{\partial_{n-1}} \dots \xrightarrow{\partial_1} C_0 \xrightarrow{\partial_0} 0$. One can check that $\partial_{p-1}(\partial_p(c)) = 0$ for all p -chains c . This sequence is called a *free chain complex*.

Cycle, boundary, hole. The homology groups of a combinatorial object are derived from specific subgroups of the chains of a free chain complex.

The p -chains whose boundary is empty are called p -cycles. For example, on Fig. 1c, the 1-chains $A_1 - A_2 + A_3$ and $A_1 + A_4$ are 1-cycles. The set of p -cycles is a subgroup of C_p , denoted by Z_p .

Some p -chains are the boundary of a $(p+1)$ -chain. They are called p -boundaries. For example, on Fig. 1c, the 1-chain $A_1 - A_2 + A_3$ is the boundary of the 2-chain F . The set of p -boundaries form a subgroup of C_p , denoted by B_p . Since $\forall c \in C_p, \partial_{p-1}(\partial_p(c)) = 0$, we have $B_p \subset Z_p \subset C_p$.

A *p-dimensional hole* is a *p*-cycle which is not a *p*-boundary. For example, on Fig. 1c, the 1-cycle $z_1 = A_1 + A_4$ is not a boundary. We define an equivalence relation in the group of *p*-cycles as follows: two *p*-cycles *s* and *t* are in the same equivalence class iff there exist a chain *c* with $s = t + \partial_{p+1}c$. They are then said to be *homologous*. In particular, when $s = \partial_{p+1}c$ then *s* is homologous to 0. The set of cycles is then partitioned by the homology relation, according to the hole they surround. Two cycles in the same equivalence class surround the same hole. The set of *p*-boundaries is the 0-equivalence class. For example, the cycle $z_2 = A_2 - A_3 + A_4$ is in the z_1 equivalence class because $z_1 = z_2 + \partial_2(F)$.

Homology groups, weak boundary. In any dimension *p*, the *homology group* H_p is defined as the group of the equivalent classes for the homology relation. It is exactly the quotient group of the *p*-cycles by the *p*-boundaries, $H_p = Z_p/B_p$. Homology groups are known to be topological invariants, meaning homeomorphic shapes have isomorphic homology groups.

For all *p*, there exists a finite number of elements of H_p from which we can deduce all H_p elements, thus H_p is called finitely generated. So, the group H_p verifies the fundamental theorem of finitely generated abelian groups [Mun84], and H_p is isomorphic to a direct sum:

$$\underbrace{\mathbb{Z} \oplus \dots \oplus \mathbb{Z}}_{\beta_p} \oplus \mathbb{Z}/t_1^p\mathbb{Z} \oplus \dots \oplus \mathbb{Z}/t_n^p\mathbb{Z}.$$

We denote by β_p the number of apparitions of \mathbb{Z} in this direct sum: it is the number of elements of H_p with infinite order and is called the *p-th Betti number*. The numbers t_1^p, \dots, t_n^p are called the *torsion coefficients* of H_p . To each group \mathbb{Z} of H_p is associated a set of *p*-dimensional homologous cycles: they surround the same *p*-dimensional topological hole and are not the boundary of any *p*+1-chain. It is the same for each group $\mathbb{Z}/t_i^p\mathbb{Z}$: the associated homologous cycles are not the boundary of any *p*+1-chain. However, when taken t_i^p times, they become the boundary of some *p*+1-chain. An example is the 1-cycle A_2 on Fig. 2, which becomes a boundary only when taken two times: $2A_2 = \partial(F_1 + F_2)$.

3 Related works

Kaczynski *et al.* [KMS98] proposed to compute a chain complex homology with a sequence of reductions. The idea is to derive a new object with less cells while preserving homology at each step of the transformation. To reduce the number of cells, one chooses two cells *a* and *b*, such that $\partial b = \lambda a + r$ and λ is invertible. These cells are then suppressed and the boundary homomorphism is updated. To ensure invertible coefficients, Kaczynski *et al.* choose them in a field. In this case, it can be verified that the reduction algorithm stops on a smallest chain complex with same homology where each cell is a cycle representing the homology class.

González-Díaz and Real [GDR03] recently proposed an algorithm to compute cohomology information on digital objects that are subsets of the 3D body-centered cubic grid. They first construct a simplicial complex with identical

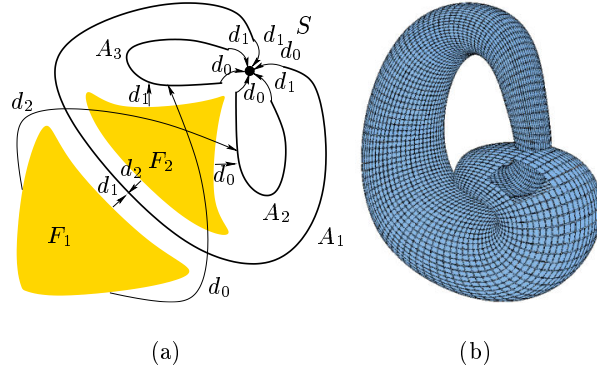


Fig. 2. Klein bottle. (a) Semi-simplicial and (b) geometric representation.

topology. After that the cohomology is obtained by the construction of a chain contraction in two passes: (i) a thinning that reduces the size of the data by simplicial collapses, (ii) an incremental algebraic thinning that progressively extracts the equivalence classes of the cohomology groups. A further computation provides the cohomology ring of the digital object, which appears to carry complementary topological information. All coefficients are in $\mathbb{Z}/2\mathbb{Z}$ (also a field).

The preceding approaches are interesting when dealing with embedded objects in 2D or 3D. Homology over a field is then enough to characterize shapes, since objects have no torsion. On the contrary, we choose a more generic approach, valid for arbitrary dimension and shapes. In the following section, we address the problem of computing the whole homology over the coefficient domain \mathbb{Z} (a ring, not a field).

4 Computation of homology groups and generators

In this section, we show how to compute the homology groups H_p from the boundary homomorphisms. First, the Betti number and torsion coefficients are deduced from the classical Smith Normal Form (SNF) of ∂_p and ∂_{p+1} . Then, we briefly explain why their SNF must be slightly modified to compute a set of generators of H_p . We finally discuss about implementation problems linked to that class of methods. We thus propose a new algorithm which benefits from improvements proposed by Dumas *et al.* [DSV01] and computes the Betti numbers, the torsion coefficients and a set of "moduli generators".

4.1 Homology groups via Smith Normal Form

Information on homology groups may be deduced from matrix representations of boundary homomorphisms. A natural basis of the group of p -chains of a chain

complex is the one made of all its p -simplices, i.e. K^p . In the following, the matrix E_{p+1} , called p -th incidence matrix, represents the homomorphism ∂_{p+1} relatively to the canonical bases K^p (rows) and K^{p+1} (columns). Each column in E_{p+1} is the boundary of one $p+1$ -simplex, decomposed on the base of p -simplices.

There exists bases in which any homomorphism has a very specific matrix form, the so-called *Smith Normal Form (SNF)*. It is a matrix full of 0's except for an upper left square submatrix which is diagonal with increasing coefficients: $\text{diag}(\lambda_1, \dots, \lambda_l)$ such that each λ_i is greater than 1 and divides each λ_j for $j > i$. The (λ_i) are called *invariant factors* of the homomorphism. Let D_{p+1} be the SNF of ∂_{p+1} with associated bases (e_k^{p+1}) and (f_k^p) :

$$\mathbf{D}_{p+1} = \begin{array}{c} \begin{array}{ccc} e_1^{p+1} & \dots & e_{\gamma_p}^{p+1} \\ \lambda_1^p & & 0 \\ & \ddots & \\ 0 & & \lambda_{\gamma_p}^p \end{array} & \begin{array}{ccc} e_{\gamma_{p+1}}^{p+1} & \dots & e_{n_{p+1}}^{p+1} \\ & & 0 \\ & & 0 \end{array} \\ \hline \begin{array}{ccc} & & 0 \\ & & 0 \end{array} \end{array} \begin{array}{c} f_1^p \\ \vdots \\ f_{\gamma_p}^p \\ f_{\gamma_{p+1}}^p \\ \vdots \\ f_{n_p}^p \end{array}$$

With these notations, it may be proved that:

1. $(e_{\gamma_{p+1}}^{p+1}, \dots, e_{n_{p+1}}^{p+1})$ is a basis of Z_{p+1} ,
2. $(\lambda_1^p f_1^p, \dots, \lambda_{\gamma_p}^p f_{\gamma_p}^p)$ is a basis of B_p ,
3. $(f_1^p, \dots, f_{\gamma_p}^p)$ is a basis of a group W_p , known as the group of *weak boundaries* ($W_p = \{c_p \in C_p / \exists \lambda \in \mathbb{Z}^*, \lambda c_p \in B_p\}$).

Moreover, the group H_p is isomorphic to the direct sum $Z_p/W_p \oplus W_p/B_p$ where Z_p/W_p is a free group and W_p/B_p is a torsion group. The torsion coefficients of H_p are exactly the invariant factors of ∂_{p+1} strictly greater than 1 (given by D_{p+1}). Furthermore, the Betti number of H_p is equal to $\text{rank}(Z_p) - \text{rank}(W_p)$. They are read respectively on D_p and D_{p+1} with $\text{rank}(Z_p) = n_p - \gamma_{p-1}$ and $\text{rank}(W_p) = \gamma_p$.

However all the generators of the homology groups cannot be deduced from the bases of the SNF. More precisely, we cannot determine the set of cycles which are not weak boundaries. To do it effectively, two successive boundary homomorphisms ∂_{p-1} and ∂_p must respectively share the same upper and lower bases (i.e. $(f_1^p, \dots, f_{n_p}^p) = (e_1^p, \dots, e_{n_p}^p)$). This is obviously not the case since $D_{p-1}D_p \neq 0$ (recall that $\partial_{p-1}\partial_p = 0$ in a free chain complex).

4.2 Generators with modified SNF

Cairn [Cai61] proved that it is possible to simultaneously choose bases for each group of p -chains such that the matrix N_p representing each boundary operator relatively to these bases is in a normal form quite similar to SNF. Moreover he

	$(p+1) - \text{Cycles}$		$\text{Weak Boundaries Antecedents}$				
	$a_1^{p+1} \dots a_{\gamma_{p+1}}^{p+1}$	$b_1^{p+1} \dots b_{\beta_{p+1}}^{p+1}$	$c_1^{p+1} \dots c_{\rho_p}^{p+1}$	$c_{\rho_p+1}^{p+1} \dots c_{\gamma_p}^{p+1}$			
a_1^p	0	0	$\lambda_{\gamma_p}^p$	0	0	<i>Weak Boundaries</i>	
\vdots				\ddots			
$a_{\rho_p}^p$			0	$\lambda_{\gamma_p - \rho_p + 1}^p$	1		
$a_{\rho_p+1}^p$	0	0	0	\ddots	0		
\vdots							
$a_{\gamma_p}^p$				0	1		
b_1^p	0	0	0	0	0	<i>Cycles but not Weak Boundaries</i>	
\vdots							
$b_{\beta_p}^p$							
c_1^p	0	0	0	0	0		
\vdots							
$c_{\gamma_{p-1}}^p$							

Table 1. Modified SNF of boundary homomorphism ∂_{p+1} .

explains how to deduce a set of generators of the homology group H_p directly from the matrix N_{p+1} . N_{p+1} is shown on Tab. 1. The number of invariant factors of ∂_{p+1} is γ_p and ρ_p of them are strictly greater than 1.

The set $\{b_1^p, \dots, b_{\beta_p}^p\}$ generates the free part of H^p : they are p -cycles when read as a column in N_p and they have no boundary antecedent when read as a row in N_{p+1} . The set $\{a_1^p, \dots, a_{\gamma_p}^p\}$ generates the torsion part of H^p : they are p -cycles when read as a column in N_p and they must be multiplied by the λ_i^p to have a boundary antecedent when read as a row in N_{p+1} .

Agoston [Ago76] proposed an algorithm to compute all matrices N_p and keep tracks of changes of bases. The idea is to compute successively all matrices N_p from 0 to the maximal index of the desired homology groups. Each homomorphism is successively expressed in four pairs of bases as in Tab. 2.

At the end of the whole computation, all the matrices N^p represent the homomorphisms ∂_p relatively to bases Γ^p such that $\Gamma^0 = V_0^{-1}K^0$, $\Gamma^1 = U_1V_1^{-1}K^1$, \dots , $\Gamma^{n-1} = U_{n-1}V_{n-1}^{-1}K^{n-1}$, $\Gamma^n = U_nK^n$.

4.3 Optimizations for effective computation

Algorithms for computing the SNF or the presented modified version are well known (e.g. see [Ago76, Mun84]). But major difficulties arise when trying to program them effectively. These problems are mainly linked to the high computational cost of the algorithms and to the appearance of very big integers during the process. The algorithm is namely valid as long as integer computations have an arbitrary precision. With standard 32 or 64 bits integers, the algorithm is no more accurate. This problem arises even in small chain complexes. Hafner *et al.*

Step	Bases $\llbracket \cdot \rrbracket$ and Matrix of ∂_p	Bases $\llbracket \cdot \rrbracket$ and Matrix of ∂_{p+1}
0. input from iteration p	$[(V_{p-1}U_{p-1}^{-1})^{-1}K^{p-1}] \llbracket [U_p K^p]$ (mSNF) $N_p = V_{p-1}U_{p-1}^{-1}E_pU_p$	
1. Incidence matrix of ∂_{p+1}		$[K^p] \llbracket [K^{p+1}]$ (incidence) E_{p+1}
2. Left-multiply E_{p+1} by U_p^{-1}		$[(U_p^{-1})^{-1}K^p] \llbracket [K^{p+1}]$ $E'_{p+1} = U_p^{-1}E_{p+1}$
3. Compute the mSNF N_{p+1} of ∂_{p+1} from E'_{p+1}		$[(V_pU_p^{-1})^{-1}K^p] \llbracket [U_{p+1}K^{p+1}]$ (mSNF) $N_{p+1} = V_pU_p^{-1}E_{p+1}U_{p+1}$
4. Right-multiply N_p by V_p^{-1}	$[(V_{p-1}U_{p-1}^{-1})^{-1}K^{p-1}] \llbracket [U_pV_p^{-1}K^p]$ $N_pV_p^{-1}$ (same as N_p)	

Table 2. Expression of the homomorphisms.

have exhibited a 10×10 incidence matrix, with no value greater than 10, that induces huge *intermediate* integer numbers in SNF computation.

Deterministic and stochastic algorithms have been proposed to tackle these difficulties. The best known deterministic algorithm has been proposed by Storjohann [Sto96]. Stochastic algorithms have for example been proposed by Giesbrecht *et al.* [Gie95]. They are generally more efficient than deterministic ones on sparse matrices, but are quite equivalent on dense matrices. They are however restricted to the SNF computation and do not extract generators.

As far as we know, only Agoston [Ago76] proposed an algorithm to compute all homology information (including generators), but its implementation does not address the difficulties mentioned above. We propose here an adaptation of a Gaussian elimination algorithm developed by Dumas *et al.* [DSV01], which was originally only dedicated to the computation of the SNF of unrestricted simplicial complex. We combine this work to the work of Agoston to compute all homology information of semi-simplicial sets: Betti number and torsion coefficients of all homology groups, sets of “moduli generators”. The main steps of the algorithm are described below. All operations made on the incidence matrix implies changes of bases that are stored in suitable matrices.

1. (*Prepare matrix for Dumas’s algorithm.*) The rows of the incidence matrix are ordered by increasing pivot,
2. (*Same as Dumas.*) The matrix is put in echelon form with as many pivots at 1 as possible by
 - first pass: only elementary row operations are applied,
 - second pass: all rows are reduced according to their gcd.
 - the matrix is now in triangular form: deduce submatrix determinant (which is also the product of the invariant factors).
 - All further integer operations are made modulo twice this determinant. It has indeed been proved (for example by Storjohann) that such a computation using an appropriate modulo preserves the homology information.
3. (*Different from Dumas.*) Elementary rows and columns operations are performed to compute the modified SNF on the submatrix with non-zero rows.

Changes of bases are traced. Agoston’s algorithm is used to compute the generators, which are “moduli generators” in the sense they have been partly computed with a modulo.

5 Experimentations

We validate our approach on shapes classically encountered when testing topological invariants. For each shape, Betti numbers and torsion coefficients are extracted from the mSNF. The generators are read in the matrices F^i . With this information, we are able to delineate each hole of the complex. It should be noted that we only present the generators for surfaces because the nature of 2-cycles on volumes is not well captured by 2D pictures.

Fig. 3 shows the shapes and the corresponding generators. Only the generators of the homology group H_1 are displayed since the others are trivial. For the torus, we have two cycles, one for each 1-dimensional hole ($H_1(K) \cong \mathbb{Z} \oplus \mathbb{Z}$). According to the topological nature of the Moebius strip (homotopic to a circle), we found only one cycle ($H_1(K) \cong \mathbb{Z}$). For the Klein bottle, two cycles are found, one for the free part of the homology and one for the torsion part ($H_1(K) \cong \mathbb{Z} \oplus \mathbb{Z}/2\mathbb{Z}$).

We have computed these generators using the previously described method with moduli. We observe that the “moduli” generators are homologous to those that would have been computed with arbitrary precision integer. On Fig. 3 each object has approximately 2000 triangles.

We guess that this property can be justified in a strict mathematical way but as far we know there is no indication to invalid or to confirm this property. Usual mathematical approaches are not really interested by the effective representation of the generators, which explains the lack of theoretical results on “moduli” generators.

To conclude, we have presented and implemented a technique to compute the whole homology of arbitrary finite shapes. We have addressed the problem of extracting generators of the homology groups with a modulo. For future works, we would like to exhibit the theoretical link between generator modulo and \mathbb{Z} -generators. We want also to study the simplicial homology for discrete objects. Cubes are indeed simplices and simplicial representations of discrete objects are thus more compact than simplicial ones.

References

- [ACZ04] M. Allili, D. Corriveau, and D. Ziou. Morse homology descriptor for shape characterization. In *Proc. ICPR 2004*, 2004.
- [ADFQ03] R. Ayala, E. Dominguez, A.R. Francès, and A. Quintero. Homotopy in digital spaces. *DAMATH: Discrete Appl. Math. and Combin. Oper. Research and Comput. Science*, 125, 2003.
- [Ago76] M. K. Agoston. *Algebraic Topology, a first course*. Marcel Dekker Ed., 1976.
- [Box99] L. Boxer. A classical construction for the digital fundamental group. *J. Math. Imaging Vision*, 10:51–62, 1999.

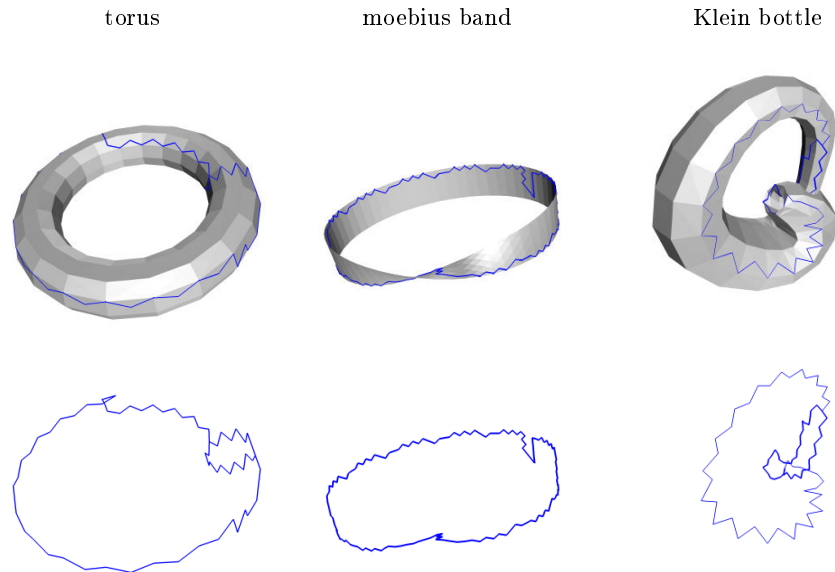


Fig. 3. Examples of homology generators on some classical surfaces.

- [Cai61] S. S. Cairns. *Introductory Topology*. Ronald Press Company, 1961.
- [Cur71] E. Curtis. Simplicial homotopy theory. *Adv. Math.*, 6:107–209, 1971.
- [DE95] 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.
- [DG03] P. Desbarats and S. Gueorguieva. Topological mainframe for numerical representations of objects. In *Proc. Int. Conf. Comput. Science and its Appl.*, volume 2668 of *LNCS*, pages 498–507. Springer, 2003.
- [DSV01] J. G. Dumas, B.D. Saunders, and G. Villard. On efficient sparse integer matrix smith normal form computations. *Journal of Symbolic Computation*, 2001.
- [GDR03] R. González-Díaz and P. Real. Toward digital cohomology. In G. Sanniti di Baja, S. Svensson, and I. Nyström, editors, *Proc. DGCI'2003*, volume 2886 of *LNCS*, pages 92–101. Springer, 2003.
- [Gie95] M. Giesbrecht. Nearly optimal-algorithms for canonical matrix-forms. *SIAM J. COMPUT.*, 24(5):948–969, OCT 1995.
- [Hat02] A. Hatcher. *Algebraic Topology*. Cambridge University Press, 2002. disponible sur <http://www.math.cornell.edu/~hatcher/AT/ATpage.html>.
- [KMS98] T. Kaczynski, M. Mrozek, and M. Slusarek. Homology computation by reduction of chain complexes. *Computers & Math. Appl.*, 34(4):59–70, 1998.
- [Kon89] T. Y. Kong. A digital fundamental group. *Computers and Graphics*, 13(2):159–166, 1989.
- [KR89] T. Y. Kong and A. Rosenfeld. Digital Topology: Introduction and Survey. *Comput. Vision, Graphics, and Image Processing*, 48(3):357–393, 1989.
- [LE93] P. Lienhardt and H. Elter. Different combinatorial models based on the map concept for the representation of different types of cellular complexes.

In *Proceedings of IFIP TC 5/WG II Work. Conf. on Geom. Modeling in Comp. Graphics*. Springer, 1993.

- [LL95] V. Lang and P. Lienhardt. Geometric modeling with simplicial sets. In *Proc. of Pacific Graphics'95*, Seoul, Korea, 1995.
- [Mal01] R. Malgouyres. Computing the fundamental group in digital spaces. *IJPRAI*, 15(7):1075–1088, 2001.
- [May67] J. P. May. *Simplicial Objects in Algebraic Topology*. Van Nostrand, 1967.
- [Mun84] J. R. Munkres. *Elements of algebraic topology*. Perseus Books, 1984.
- [Sto96] A. Storjohann. Near optimal algorithms for computing Smith normal forms of integer matrices. In Y. N. Lakshman, editor, *Proceedings of the 1996 International Symposium on Symbolic and Algebraic Computation*, pages 267–274. ACM, 1996.