

# Smith Normal Form & Homology Over PIDs

Or, How To Make Mathematica Do Your Topology Homework

Isaac Cheng

September 8, 2026

## Introduction

We begin by recalling the definition of a chain complex.

**Definition 1.** A **chain complex**  $C^\bullet$  is a sequence of  $R$ -modules  $\{C_i\}_{i \in \mathbb{Z}}$  and  $R$ -linear maps  $d_i : C_i \rightarrow C_{i-1}$  such that  $d_i \circ d_{i+1} = 0$  for all  $i$ .

$$C^\bullet : \dots \rightarrow C_{n+1} \rightarrow C_n \rightarrow C_{n-1} \rightarrow \dots$$

Chain complexes occur throughout mathematics, particularly in topology. Perhaps surprisingly, it turns out that the most valuable data of a chain complex is generally encoded in its *homology*, the quotients  $H_i(C^\bullet) = \ker d_i / \operatorname{im} d_{i+1}$  for every  $i \in \mathbb{Z}$ .<sup>1</sup> When  $C^\bullet$  is somehow derived from a mathematical object (for example, a CW complex), the homology of  $C^\bullet$  can give us useful and easily computable information about that object.

<sup>1</sup> There is a lot of beautiful intuition behind this definition, but that is a topic for another exposition.

**Example 2** (Cellular Homology). Given a CW-complex  $X$ , its **cellular chain complex** is the chain complex  $C^\bullet$  where  $C_n$  is the free abelian on the  $n$ -cells of  $X$ , and  $d_n : C_n \rightarrow C_{n-1}$  takes each cell to its boundary and extends by  $R$ -linearity to the rest of  $C_n$ . The **cellular homology groups**  $H_i^{CW}(X)$  are the homology groups of this complex.

Topology contains many other variations of homology, such as simplicial or singular homology. Crucially, though, the modules involved in topological homology theories are almost always free, and the ground rings are generally PIDs.<sup>2</sup> In many cases, the modules are also finitely generated. This exposition will describe an algorithmic way to compute  $H_i(C^\bullet)$  when all of these nice properties hold.

<sup>2</sup> These are related; generally one wants to work with projective modules, and projective modules over PIDs are free.

## Towards an Algorithm

We begin by computing an example in multiple ways, gradually building up towards full generality.

**Example 3** (Manual computation). Consider the following chain complex  $C^\bullet$ :

$$\begin{array}{ccccccc} 0 & \longrightarrow & C_3 & \xrightarrow{0} & C_2 & \xrightarrow{\alpha} & C_1 & \xrightarrow{\beta} & C_0 & \longrightarrow & 0 \\ & & \parallel & & \parallel & & \parallel & & \parallel & & \\ & & \mathbb{Z} & & \mathbb{Z}^3 & & \mathbb{Z}^4 & & \mathbb{Z}^2 & & \end{array}$$

where we write  $F_1, F_2, F_3$  for the generators of  $C_2$ ,  $a, b, c, d$  for the generators of  $C_1$ , and  $x, y$  for the generators of  $C_0$ . Define the maps as follows:

$$\begin{aligned}\alpha(F_1) &= a + b + c + d \\ \alpha(F_2) &= -a - b + c + d \\ \alpha(F_3) &= -a + b + c - d \\ \beta(a) &= \beta(c) = y - x \\ \beta(b) &= \beta(d) = x - y\end{aligned}$$

We will now compute  $H_1(C^\bullet) = \ker \beta / \operatorname{im} \alpha$ .<sup>3</sup>

Observe that  $\ker \beta = \langle a + b, a + d, c + d \rangle$ . Quotienting by  $-a - b + c + d$  says that  $a + b = c + d$ , reducing our generating set to  $\langle a + b, a + d \rangle$ . Quotienting by  $a + b + c + d = a + b + (a + b)$  tells us that  $2(a + b) = 0$ . Finally, quotienting by  $-a + b + c - d$  tells us that  $a + d = b + c = 0$ . In conjunction with the quotient by  $a + b + c + d$ , which tells us that  $a + d = -(b + c)$ , we find that  $2(a + d) = 0$ . Thus  $H_1(C^\bullet) = \mathbb{Z}/2\mathbb{Z} \oplus \mathbb{Z}/2\mathbb{Z}$ , generated by  $a + b$  and  $a + d$ .  $\square$

<sup>3</sup> This is Hatcher Exercise 2.2.11; the homology of a cube with opposite sides identified via a clockwise twist.

This was a lot of work, and it took a fair amount of ingenuity to reduce the equations into a clear solution. We're just simplifying systems of linear equations! Surely there is a faster way.

**Example 4** (Matrices and cokernels). As  $\alpha, \beta$  are homomorphisms of free  $\mathbb{Z}$ -modules, we may more succinctly describe them as matrices with  $\mathbb{Z}$ -coefficients. In the case

above,  $\alpha$  corresponds to  $A = \begin{bmatrix} 1 & -1 & -1 \\ 1 & -1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & -1 \end{bmatrix}$  and  $\beta$  to  $B = \begin{bmatrix} -1 & 1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix}$ .

How do we compute the homology this way? For sake of a simpler example, suppose that  $\beta$  is actually equal to 0. Then  $H_1(C^\bullet) = C_1 / \operatorname{im} \alpha$ , the *cokernel* of  $\alpha$ . To compute the cokernel of  $\alpha$ , it doesn't matter which bases we choose for  $C_2$  and  $C_1$ , so we might as well pick nice ones. By row and column reducing over  $\mathbb{Z}$ , we can turn  $A$  into its **Smith normal form**:

$$\begin{bmatrix} 1 & -1 & -1 \\ 1 & -1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & -1 \end{bmatrix} \mapsto \begin{bmatrix} 1 & -1 & -1 \\ 0 & 0 & 2 \\ 0 & 2 & 2 \\ 0 & 2 & 0 \end{bmatrix} \mapsto \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 2 \\ 0 & 2 & 2 \\ 0 & 2 & 0 \end{bmatrix} \mapsto \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 2 \\ 0 & 0 & 0 \\ 0 & 2 & 0 \end{bmatrix} \mapsto \begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \\ 0 & 0 & 0 \end{bmatrix}$$

When we quotient  $C_1$  by the image of this matrix, we eliminate the first basis element (in our new basis for  $C_1$ ) while giving the second and third basis elements 2-torsion. The fourth basis element is unaffected. Thus,  $C_1 / \operatorname{im} \alpha \cong \mathbb{Z} \oplus \mathbb{Z}/2\mathbb{Z} \oplus \mathbb{Z}/2\mathbb{Z}$ .

We must now account for the case when  $\beta$  is not fact 0. To do this though, it is helpful to introduce more formalism.

## Formalizing

Suppose  $C_2 \xrightarrow{\alpha} C_1 \xrightarrow{\beta} C_0$  is a chain complex of finitely generated free  $R$ -modules, where  $R$  is a PID. Let  $A, B$  be the matrices for  $\alpha, \beta$  with regard to the standard bases. Since  $C_1$  is free and  $R$  is a PID,  $\ker \beta \subseteq C_1$  is also free, and we may write  $C_1 = (\ker \beta) \oplus W$  for some subspace  $W \subseteq C_1$ ,<sup>4</sup>

Now, let  $\gamma : C^1 \rightarrow C^1$  be an invertible map that sends the standard basis of  $C^1$  to one that extends a basis for  $\ker \beta$ . Represent  $\gamma$  by a matrix  $Q$ ,<sup>5</sup> so the matrix of  $\alpha$  with regard to this new basis is  $QA$ .

Since  $\text{im } \alpha \subseteq \ker \beta$ , the matrix of  $\alpha$  with regard to this new basis (which is  $QA$ ) is identically zero on the rows corresponding to  $W$ .

$$QA = \left[ \begin{array}{c} \cdots \\ \mathbf{0} \end{array} \right] \begin{array}{l} \xleftarrow{\ker \beta} \\ \xleftarrow{W} \end{array}$$

Since we are interested only in  $\ker \beta / \text{im } \alpha$ , we may safely ignore the bottom rows (which are all 0). Instead, we wish to compute the cokernel of the map  $\alpha : C_2 \rightarrow \ker \beta$ , whose matrix representation is the remaining (top) rows of  $QA$ . As in Example 4, this can be done by reducing this matrix to Smith normal form (which is always possible over a PID) and analyzing the result. Observing that  $QA$  and  $A$  have the same Smith normal form, as  $Q$  is invertible, we are left with a single, clean formula.

**Theorem 5.** Let  $C^\bullet : C_2 \xrightarrow{\alpha} C_1 \xrightarrow{\beta} C_0$  be a chain complex of finitely generated free  $R$ -modules, where  $R$  is a PID. Let  $A$  be a matrix for  $\alpha$  corresponding to any bases.

Write  $A$  in Smith normal form, so it appears of the form

$$\left[ \begin{array}{cccccc|c} d_1 & 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & d_2 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & d_r & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & \cdots & 0 \end{array} \right] \begin{array}{l} \\ \\ \\ \mathbf{0} \\ \\ \\ \mathbf{0} \end{array},$$

where  $d_1, \dots, d_r \neq 0$ . We have  $H_1(C^\bullet) = \ker \beta / \text{im } \alpha = (\bigoplus_{i=1}^r R/d_i R) \oplus R^{\text{rk } C_1 - \text{rk } \beta - \text{rk } \alpha}$ .

*Proof.* Let  $Q$  be a change of basis matrix taking the basis of  $C^1$  used in  $A$  to one that extends a basis for  $\ker \beta$ . We observe that when converting  $QA$  to Smith normal form, the bottom zeroes are preserved, so the top  $\text{rk}(\ker \beta) = \text{rk } C_1 - \text{rk } \beta$  rows are the Smith normal form for the map  $\alpha : C_2 \rightarrow \ker \beta$ .

The torsion part of  $\ker \beta / \text{im } \alpha$  will be the direct sum of cyclic modules corresponding to quotients by the ideals generated by the diagonal entries of this top matrix. There will be  $\text{rk } \alpha$  of these torsion modules, which leaves  $\text{rk } C_1 - \text{rk } \beta - \text{rk } \alpha$  copies of  $R$ . Since  $Q$  is invertible, the Smith normal form of  $QA$  is the same as the Smith normal form of  $A$ , so the result follows.  $\square$

<sup>4</sup> We can just use  $W = (\ker \beta)^\perp$  using the standard inner product.

<sup>5</sup> Inner products are useful for constructing this map, but existence is automatic.

**Example 6** (Using our new tools). Return to the situation from Example 3, where  $\alpha$

corresponds to  $\begin{bmatrix} 1 & -1 & -1 \\ 1 & -1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & -1 \end{bmatrix}$  and  $\beta$  to  $\begin{bmatrix} -1 & 1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix}$ .

Disappointingly, WolframAlpha cannot compute Smith normal form, but Math-

ematica can.<sup>6</sup>  $A$  reduces to  $\begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \\ 0 & 0 & 0 \end{bmatrix}$ ,<sup>7</sup> and we see that  $\text{rk } \beta = 1$ .<sup>8</sup> Thus, we have

$H_1(C^\bullet) = 0 \oplus \mathbb{Z}/2\mathbb{Z} \oplus \mathbb{Z}/2\mathbb{Z} \oplus \mathbb{Z}^{4-3-1} = \mathbb{Z}/2\mathbb{Z} \oplus \mathbb{Z}/2\mathbb{Z}$ , as computed in Example 3.

<sup>6</sup> The algorithm is also not too difficult to do by hand.

<sup>7</sup> `SmithDecomposition[{{1, -1, -1}, {1, -1, 1}, {1, 1, 1}, {1, 1, -1}}]`. The second matrix in the list is the reduced form.

<sup>8</sup> `MatrixRank[{{-1, 1, -1, 1}, {1, -1, 1, -1}}]`. WolframAlpha can also do this one, or we could just observe that the second row is a multiple of the first.

## References

- [Alu] Paolo Aluffi. *Algebra: Chapter 0*. The sixth chapter, on linear algebra, was my reference for Smith normal form and other matrix computations.
- [Bun] Eric Bunch. *Calculating Homology of a Simplicial Complex*. [https://eric-bunch.github.io/blog/calculating\\_homology\\_of\\_simplicial\\_complex](https://eric-bunch.github.io/blog/calculating_homology_of_simplicial_complex). This article was the source for my motivation in writing this exposition.