

A Coq formalization of finitely presented modules

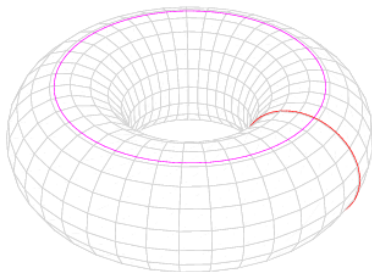
Cyril Cohen and Anders Mörtberg

University of Gothenburg
(`cyril.cohen|anders.mortberg`)@gu.se

November 22, 2013

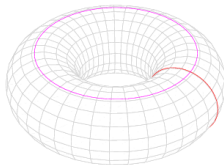
What is homology?

Homology is a rigorous mathematical method for defining and categorizing holes in a shape



Homological algebra: Linear algebra over rings

Homology of the torus



$$\begin{aligned}H_0(\mathbb{T}) &= R \\H_1(\mathbb{T}) &= R \oplus R \\H_2(\mathbb{T}) &= R\end{aligned}$$

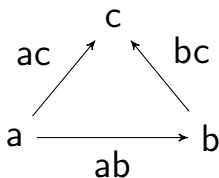
Definition

$$\cdots \longrightarrow C_{i+1} \xrightarrow{d_{i+1}} C_i \xrightarrow{d_i} C_{i-1} \longrightarrow \cdots$$

$$H_i = \text{Ker}(d_i) / \text{Im}(d_{i+1})$$

- C_i is the i -dimensional “content”,
- d_i maps the i -dimensional “content” on its boundary,
- H_i is the i^{th} -homology group.

The triangle



$$0 \xrightarrow[d_2]{0} \mathbb{Z}[ab, ac, bc] \xrightarrow[d_1]{\begin{pmatrix} -1 & 1 & 0 \\ -1 & 0 & 1 \\ 0 & -1 & 1 \end{pmatrix}} \mathbb{Z}[a, b, c] \xrightarrow[d_0]{0} 0$$

$$H_0 = \text{Ker}(d_0)/\text{Im}(d_1) = \mathbb{Z}[a, b, c]/\text{Im}(d_1) \simeq \mathbb{Z}$$

$$H_1 = \text{Ker}(d_1)/\text{Im}(d_2) \simeq \text{Ker}(d_1) \simeq \mathbb{Z}$$

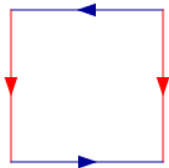
\mathbb{Z} -homology vs. \mathbb{F}_2 -homology

For the Torus it does not matter if computations are done in \mathbb{Z} or \mathbb{F}_2 :

	\mathbb{Z}	\mathbb{F}_2
$H_0(\mathbb{T})$	\mathbb{Z}	\mathbb{F}_2
$H_1(\mathbb{T})$	$\mathbb{Z} \oplus \mathbb{Z}$	$\mathbb{F}_2 \oplus \mathbb{F}_2$
$H_2(\mathbb{T})$	\mathbb{Z}	\mathbb{F}_2

This is not always true, e.g. Klein bottle.

Klein bottle



Homology groups:

	\mathbb{Z}	\mathbb{F}_2
$H_0(\mathbb{K})$	\mathbb{Z}	\mathbb{F}_2
$H_1(\mathbb{K})$	$\mathbb{Z} \oplus \mathbb{Z}_2$	$\mathbb{F}_2 \oplus \mathbb{F}_2$
$H_2(\mathbb{K})$	0	\mathbb{F}_2

\mathbb{Z} -homology vs. \mathbb{F}_2 -homology

With \mathbb{F}_2 -homology \mathbb{T} and \mathbb{K} are indistinguishable!

\mathbb{Z} -homology is more informative.

Goal: Compute R -homology with Coq.

Definition

$$\cdots \longrightarrow C_{i+1} \xrightarrow{d_{i+1}} C_i \xrightarrow{d_i} C_{i-1} \longrightarrow \cdots$$

$$H_i = \text{Ker}(d_i)/\text{Im}(d_{i+1})$$

Required features:

- **finitely presented modules**: C_i ,
- **morphisms** of f.p. modules: d_i ,
- **kernel** and **image submodules**,
- **quotient** of a submodule by another.

Finitely presented modules

We take an approach close to SSReflect
matrix/mxalgebra/vector libraries.

Finitely presented modules

We take an approach close to SSReflect
matrix/mxalgebra/vector libraries.

A finitely presented modules is characterized by:

- its generators (e.g. e_0, e_1),
- and relations between them (e.g. $2e_0 = 0$).

Finitely presented modules

We take an approach close to SSReflect matrix/mxalgebra/vector libraries.

A finitely presented modules is characterized by:

- its generators (e.g. e_0, e_1),
- and relations between them (e.g. $2e_0 = 0$).

We represent:

- generators by a number n (e.g. 2),
- relations by a matrix with n columns (e.g. $\begin{pmatrix} 2 & 0 \end{pmatrix}$).

Module presentation

$$R^{m_0} \xrightarrow{M} R^m \xrightarrow{1} \mathcal{M} \xrightarrow{0} 0$$

A is zero in the module \mathcal{M}

$$\Leftrightarrow \exists B, BM = A$$

$$\Leftrightarrow M \mid A$$

Morphisms

$$R^{m_0} \xrightarrow{M} R^m \xrightarrow{1} \mathcal{M} \xrightarrow{0} 0$$

$$R^{n_0} \xrightarrow{N} R^n \xrightarrow{1} \mathcal{N} \xrightarrow{0} 0$$

Morphisms

Morphisms must preserve relations:

$$\begin{array}{ccccccccc} R^{m_0} & \xrightarrow{M} & R^m & \xrightarrow{1} & \mathcal{M} & \xrightarrow{0} & 0 \\ X \downarrow & & \downarrow Y & & \downarrow \varphi & & \\ R^{n_0} & \xrightarrow{N} & R^n & \xrightarrow{1} & \mathcal{N} & \xrightarrow{0} & 0 \end{array}$$

$$\begin{aligned} & \exists X, MY = XN \\ \Leftrightarrow & N \mid MY \end{aligned}$$

Morphisms

Morphisms must preserve relations:

$$\begin{array}{ccccccc} R^{m_0} & \xrightarrow{M} & R^m & \xrightarrow{1} & \mathcal{M} & \xrightarrow{0} & 0 \\ X \downarrow & & \downarrow Y & & \downarrow \varphi & & \\ R^{n_0} & \xrightarrow{N} & R^n & \xrightarrow{1} & \mathcal{N} & \xrightarrow{0} & 0 \end{array}$$

```
Record morphism_of m0 m n0 n (M : 'M[R]_(m0, m)) (N : 'M[R]_(n0, n)) :=  
  Morphism { matrix_of_morphism : 'M[R]_(m, n);  
    _ : (N %| M *m matrix_of_morphism)%MP }.
```


Submodules

Submodules are just injective morphisms:

$$\begin{array}{ccccccccc} R^{m_0} & \xrightarrow{M} & R^m & \xrightarrow{1} & \mathcal{M} & \xrightarrow{0} & 0 \\ X \downarrow & & \downarrow Y & & \downarrow \iota & & \\ R^{n_0} & \xrightarrow{N} & R^n & \xrightarrow{1} & \mathcal{N} & \xrightarrow{0} & 0 \end{array}$$

Submodules

Submodules are just injective morphisms:

$$\begin{array}{ccccccc} R^{m_0} & \xrightarrow{M} & R^m & \xrightarrow{1} & \mathcal{M} & \xrightarrow{0} & 0 \\ X \downarrow & \circlearrowleft & \downarrow Y & & \downarrow \iota & & \\ R^{n_0} & \xrightarrow{N} & R^n & \xrightarrow{1} & \mathcal{N} & \xrightarrow{0} & 0 \end{array}$$

But we define them just as matrices:

```
Record submodule_of n0 n (N : 'M[R]_(n0, n)) :=  
  Submodule {dim_of_submodule : nat;  
            matrix_of_submodule : 'M[R]_(dim_of_submodule, n)}.
```

because we can find a M which turns it into an injective morphism. (But what is injectivity?)

Injectivity, surjectivity

$$\begin{array}{ccccccccc} R^{m_0} & \xrightarrow{M} & R^m & \xrightarrow{1} & \mathcal{M} & \xrightarrow{0} & 0 \\ X \downarrow & & \circlearrowleft & & \downarrow \varphi & & \\ R^{n_0} & \xrightarrow{N} & R^n & \xrightarrow{1} & \mathcal{N} & \xrightarrow{0} & 0 \end{array}$$

$\varphi : \mathcal{M} \rightarrow \mathcal{N}$ is

- injective
if the kernel of L modulo \mathcal{N} is zero modulo \mathcal{M}
- surjective if its cokernel is trivial (its image is full).

Injectivity, surjectivity

$$\begin{array}{ccccccc} R^{m_0} & \xrightarrow{M} & R^m & \xrightarrow{1} & \mathcal{M} & \xrightarrow{0} & 0 \\ X \downarrow & & \circlearrowleft & & \downarrow \varphi & & \\ R^{n_0} & \xrightarrow{N} & R^n & \xrightarrow{1} & \mathcal{N} & \xrightarrow{0} & 0 \end{array}$$

$\varphi : \mathcal{M} \rightarrow \mathcal{N}$ is

- injective
if the **kernel of L modulo \mathcal{N}** is zero modulo \mathcal{M}
- surjective if its cokernel is trivial (its image is full).

Kernel is a primitive

Ker is a primitive giving the space of solutions (e.g. obtained from Smith Normal Form).

$$\begin{aligned} XN = 0 &\Leftrightarrow \exists Z, Z \cdot \text{Ker}(N) = X \\ &\Leftrightarrow \text{Ker}(N) \mid X \end{aligned}$$

But finding the space of solutions of

$$XN = 0 \text{ in } \mathcal{M}$$

requires a “kernel modulo”.

Kernel modulo

$$\begin{aligned} XN = 0 \text{ in } \mathcal{M} &\Leftrightarrow \exists Y, (Y \ X) \cdot \begin{pmatrix} M \\ N \end{pmatrix} = 0 \\ &\Leftrightarrow \exists Y \ Z, Z \cdot \text{Ker} \begin{pmatrix} M \\ N \end{pmatrix} = (Y \ X) \\ &\Leftrightarrow \exists Y \ Z, Z \cdot (K_0 \ K_1) = (Y \ X) \\ &\Leftrightarrow \exists Y \ Z, (Z \cdot K_0 \ Z \cdot K_1) = (Y \ X) \\ &\Leftrightarrow \exists Z, Z \cdot K_1 = X \\ &\Leftrightarrow K_1 \mid X \end{aligned}$$

where $(K_0 \ K_1) = \text{Ker} \begin{pmatrix} M \\ N \end{pmatrix}$.

$\ker_M(N) := K_1$ is the *kernel of N modulo M* .

Fundamental property for kernel modulo

$$M \mid XN \Leftrightarrow \ker_M(N) \mid X$$

From kernel modulo, we have:

- the presentation matrix (pres) of a submodule of \mathcal{N}
- \Rightarrow promotion of a submodule to a module \mathcal{M} ,
- \Rightarrow this inclusion $\mathcal{M} \rightarrow \mathcal{N}$ is injective,
- the kernel of a morphism as a submodule (the image is trivial).

Kernel of a morphism

$$\begin{array}{ccccccc}
 R^{k_0} & \xrightarrow{\ker_M(\ker_N(Y))} & R^k & \xrightarrow{1} & \mathcal{K} & \xrightarrow{0} & 0 \\
 \downarrow & \circlearrowleft & \downarrow \ker_N(Y) & & \downarrow \text{kerm}(\varphi) & & \\
 R^{m_0} & \xrightarrow{M} & R^m & \xrightarrow{1} & \mathcal{M} & \xrightarrow{0} & 0 \\
 X \downarrow & \circlearrowleft & \downarrow Y & & \downarrow \varphi & & \\
 R^{n_0} & \xrightarrow{N} & R^n & \xrightarrow{1} & \mathcal{N} & \xrightarrow{0} & 0
 \end{array}$$

Definition `kerm := Submodule M (N.-ker (matrix_of_morphism phi)).`

Quotienting

$$\begin{array}{ccccccc}
 R^{m_0} & \xrightarrow{M} & R^m & \xrightarrow{1} & \mathcal{M} & \xrightarrow{0} & 0 \\
 X \downarrow & \circlearrowleft & \downarrow Y & & \downarrow \varphi & & \\
 R^{n_0} & \xrightarrow{N} & R^n & \xrightarrow{1} & \mathcal{N} & \xrightarrow{0} & 0 \\
 (1 \ 0) \downarrow & \circlearrowleft & \downarrow 1 & & \downarrow \text{cokernel}(\varphi) & & \\
 R^{q_0} & \xrightarrow{\begin{pmatrix} N \\ Y \end{pmatrix}} & R^q & \xrightarrow{1} & \mathcal{Q} & \xrightarrow{0} & 0
 \end{array}$$

(** Quotienting is just matrix stacking *)

Definition `quot_by (L : Sub N) := col_mx N (incl L)`.

Sharpening and quotienting

If $\varphi : M \rightarrow N$ and $\text{Image}(\varphi) \subset L$ we factor

$$\varphi = \text{sharpen}_L(\varphi) \cdot \text{incl}(L)$$

If $L \subset N$ are submodules of M , we factor

$$\text{incl}(L) = \text{incl}_N(L) \cdot \text{incl}(N)$$

Notation " $N \% L$ " := `(quot_by (image (N.-incl L)))`.

Definition of an homology module

Variables $m0\ m1\ n0\ n1\ p0\ p1 : \text{nat}$.

Variables $(M : 'M[R]_(m0,m1))\ (N : 'M[R]_(n0,n1))\ (P : 'M[R]_(p0,p1))$.

Variables $(\text{phi} : 'Mor(M, N))\ (\text{psi} : 'Mor(N, P))$.

Lemma $\text{mulm_eq0} : (P \%| \text{phi} *m \text{psi}) = (\text{image phi} \leq \text{ker psi})$.

Proof.

rewrite /submodule imageK -dvd_ker ker_modE.

apply/idP/idP=> [? | /(dvdmx_trans _)-> //].

by rewrite (dvdmx_trans (dvd_quot_mx_incl _)).

by rewrite dvd_mx_col dvdmx_refl andbT dvd_ker dvdmx_morphism.

Qed.

Hypothesis $\text{mul_phi_psi} : P \%| \text{phi} *m \text{psi}$.



Lemma $\text{imphi_kerpsi} : \text{image phi} \leq \text{ker psi}$.

Proof. **by rewrite** -mulm_eq0. **Qed.**

Definition $\text{homology} := \text{ker psi} \% / \text{image phi}$.

Related work

Inspired by [BR08] and [Gon11]:

-  MOHAMED BARAKAT and DANIEL ROBERTZ, *homalg – a meta-package for homological algebra*, Journal of Algebra and Its Applications **07** (2008), no. 03, 299–317.
-  Georges Gonthier, *Point-free, set-free concrete linear algebra*, ITP, 2011, pp. 103–118.

(and thanks to wikipedia.org for the torus and Klein bottle pictures)

Future work and conclusions

Future work:

- More functors: Hom, cohomology, Ext, Tor...
- Quotienting by equality modulo in a module.
- Quotienting by module isomorphism (requires S.N.F. or univalence).
- Free resolutions.
- Abelian categories.

Conclusions:

- Finitely presented modules provides a nice setting for formalizing constructive module theory.
- Everything is a matrix \Rightarrow CoqEAL ready.
- Everything is based on system solving.

Thank you!

