A Coq Formalization of Finitely Presented Modules

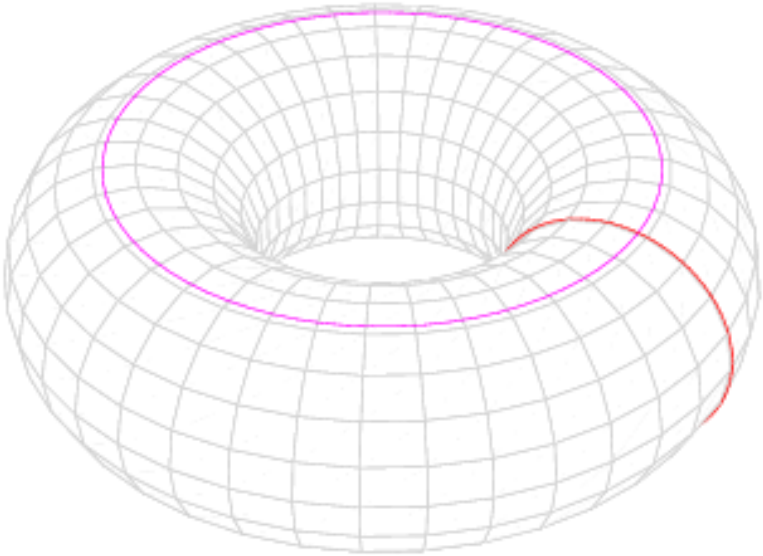
Cyril Cohen and **Anders Mörtberg**

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

July 16, 2014

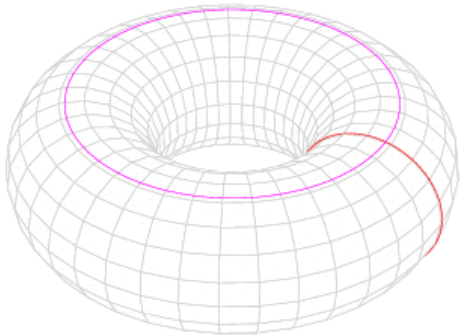
What is homology?

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



Homological algebra: Linear algebra over rings

Homology of the torus



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

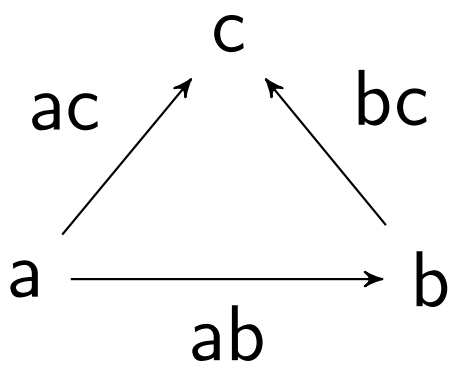
Definition of homology

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

$$H_i = \ker(d_i)/\mathrm{im}(d_{i+1})$$

- 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



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

$$\begin{aligned} H_0 &= \ker(d_0)/\mathrm{im}(d_1) &&= \mathbb{Z}[a, b, c]/\mathrm{im}(d_1) &&\simeq \mathbb{Z} \\ H_1 &= \ker(d_1)/\mathrm{im}(d_2) &&\simeq \ker(d_1) &&\simeq \mathbb{Z} \end{aligned}$$

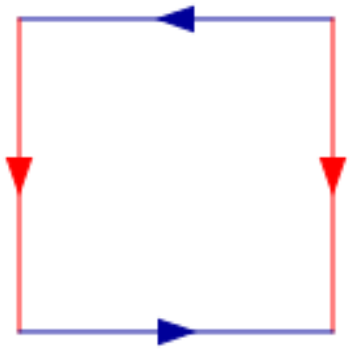
$\mathbb{Z}$-homology vs. $\mathbb{F}_2$-homology

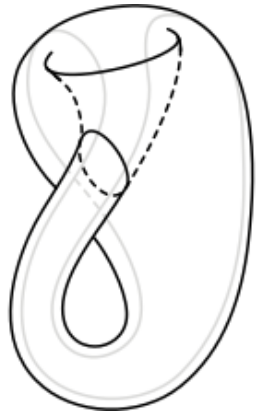
For the Torus it does not matter if computations are done in $\mathbb{Z}$ or $\mathbb{F}_2$:

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

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 of homology

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

$$H_i = \ker(d_i)/\mathrm{im}(d_{i+1})$$

Required features:

- R-modules: C_i,
- morphisms of R-modules: d_i,
- kernel and image,
- quotient of a submodule by another.

Finitely presented modules

We take an approach similar to the one in the SSReflect library where finite dimensional vector spaces are represented using matrices and all subspace operations are defined from Gaussian elimination

- Finite dimensional vector spaces $\Longrightarrow$ Finitely presented modules
- Gaussian elimination $\Longrightarrow$ Coherent and strongly discrete rings

Finitely presented modules

An R-module is **finitely presented** if it can be represented by:

- a finite number of generators (*e.g.* e_0, e_1),
- and relations between these (*e.g.* $2e_1 = 0$).

That is, it can be represented by an exact sequence:

$$R^{m_1} \xrightarrow{M} R^{m_0} \longrightarrow \mathcal{M} \longrightarrow 0$$

Finitely presented modules

The $\mathbb{Z}$-module $\mathbb{Z} \oplus \mathbb{Z}/2\mathbb{Z}$ is given by the presentation:

$$\mathbb{Z} \xrightarrow{\begin{pmatrix} 0 & 2 \end{pmatrix}} \mathbb{Z}^2 \longrightarrow \mathbb{Z} \oplus \mathbb{Z}/2\mathbb{Z} \longrightarrow 0$$

as if $\mathbb{Z} \oplus \mathbb{Z}/2\mathbb{Z}$ is generated by (e_0, e_1) there is one relation, namely $0e_0 + 2e_1 = 0$.

Coherent and strongly discrete rings

To convieniently represent morphisms and compute their kernel the underlying ring needs to be:

- Coherent: it is possible to compute generators of the kernel of any matrix
- Strongly discrete: membership in finitely generated ideals is decidable

Coherent and strongly discrete rings

If R is both coherent and strongly discrete we can solve arbitrary systems $XM = A$, we write:

$$M \mid A \Leftrightarrow \exists X.XM = A$$

Examples: fields (Gaussian elimination), $\mathbb{Z}$ (Smith normal form), $k[x_1, \ldots, x_n]$ (Gröbner bases)...

Coherent and strongly discrete rings

This gives a means for testing if a matrix is zero modulo a set of relations:

$$\begin{aligned} & A \text{ is zero in the module } \mathcal{M} \\ \Leftrightarrow \quad & \exists X,\ XM = A \\ \Leftrightarrow \quad & M \mid A \end{aligned}$$

and we get a nice divisibility theory for matrices:

```
Lemma dvdmxD m n k (M : 'M[R]_(m,n))
  (N K : 'M[R]_(k,n)) :
  M %| N -> M %| K -> M %| N + K.
```

Finitely presented modules: morphisms

A morphism between finitely presented R-modules is given by:

$$\begin{array}{ccccccc} R^{m_1} & \xrightarrow{M} & R^{m_0} & \longrightarrow & \mathcal{M} & \longrightarrow & 0 \\ \downarrow{\scriptstyle Y} & & \downarrow{\scriptstyle X} & & \downarrow{\scriptstyle \varphi} & & \\ R^{n_1} & \xrightarrow{N} & R^{n_0} & \longrightarrow & \mathcal{N} & \longrightarrow & 0 \end{array}$$

In our setting φ may be represented by X and a proof that $N \mid MX$

Finitely presented modules: morphisms

```
Record fpmodule := FPModule {
  nbrel : nat;
  nbgen : nat;
  pres : 'M[R]_(nbrel, nbgen)
}.

Record morphism_of (M N : fpmodule) :=
  Morphism { X : 'M[R]_(nbgen M,nbgen N);
             _ : pres N %| pres M *m X }.
```

Kernel modulo

To compute the kernel of a morphism we need to compute the kernel of a matrix modulo a set of relations.

$$XM = 0 \text{ in } \mathcal{N} \Leftrightarrow \quad \exists Y,\ XM + YN = 0$$
$$\Leftrightarrow \quad \exists Y,\ \begin{pmatrix} X & Y \end{pmatrix} \begin{pmatrix} M \\ N \end{pmatrix} = 0$$

Solving this amounts to computing $\ker \begin{pmatrix} M \\ N \end{pmatrix}$

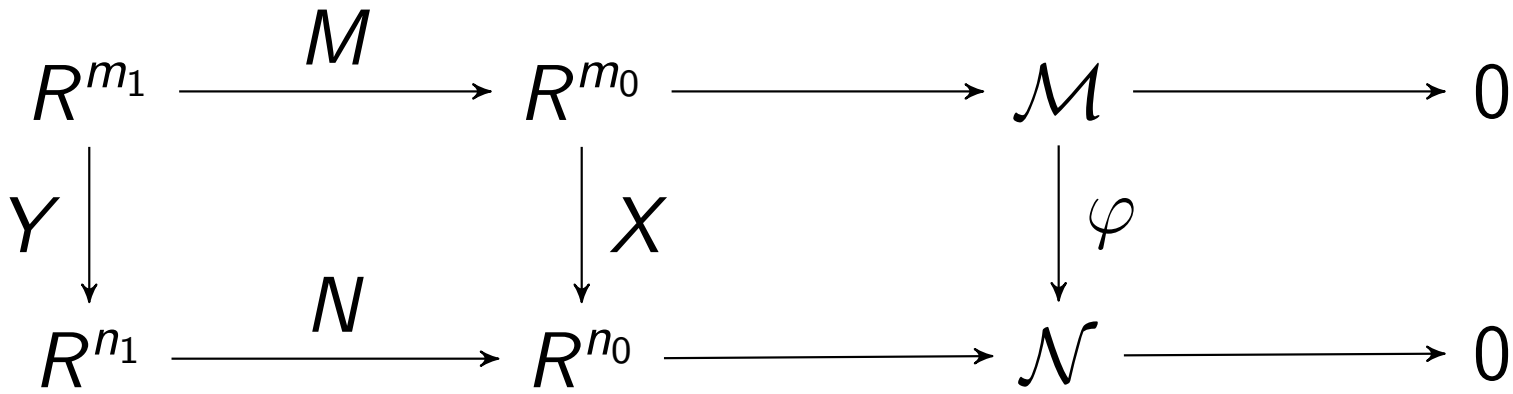
Kernel of a morphism

The kernel of a morphism $\varphi : \mathcal{M} \to \mathcal{N}$ is given by *two* things:

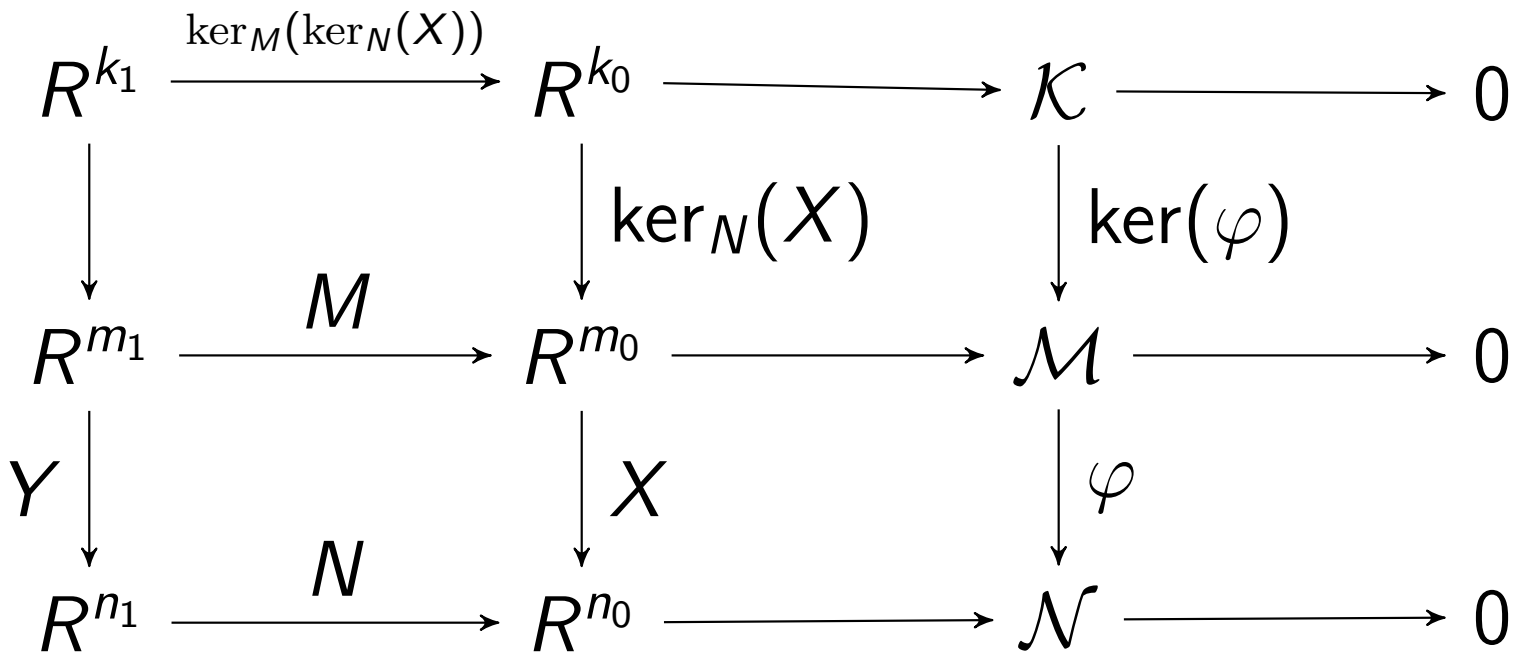
1. A finitely presented module $\mathcal{K}$
2. An injection $\ker(\varphi) : \mathcal{K} \to \mathcal{M}$

Both of these are computed by the "kernel modulo" operation

Kernel of a morphism



Kernel of a morphism



Definition of homology in Coq

Using our formalization we can write:

```
Variables (M N P : fpmodule R)
  (phi : 'Mor(M, N)) (psi : 'Mor(N, P)).

Hypothesis mul_phi_psi : phi ** psi %= 0.

Definition homology := kernel psi %/ phi.
```

Abelian categories

How do we know that our definitions are correct?

Abelian categories

How do we know that our definitions are correct?

Abelian categories: Introduced by Grothendieck to describe categories suitable for developing homological algebra

Goal: Prove that our implementation of finitely presented modules form an abelian category

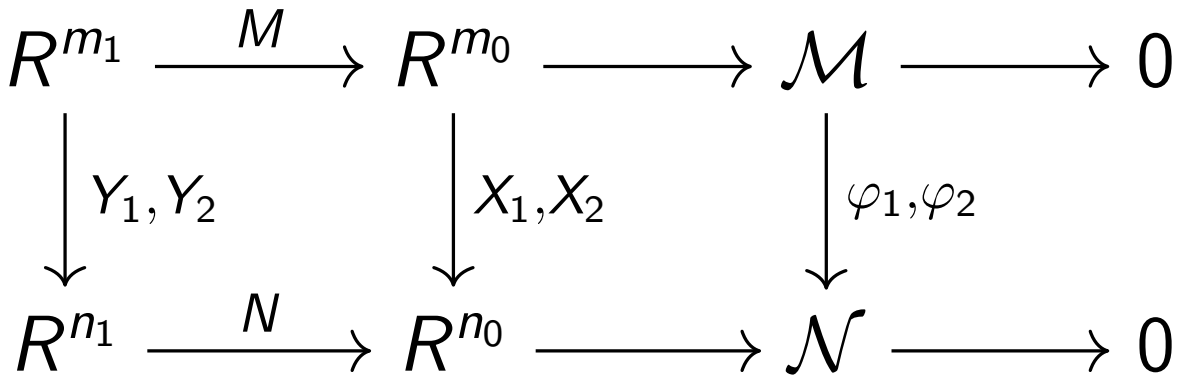
Abelian categories

A category is **abelian** if:

- the hom-sets form abelian groups ($+$, $-$ and 0 for morphisms),
- it has (finite) sums and products,
- any morphism has a kernel and cokernel, and
- all monomorphisms and epimorphisms are normal.

Addition of morphisms

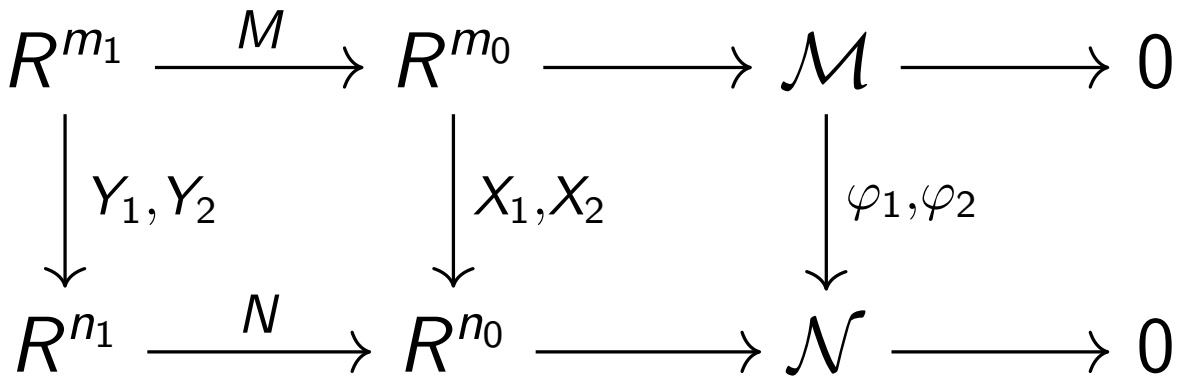
Given



How do we form $\varphi_1 + \varphi_2$?

Addition of morphisms

Given

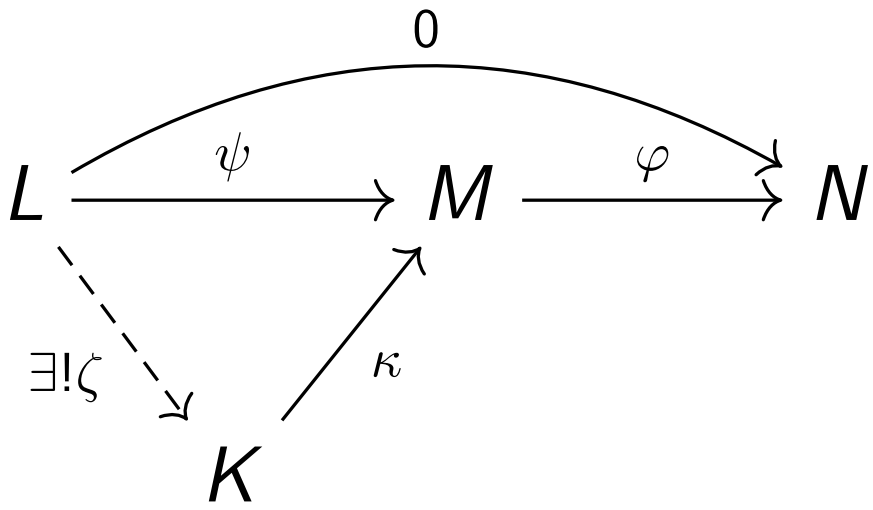


How do we form $\varphi_1 + \varphi_2$?

Recall that $N \mid MX_1$ and $N \mid MX_2$. So by the divisibility theory for matrices we get: $N \mid M(X_1 + X_2)$

Kernels of morphisms

A morphism $\varphi : M \to N$ has a kernel $\kappa : K \to M$ if $\kappa\varphi = 0$ and for all $\psi : L \to M$ with $\psi\varphi = 0$ there exists $\zeta : L \to K$ such that the following diagram commutes:



Kernels of morphisms

```
Definition is_kernel (M N K : fpmodule R)
  (phi : 'Mor(M,N)) (k : 'Mor(K,M)) :=
  (k ** phi %= 0) *
  forall L (psi : 'Mor(L,M)),
    reflect (exists Y, Y ** k %= psi)
            (psi ** phi %= 0).

Lemma kernelP M N (phi : 'Mor(M,N)) :
  is_kernel phi (kernel phi).
```

Monomorphisms are normal

As we can subtract morphisms we say that a morphism φ is **monomorphism** if whenever $\psi\varphi = 0$ then $\psi = 0$

```
Definition is_mono M N (phi : 'Mor(M,N)) :=
  forall L (psi : 'Mor(L,M)),
    psi ** phi %= 0 -> psi %= 0.
```

Monomorphisms are normal

A monomorphism is **normal** if it is a kernel of its cokernel

```
Lemma mono_ker M N (phi : 'Mono(M,N)) :
  is_kernel (coker phi) phi.
```

Hence we get that the category of finitely presented modules over coherent strongly discrete rings is abelian

Conclusions

- Finitely presented modules provides a nice setting for formalizing constructive module theory
- Kernel modulo and matrix division gives short and simple proofs
- Finitely presented modules is a good setting for developing constructive homological algebra

Future work

- Develop the theory of abelian categories
- Formalize graded objects like graded modules and chain complexes
- More functors: resolutions, Hom, cohomology, Ext, Tor...
- Quotient by morphism equality
- Quotient by module isomorphism (requires SNF or HoTT)

Thank you!