Interoperating between Computer Algebra Systems: Computing Homology of Groups with Kenzo and GAP

Ana Romero, Graham Ellis, and Julio Rubio

Universidad de La Rioja, Spain - National University of Ireland, Galway

Seoul, Korea, July 2009

Homology of groups: introduction

Homology of groups: introduction

Given a group G, $H_i(G)$?

Homology of groups: introduction

Given a group G, $H_i(G)$?

Several constructions:

Homology of groups: introduction

Given a group G, $H_i(G)$?

Several constructions:

- Topology:

Homology of groups: introduction

Given a group G, $H_i(G)$?

Several constructions:

- Topology: we identify $H_i(G)$ with the homology of a canonical topological space $K(G,1)$ associated with G

Homology of groups: introduction

Given a group G, $H_i(G)$?

Several constructions:

- Topology: we identify $H_i(G)$ with the homology of a canonical topological space $K(G,1)$ associated with G, with $\pi_1(K) = G$ and $\pi_i(K) = 0$ for all $i > 1$.

Homology of groups: introduction

Given a group G, $H_i(G)$?

Several constructions:

- Topology: we identify $H_i(G)$ with the homology of a canonical topological space $K(G,1)$ associated with G, with $\pi_1(K) = G$ and $\pi_i(K) = 0$ for all $i > 1$.
- Algebra:

Homology of groups: introduction

Given a group G, $H_i(G)$?

Several constructions:

- Topology: we identify $H_i(G)$ with the homology of a canonical topological space $K(G,1)$ associated with G, with $\pi_1(K) = G$ and $\pi_i(K) = 0$ for all $i > 1$.
- Algebra: $H_i(G)$ are computed by means of *resolutions*.

Homology of groups: Symbolic Computation Systems

Homology of groups: Symbolic Computation Systems

- GAP: Computational Group Theory.

Homology of groups: Symbolic Computation Systems

- GAP: Computational Group Theory.
    - Package HAP: Homological Algebra library, focused on cohomology of groups, making use of resolutions.

Homology of groups: Symbolic Computation Systems

- GAP: Computational Group Theory.
  - Package HAP: Homological Algebra library, focused on cohomology of groups, making use of resolutions.
- Kenzo: Constructive Algebraic Topology, makes use of the effective homology theory to determine homology groups of *topological spaces*.

Homology of groups: Symbolic Computation Systems

- GAP: Computational Group Theory.
    - Package HAP: Homological Algebra library, focused on cohomology of groups, making use of resolutions.
- Kenzo: Constructive Algebraic Topology, makes use of the effective homology theory to determine homology groups of *topological spaces*. It does not know what a resolution is, but it implements the spaces $K(G,1)$ for some groups G.

Homology of groups: Symbolic Computation Systems

- GAP: Computational Group Theory.
  - Package HAP: Homological Algebra library, focused on cohomology of groups, making use of resolutions.
- Kenzo: Constructive Algebraic Topology, makes use of the effective homology theory to determine homology groups of *topological spaces*. It does not know what a resolution is, but it implements the spaces $K(G,1)$ for some groups G.

Question: could HAP and Kenzo cooperate in computations where homology of groups is needed?

Resolutions

Resolutions

Definition

A *resolution* F_* for a group G is an acyclic chain complex of $\mathbb{Z}G$-modules

$$\cdots \longrightarrow F_2 \xrightarrow{d_2} F_1 \xrightarrow{d_1} F_0 \xrightarrow{\varepsilon} F_{-1} = \mathbb{Z} \longrightarrow 0$$

Resolutions

Definition

A *resolution* F_* for a group G is an acyclic chain complex of $\mathbb{Z}G$-modules

$$\cdots \longrightarrow F_2 \xrightarrow{d_2} F_1 \xrightarrow{d_1} F_0 \xrightarrow{\varepsilon} F_{-1} = \mathbb{Z} \longrightarrow 0$$

A chain complex of Abelian groups is obtained: $\mathbb{Z} \otimes_{\mathbb{Z}G} F_*$

Resolutions

Definition

A *resolution* F_* for a group G is an acyclic chain complex of $\mathbb{Z}G$-modules

$$\cdots \longrightarrow F_2 \xrightarrow{d_2} F_1 \xrightarrow{d_1} F_0 \xrightarrow{\varepsilon} F_{-1} = \mathbb{Z} \longrightarrow 0$$

A chain complex of Abelian groups is obtained: $\mathbb{Z} \otimes_{\mathbb{Z}G} F_*$

Theorem

Let G be a group and F_*, F'_* two free resolutions of G. Then

$$H_n(\mathbb{Z} \otimes_{\mathbb{Z}G} F_*) \cong H_n(\mathbb{Z} \otimes_{\mathbb{Z}G} F'_*) \cong H_n(K(G,1)) \quad \textit{for all } n \in \mathbb{N}$$

Resolutions

Definition

A *resolution* F_* for a group G is an acyclic chain complex of $\mathbb{Z}G$-modules

$$\cdots \longrightarrow F_2 \xrightarrow{d_2} F_1 \xrightarrow{d_1} F_0 \xrightarrow{\varepsilon} F_{-1} = \mathbb{Z} \longrightarrow 0$$

A chain complex of Abelian groups is obtained: $\mathbb{Z} \otimes_{\mathbb{Z}G} F_*$

Theorem

Let G be a group and F_*, F'_* two free resolutions of G. Then

$$H_n(\mathbb{Z} \otimes_{\mathbb{Z}G} F_*) \cong H_n(\mathbb{Z} \otimes_{\mathbb{Z}G} F'_*) \cong H_n(K(G,1)) \quad \textit{for all } n \in \mathbb{N}$$

Definition

Given a group G, the *homology groups* $H_n(G)$ are defined as $H_n(G) = H_n(\mathbb{Z} \otimes_{\mathbb{Z}G} F_*)$, $n \in \mathbb{N}$, where F_* is any free resolution for G.

Resolutions

Resolutions

One can always consider the *bar resolution* $B_* = \mathrm{Bar}_*(G)$

Resolutions

One can always consider the *bar resolution* $B_* = \mathrm{Bar}_*(G)$, which satisfies $\mathbb{Z} \otimes_{\mathbb{Z}G} B_* \equiv K(G, 1)$.

Resolutions

One can always consider the *bar resolution* $B_* = \mathrm{Bar}_*(G)$, which satisfies $\mathbb{Z} \otimes_{\mathbb{Z}G} B_* \equiv K(G,1)$. Drawback: for $n > 1$, $K(G,1)_n = G^n$.

Resolutions

One can always consider the *bar resolution* $B_* = \text{Bar}_*(G)$, which satisfies $\mathbb{Z} \otimes_{\mathbb{Z}G} B_* \equiv K(G,1)$. Drawback: for $n > 1$, $K(G,1)_n = G^n$.

For some particular cases, small (or minimal) resolutions can be directly constructed.

Resolutions

One can always consider the *bar resolution* $B_* = \mathrm{Bar}_*(G)$, which satisfies $\mathbb{Z} \otimes_{\mathbb{Z}G} B_* \equiv K(G,1)$. Drawback: for $n > 1$, $K(G,1)_n = G^n$.

For some particular cases, small (or minimal) resolutions can be directly constructed.

For instance, let $G = C_m$ with generator t. The resolution F_*

$$\cdots \xrightarrow{t-1} \mathbb{Z}G \xrightarrow{N} \mathbb{Z}G \xrightarrow{t-1} \mathbb{Z}G \longrightarrow \mathbb{Z} \longrightarrow 0$$

produces

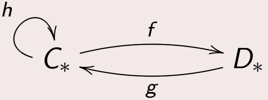
$$H_i(G) = \begin{cases} \mathbb{Z} & \text{if } \ i = 0 \\ \mathbb{Z}/m\mathbb{Z} & \text{if } \ i \text{ is odd} \\ 0 & \text{if } \ i \text{ is even}, i > 0 \end{cases}$$

Effective homology

Effective homology

Definition

A *reduction* ρ between two chain complexes C_* and D_* (denoted by $\rho : C_* \Rrightarrow D_*$) is a triple $\rho = (f, g, h)$



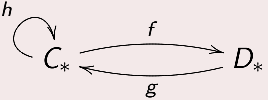
satisfying the following relations:

1) $fg = \mathrm{Id}_{D_*}$;
2) $d_C h + h d_C = \mathrm{Id}_{C_*} - gf$;
3) $fh = 0; \quad hg = 0; \quad hh = 0.$

Effective homology

Definition

A *reduction* ρ between two chain complexes C_* and D_* (denoted by $\rho : C_* \Rrightarrow D_*$) is a triple $\rho = (f, g, h)$



satisfying the following relations:

1) $fg = \mathrm{Id}_{D_*}$;
2) $d_C h + h d_C = \mathrm{Id}_{C_*} - gf$;
3) $fh = 0$; $\quad hg = 0$; $\quad hh = 0$.

If $C_* \Rrightarrow D_*$, then $C_* \cong D_* \oplus A_*$, with A_* acyclic, which implies that $H_n(C_*) \cong H_n(D_*)$ for all n.

Effective homology

Effective homology

Definition

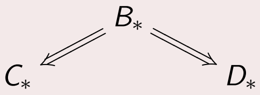
A *(strong chain) equivalence* ε between C_* and D_*, $\varepsilon : C_* \Lleftarrow\!\!\Rrightarrow D_*$, is a triple $\varepsilon = (B_*, \rho, \rho')$ where B_* is a chain complex, $\rho : B_* \Rrightarrow C_*$ and $\rho' : B_* \Rrightarrow D_*$.

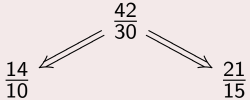
$$\begin{array}{ccccc} & & B_* & & \\ & \swarrow\!\!\swarrow & & \searrow\!\!\searrow & \\ C_* & & & & D_* \end{array} \qquad\qquad \begin{array}{ccccc} & & \frac{42}{30} & & \\ & \swarrow\!\!\swarrow & & \searrow\!\!\searrow & \\ \frac{14}{10} & & & & \frac{21}{15} \end{array}$$

Effective homology

Definition

A *(strong chain) equivalence* ε between C_* and D_*, $\varepsilon : C_* \Longleftarrow\!\!\Longrightarrow D_*$, is a triple $\varepsilon = (B_*, \rho, \rho')$ where B_* is a chain complex, $\rho : B_* \Rrightarrow C_*$ and $\rho' : B_* \Rrightarrow D_*$.





Definition

An *object with effective homology* is a quadruple $(X, C_*(X), HC_*, \varepsilon)$ where HC_* is an effective chain complex and $\varepsilon : C_*(X) \Longleftarrow\!\!\Longrightarrow HC_*$.

Effective homology

Definition

A *(strong chain) equivalence* ε between C_* and D_*, $\varepsilon : C_* \Longleftarrow\!\!\!\Longrightarrow D_*$, is a triple $\varepsilon = (B_*, \rho, \rho')$ where B_* is a chain complex, $\rho : B_* \Rrightarrow C_*$ and $\rho' : B_* \Rrightarrow D_*$.

$$\begin{array}{ccccc} & & B_* & & \\ & \swarrow\!\!\swarrow & & \searrow\!\!\searrow & \\ C_* & & & & D_* \end{array} \qquad\qquad \begin{array}{ccccc} & & \frac{42}{30} & & \\ & \swarrow\!\!\swarrow & & \searrow\!\!\searrow & \\ \frac{14}{10} & & & & \frac{21}{15} \end{array}$$

Definition

An *object with effective homology* is a quadruple $(X, C_*(X), HC_*, \varepsilon)$ where HC_* is an effective chain complex and $\varepsilon : C_*(X) \Longleftarrow\!\!\!\Longrightarrow HC_*$.

This implies that $H_n(X) \cong H_n(HC_*)$ for all n.

Algorithm computing the effective homology of a group

Algorithm computing the effective homology of a group

Given G a group, F_* a (small) free $\mathbb{Z}G$-resolution with a *contracting homotopy* $h_n : F_n \longrightarrow F_{n+1}$.

Algorithm computing the effective homology of a group

Given G a group, F_* a (small) free $\mathbb{Z}G$-resolution with a *contracting homotopy* $h_n : F_n \rightarrow F_{n+1}$.

Goal: an equivalence $C_*(K(G,1)) \Longleftrightarrow E_*$ where E_* is an effective chain complex.

Algorithm computing the effective homology of a group

Given G a group, F_* a (small) free $\mathbb{Z}G$-resolution with a *contracting homotopy* $h_n : F_n \rightarrow F_{n+1}$.

Goal: an equivalence $C_*(K(G,1)) \Longleftarrow\!\!\Longrightarrow E_*$ where E_* is an effective chain complex.

We consider the bar resolution $B_* = \mathrm{Bar}_*(G)$ for G with contracting homotopy h'.

Algorithm computing the effective homology of a group

Given G a group, F_* a (small) free $\mathbb{Z}G$-resolution with a *contracting homotopy* $h_n : F_n \to F_{n+1}$.

Goal: an equivalence $C_*(K(G,1)) \Longleftrightarrow E_*$ where E_* is an effective chain complex.

We consider the bar resolution $B_* = \mathrm{Bar}_*(G)$ for G with contracting homotopy h'.

It is well known that there exists a morphism of chain complexes of $\mathbb{Z}G$-modules $f : B_* \to F_*$ which is a homotopy equivalence.

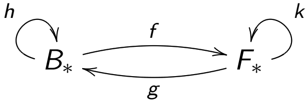
Algorithm computing the effective homology of a group

Given G a group, F_* a (small) free $\mathbb{Z}G$-resolution with a *contracting homotopy* $h_n : F_n \to F_{n+1}$.

Goal: an equivalence $C_*(K(G,1)) \Longleftarrow\!\!\!\Longrightarrow E_*$ where E_* is an effective chain complex.

We consider the bar resolution $B_* = \text{Bar}_*(G)$ for G with contracting homotopy h'.

It is well known that there exists a morphism of chain complexes of $\mathbb{Z}G$-modules $f : B_* \to F_*$ which is a homotopy equivalence. An algorithm has been designed constructing the explicit expressions of f and the corresponding maps g, h and k



Algorithm computing the effective homology of a group

Algorithm computing the effective homology of a group

Applying the functor $\mathbb{Z} \otimes_{\mathbb{Z}G} -$ we obtain an equivalence of chain complexes (of $\mathbb{Z}$-modules):

$$\overset{h}{\curvearrowright}\; \mathbb{Z} \otimes_{\mathbb{Z}G} B_* \underset{g}{\overset{f}{\rightleftarrows}} \mathbb{Z} \otimes_{\mathbb{Z}G} F_* \;\overset{k}{\curvearrowleft}$$

Algorithm computing the effective homology of a group

Applying the functor $\mathbb{Z} \otimes_{\mathbb{Z}G} -$ we obtain an equivalence of chain complexes (of $\mathbb{Z}$-modules):

$$\overset{h}{\curvearrowright}\;\mathbb{Z} \otimes_{\mathbb{Z}G} B_* \underset{g}{\overset{f}{\rightleftarrows}} \mathbb{Z} \otimes_{\mathbb{Z}G} F_*\;\overset{k}{\curvearrowleft}$$

In order to obtain a strong chain equivalence we make use of the mapping cylinder construction.

$$\mathbb{Z} \otimes_{\mathbb{Z}G} B_* \overset{\rho'}{\Lleftarrow} \text{Cylinder}(f)_* \overset{\rho}{\Rrightarrow} \mathbb{Z} \otimes_{\mathbb{Z}G} F_*$$

Algorithm computing the effective homology of a group

Applying the functor $\mathbb{Z} \otimes_{\mathbb{Z}G} -$ we obtain an equivalence of chain complexes (of $\mathbb{Z}$-modules):

$$h \circlearrowright \mathbb{Z} \otimes_{\mathbb{Z}G} B_* \underset{g}{\overset{f}{\rightleftarrows}} \mathbb{Z} \otimes_{\mathbb{Z}G} F_* \circlearrowleft k$$

In order to obtain a strong chain equivalence we make use of the mapping cylinder construction.

$$\mathbb{Z} \otimes_{\mathbb{Z}G} B_* \overset{\rho'}{\Leftarrow\!\!\Leftarrow} \text{Cylinder}(f)_* \overset{\rho}{\Rightarrow\!\!\Rightarrow} \mathbb{Z} \otimes_{\mathbb{Z}G} F_*$$

Finally we observe that the left chain complex $\mathbb{Z} \otimes_{\mathbb{Z}G} B_*$ is equal to $C_*(K(G,1))$

Algorithm computing the effective homology of a group

Applying the functor $\mathbb{Z} \otimes_{\mathbb{Z}G} -$ we obtain an equivalence of chain complexes (of $\mathbb{Z}$-modules):

$$\overset{h}{\circlearrowright}\ \mathbb{Z} \otimes_{\mathbb{Z}G} B_* \underset{g}{\overset{f}{\rightleftarrows}} \mathbb{Z} \otimes_{\mathbb{Z}G} F_*\ \overset{k}{\circlearrowleft}$$

In order to obtain a strong chain equivalence we make use of the mapping cylinder construction.

$$\mathbb{Z} \otimes_{\mathbb{Z}G} B_* \overset{\rho'}{\Lleftarrow} \text{Cylinder}(f)_* \overset{\rho}{\Rrightarrow} \mathbb{Z} \otimes_{\mathbb{Z}G} F_*$$

Finally we observe that the left chain complex $\mathbb{Z} \otimes_{\mathbb{Z}G} B_*$ is equal to $C_*(K(G,1))$. Moreover, if the initial resolution F_* is of finite type (and small), then the right chain complex $\mathbb{Z} \otimes_{\mathbb{Z}G} F_* \equiv E_*$ is effective.

Algorithm computing the effective homology of a group

Algorithm computing the effective homology of a group

Algorithm

Input: a group G and a free resolution F_* of finite type with contracting homotopy.
Output: the effective homology of $K(G,1)$, that is, a (strong chain) equivalence $C_*(K(G,1)) \lll\!\!\Longleftrightarrow\!\!\ggg E_*$ where E_* is an effective chain complex.

Exporting resolutions from HAP

Exporting resolutions from HAP

In order to produce the effective homology of $K(G,1)$, we need a resolution for G.

Exporting resolutions from HAP

In order to produce the effective homology of $K(G,1)$, we need a resolution for G. Instead of programming it directly in Kenzo, we will try to obtain it from the system HAP.

Exporting resolutions from HAP

In order to produce the effective homology of $K(G,1)$, we need a resolution for G. Instead of programming it directly in Kenzo, we will try to obtain it from the system HAP.

Format of interchange: OpenMath, an XML standard for representing mathematical objects.

Exporting resolutions from HAP

In order to produce the effective homology of $K(G,1)$, we need a resolution for G. Instead of programming it directly in Kenzo, we will try to obtain it from the system HAP.

Format of interchange: OpenMath, an XML standard for representing mathematical objects.

- We use the GAP package OpenMath to produce OpenMath code from some GAP elements such as lists or groups.

Exporting resolutions from HAP

In order to produce the effective homology of $K(G,1)$, we need a resolution for G. Instead of programming it directly in Kenzo, we will try to obtain it from the system HAP.

Format of interchange: OpenMath, an XML standard for representing mathematical objects.

- We use the GAP package OpenMath to produce OpenMath code from some GAP elements such as lists or groups.
- We extend this package in order to represent a resolution as a set of five elements

Exporting resolutions from HAP

In order to produce the effective homology of $K(G,1)$, we need a resolution for G. Instead of programming it directly in Kenzo, we will try to obtain it from the system HAP.

Format of interchange: OpenMath, an XML standard for representing mathematical objects.

- We use the GAP package OpenMath to produce OpenMath code from some GAP elements such as lists or groups.
- We extend this package in order to represent a resolution as a set of five elements
    - group
    - highest degree
    - list of ranks of each $\mathbb{Z}G$-module
    - boundary map
    - contracting homotopy

Exporting resolutions from HAP

Exporting resolutions from HAP

Group:

```
<OMA>
     <OMS cd="group1" name="Group"/>
     <OMA>
          <OMS cd="permut1" name="Permutation"/>
          <OMI> 3</OMI>
          <OMI> 2</OMI>
          <OMI> 1</OMI>
     </OMA>
     <OMA>
          <OMS cd="permut1" name="Permutation"/>
          <OMI> 1</OMI>
          <OMI> 3</OMI>
          <OMI> 5</OMI>
          <OMI> 4</OMI>
          <OMI> 2</OMI>
     </OMA>
</OMA>
```

Exporting resolutions from HAP

Exporting resolutions from HAP

Highest degree:

```
<OMI> 6</OMI>
```

Exporting resolutions from HAP

Highest degree:

```
<OMI> 6</OMI>
```

List of ranks:

```
<OMA>
     <OMS cd="list1" name="list"/>
     <OMI> 1</OMI>
     <OMI> 3</OMI>
     <OMI> 6</OMI>
     <OMI> 10</OMI>
     <OMI> 15</OMI>
     <OMI> 20</OMI>
     <OMI> 26</OMI>
</OMA>
```

Exporting resolutions from HAP

Exporting resolutions from HAP

The $\mathbb{Z}G$-boundary and the contracting homotopy are represented as lists containing the images of the generators of each module F_i.

Exporting resolutions from HAP

The $\mathbb{Z}G$-boundary and the contracting homotopy are represented as lists containing the images of the generators of each module F_i. For instance, $F_1 = (\mathbb{Z}G)^3$ has three generators. The boundary of the first one is the combination $1 * (g_2, z_1) - 1 * (g_1, z_1)$, represented in OpenMath as:

```
<OMA><OMS cd="resolutions" name="zgcombination"/>
     <OMA><OMS cd="resolutions" name="zgterm"/>
          <OMI> 1</OMI>
          <OMA><OMS cd="resolutions" name="zggnrt"/>
               <OMI> 2</OMI>
               <OMI> 1</OMI>
          </OMA>
     </OMA>
     <OMA><OMS cd="resolutions" name="zgterm"/>
          <OMI> -1</OMI>
          <OMA><OMS cd="resolutions" name="zggnrt"/>
               <OMI> 1</OMI>
               <OMI> 1</OMI>
          </OMA>
     </OMA>
</OMA>
```

Exporting resolutions from HAP

Exporting resolutions from HAP

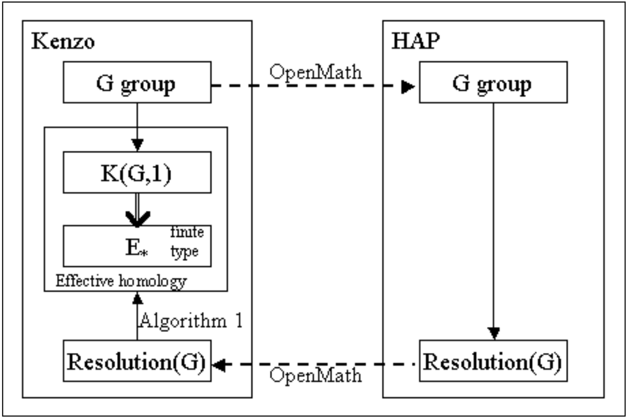


Figure: Communication between Kenzo and HAP

Applications and examples

Applications and examples

Computations with $K(G, n)'s$

Applications and examples

Computations with $K(G,n)'s$

```
> (setf KC71 (K-G-1 C7))
[K82 Abelian-Simplicial-Group]
> (efhm KC71)
[K119 Homotopy-Equivalence K82 <= K109 => K76]
```

Applications and examples

Computations with $K(G,n)'s$

```
> (setf KC71 (K-G-1 C7))
[K82 Abelian-Simplicial-Group]
> (efhm KC71)
[K119 Homotopy-Equivalence K82 <= K109 => K76]
```

The classifying space constructor $\overline{\mathcal{W}}$ gives us $\overline{\mathcal{W}}(K(G,1)) = K(G,2)$.

Applications and examples

Computations with $K(G, n)'s$

```
> (setf KC71 (K-G-1 C7))
[K82 Abelian-Simplicial-Group]
> (efhm KC71)
[K119 Homotopy-Equivalence K82 <= K109 => K76]
```

The classifying space constructor $\overline{\mathcal{W}}$ gives us $\overline{\mathcal{W}}(K(G, 1)) = K(G, 2)$.

```
> (setf KC72 (classifying-space KC71))
[K120 Abelian-Simplicial-Group]
> (efhm KC72)
[K259 Homotopy-Equivalence K120 <= K249 => K245]
> (homology KC72 3 6)
Homology in dimension 3 :
---done---
Homology in dimension 4 :
Component Z/7Z
---done---
Homology in dimension 5 :
---done---
```

Applications and examples

Applications and examples

Homology of 2-types

Applications and examples

Homology of 2-types

Let $G = C_3$, $A = \mathbb{Z}/3\mathbb{Z}$ with trivial G-action, and $[f] \in H^3(G, A) = \mathbb{Z}/3\mathbb{Z}$ a non-trivial cohomology class.

Applications and examples

Homology of 2-types

Let $G = C_3$, $A = \mathbb{Z}/3\mathbb{Z}$ with trivial G-action, and $[f] \in H^3(G, A) = \mathbb{Z}/3\mathbb{Z}$ a non-trivial cohomology class. This corresponds to a 2-type with $\pi_1 = G$ and $\pi_2 = A$, which can be seen as $X = K(A, 2) \times_f K(G, 1)$.

Applications and examples

Homology of 2-types

Let $G = C_3$, $A = \mathbb{Z}/3\mathbb{Z}$ with trivial G-action, and $[f] \in H^3(G, A) = \mathbb{Z}/3\mathbb{Z}$ a non-trivial cohomology class. This corresponds to a 2-type with $\pi_1 = G$ and $\pi_2 = A$, which can be seen as $X = K(A, 2) \times_f K(G, 1)$.

```
> (setf K-C3-1 (K-Cm-n 3 1))
[K261 Abelian-Simplicial-Group]
> (setf chml-clss (chml-clss K-C3-1 3))
[K308 Cohomology-Class on K288 of degree 3]
> (setf tau (zp-whitehead 3 K-C3-1 chml-clss))
[K323 Fibration K261 -> K309]
> (setf x (fibration-total tau))
[K329 Kan-Simplicial-Set]
> (efhm x)
[K541 Homotopy-Equivalence K329 <= K531 => K527]
> (homology x 5)
Homology in dimension 5 :
Component Z/3Z
---done---
```