

# An Algorithm for Low Dimensional Group Homology

Joshua Roberts

University Of Kentucky

JMM January 8, 2008

“Hopf’s formula” expresses the second integral homology  $H_2(G)$  of a finitely presented group  $G$  in terms of generators and relators. We give an algorithm based on GAP that exploits Hopf’s formula to estimate  $H_2(G; k)$  for a finite field  $k$ .

# First Homology Group

- Let a finite presentation for the group  $G$  be

$$1 \rightarrow R \rightarrow F \rightarrow G \rightarrow 1$$

- $H_1(G) = G/[G, G]$  can be found by putting the relation matrix in Smith Normal form.
- Take  $G = \langle a, b, c \mid a^2, b^3 \rangle$

```
gap> AbelianInvariants(G);  
[0, 2, 3]
```

- The “0” corresponds to one copy of the free cyclic group, “2” to the group of order 2, and “3” to the group of order 3.

# First Homology Group

- Let a finite presentation for the group  $G$  be

$$1 \rightarrow R \rightarrow F \rightarrow G \rightarrow 1$$

- $H_1(G) = G/[G, G]$  can be found by putting the relation matrix in Smith Normal form.
- Take  $G = \langle a, b, c \mid a^2, b^3 \rangle$

```
gap> AbelianInvariants(G);  
[0, 2, 3]
```

- The “0” corresponds to one copy of the free cyclic group, “2” to the group of order 2, and “3” to the group of order 3.

# First Homology Group

- Let a finite presentation for the group  $G$  be

$$1 \rightarrow R \rightarrow F \rightarrow G \rightarrow 1$$

- $H_1(G) = G/[G, G]$  can be found by putting the relation matrix in Smith Normal form.
- Take  $G = \langle a, b, c \mid a^2, b^3 \rangle$

```
gap> AbelianInvariants(G);  
[0, 2, 3]
```

- The “0” corresponds to one copy of the free cyclic group, “2” to the group of order 2, and “3” to the group of order 3.

# First Homology Group

- Let a finite presentation for the group  $G$  be

$$1 \rightarrow R \rightarrow F \rightarrow G \rightarrow 1$$

- $H_1(G) = G/[G, G]$  can be found by putting the relation matrix in Smith Normal form.
- Take  $G = \langle a, b, c \mid a^2, b^3 \rangle$

```
gap> AbelianInvariants(G);  
[0, 2, 3]
```

- The “0” corresponds to one copy of the free cyclic group, “2” to the group of order 2, and “3” to the group of order 3.

# First Homology Group

- Let a finite presentation for the group  $G$  be

$$1 \rightarrow R \rightarrow F \rightarrow G \rightarrow 1$$

- $H_1(G) = G/[G, G]$  can be found by putting the relation matrix in Smith Normal form.
- Take  $G = \langle a, b, c \mid a^2, b^3 \rangle$

```
gap> AbelianInvariants(G);  
[0, 2, 3]
```

- The “0” corresponds to one copy of the free cyclic group, “2” to the group of order 2, and “3” to the group of order 3.

- **Universal Coefficient Theorem:**

$$1 \rightarrow k \otimes H_1(G) \rightarrow H_1(G; k) \rightarrow \text{Tor}(H_0(G), k) \rightarrow 1$$

- The *Tor* functor vanishes on free abelian groups, so

$$k \otimes H_1(G) \cong H_1(G; k)$$

- We give an algorithm for the dimension of this **vector space over  $k$**  in “pseudocode”...

- **Universal Coefficient Theorem:**

$$1 \rightarrow k \otimes H_1(G) \rightarrow H_1(G; k) \rightarrow \text{Tor}(H_0(G), k) \rightarrow 1$$

- The *Tor* functor vanishes on free abelian groups, so

$$k \otimes H_1(G) \cong H_1(G; k)$$

- We give an algorithm for the dimension of this **vector space over  $k$**  in “pseudocode”...

- **Universal Coefficient Theorem:**

$$1 \rightarrow k \otimes H_1(G) \rightarrow H_1(G; k) \rightarrow \text{Tor}(H_0(G), k) \rightarrow 1$$

- The *Tor* functor vanishes on free abelian groups, so

$$k \otimes H_1(G) \cong H_1(G; k)$$

- We give an algorithm for the dimension of this **vector space over  $k$**  in “pseudocode”...

# First Homology with Coefficients Algorithm

FIRSTHOMOLOGYCOEFFICIENTS( $F, R, p$ )

**Input:** Free Group  $F$ ; Relators  $R$ ; Prime  $p = \text{char}(k)$

**Output:** Dimension of the vector space  $k \otimes H_1(G)$  over  $k$

1  $T := \text{AbelianInvariants}(F/R)$ ;  $X := []$ ;

2 **for**  $x \in T$

3     **do**  $\text{Is } x = 0$

4         **if** True

5             **then**  $x \in X$ ;

6     **else**  $\text{Is } x \equiv 0 \pmod{p}$ ;

7         **if** True

8             **then**  $x \in X$ ;

9 return  $\text{Size}(X)$

# Second Homology Group

- By Hopf's formula, we have the exact sequence

$$1 \rightarrow H_2(G) \rightarrow R/[F, R] \rightarrow F/[F, F] \rightarrow F/(R[F, F]) \rightarrow 1$$

- $R \cap [F, F]$  is not determined by any algorithm
- So we only estimate  $H_2(G)$  as a subgroup of  $R/[F, R]$
- To simplify the discussion we assume  $G$  is perfect

$$H_1(G) = F/R[F, F] = 1$$

and estimate  $H_2(G; k)$  where  $k$  is a field of order  $p$ .

# Second Homology Group

- By Hopf's formula, we have the exact sequence

$$1 \rightarrow H_2(G) \rightarrow R/[F, R] \rightarrow F/[F, F] \rightarrow F/(R[F, F]) \rightarrow 1$$

- $R \cap [F, F]$  is not determined by any algorithm
- So we only estimate  $H_2(G)$  as a subgroup of  $R/[F, R]$
- To simplify the discussion we assume  $G$  is perfect

$$H_1(G) = F/R[F, F] = 1$$

and estimate  $H_2(G; k)$  where  $k$  is a field of order  $p$ .

- By Hopf's formula, we have the exact sequence

$$1 \rightarrow H_2(G) \rightarrow R/[F, R] \rightarrow F/[F, F] \rightarrow F/(R[F, F]) \rightarrow 1$$

- $R \cap [F, F]$  is not determined by any algorithm
- So we only estimate  $H_2(G)$  as a subgroup of  $R/[F, R]$
- To simplify the discussion we assume  $G$  is perfect

$$H_1(G) = F/R[F, F] = 1$$

and estimate  $H_2(G; k)$  where  $k$  is a field of order  $p$ .

# Second Homology Group - Two Exact Sequences

- Then there is an exact sequence of vector spaces

$$1 \rightarrow k \otimes H_2(G) \rightarrow k \otimes R/[F, R] \rightarrow k \otimes F/[F, F] \rightarrow 1$$

and by the Universal Coefficient Theorem

$$k \otimes H_2(G) \cong H_2(G; k)$$

- Thus if we denote

$$a := \dim(H_2(G; k))$$

$$b := \dim(k \otimes R/[F, R])$$

$$c := \dim(k \otimes F/[F, F])$$

then

$$a = b - c$$

## Second Homology Group - Two Exact Sequences

- Then there is an exact sequence of vector spaces

$$1 \rightarrow k \otimes H_2(G) \rightarrow k \otimes R/[F, R] \rightarrow k \otimes F/[F, F] \rightarrow 1$$

and by the Universal Coefficient Theorem

$$k \otimes H_2(G) \cong H_2(G; k)$$

- Thus if we denote

$$a := \dim(H_2(G; k))$$

$$b := \dim(k \otimes R/[F, R])$$

$$c := \dim(k \otimes F/[F, F])$$

then

$$a = b - c$$

## Second Homology with Coefficients

- The previous algorithm calculates

$$c = \text{FirstHomologyCoefficients}(F, [F, F], p)$$

- However, to estimate  $b$  we need to reduce an element  $z$  of a group via a rewriting system

REDUCE\_WORD( $F, R, z, p, R'$ )

**Input:** Free group  $F$ , Relators  $R$ , Test word  $z$ ,  
Sublist  $R'$  of  $R$ , Prime  $p$

**Output:** Reduced word of  $z$  in  $F/[F, R]R^pR'$

- 1  $G := \frac{F}{[F, R]R^pR'}$ ;
- 2  $RG :=$  Rewriting system for  $G$ ;
- 3  $x := \text{ReducedWord}(z)$
- 4 return  $x$

## Second Homology with Coefficients

- The previous algorithm calculates

$$c = \text{FirstHomologyCoefficients}(F, [F, F], p)$$

- However, to estimate  $b$  we need to reduce an element  $z$  of a group via a rewriting system

REDUCE\_WORD( $F, R, z, p, R'$ )

**Input:** Free group  $F$ , Relators  $R$ , Test word  $z$ ,  
Sublist  $R'$  of  $R$ , Prime  $p$

**Output:** Reduced word of  $z$  in  $F/[F, R]R^pR'$

- 1  $G := \frac{F}{[F, R]R^pR'}$ ;
- 2  $RG :=$  Rewriting system for  $G$ ;
- 3  $x := \text{ReducedWord}(z)$
- 4 return  $x$

# Second Homology with Coefficients - The Basis Algorithm

- Next, we need to remove those elements  $x \in R$  that are the identity mod  $R - \{x\}$

FINDBASIS( $F, R, p, R'$ )

**Input:** Free group  $F$ , Relators  $R$ , Prime  $p$ , Sublist  $R'$  of  $R$

**Output:** Size of a generating set for  $R/[F, R] \otimes k$

```
1 for  $x \in R'$ ;
2     do  $x' := \text{Reduce\_Word}(F, R, x, \text{Difference}(R', [x]), p)$ ;
        ( $\text{Difference}(A, B)$  is the complement of  $B$  in  $A$ )
3     if  $x' = \text{identity}$ 
4         then  $R' := \text{Difference}(R', [x])$ ;
5  $b := \text{Size}(R')$ 
6 return  $b$ 
```

# Second Homology with Coefficients - The Basis Algorithm

- Next, we need to remove those elements  $x \in R$  that are the identity mod  $R - \{x\}$

FINDBASIS( $F, R, p, R'$ )

**Input:** Free group  $F$ , Relators  $R$ , Prime  $p$ , Sublist  $R'$  of  $R$

**Output:** Size of a generating set for  $R/[F, R] \otimes k$

```
1  for  $x \in R'$ ;
2      do  $x' := \text{Reduce\_Word}(F, R, x, \text{Difference}(R', [x]), p)$ ;
           ( $\text{Difference}(A, B)$  is the complement of  $B$  in  $A$ )
3      if  $x' = \text{identity}$ 
4          then  $R' := \text{Difference}(R', [x])$ ;
5   $b := \text{Size}(R')$ 
6  return  $b$ 
```

# A Word on Rewriting Systems

- We use the rewriting system given by the Knuth-Bendix completion algorithm implemented on GAP via the KBMAG package.
- When the rewriting system is confluent then the algorithm gives the exact dimension of  $H_2(G; k)$ .
- However, when the rewriting system is not confluent the algorithm gives only an upper bound.

# A Word on Rewriting Systems

- We use the rewriting system given by the Knuth-Bendix completion algorithm implemented on GAP via the KBMAG package.
- When the rewriting system is confluent then the algorithm gives the exact dimension of  $H_2(G; k)$ .
- However, when the rewriting system is not confluent the algorithm gives only an upper bound.

# A Word on Rewriting Systems

- We use the rewriting system given by the Knuth-Bendix completion algorithm implemented on GAP via the KBMAG package.
- When the rewriting system is confluent then the algorithm gives the exact dimension of  $H_2(G; k)$ .
- However, when the rewriting system is not confluent the algorithm gives only an upper bound.

## Example 1

The alternating group on 5 letters,  $A_5$

$$G = A_5$$

$$F = \langle x, y \rangle$$

$$R = \{x^2, y^3, (xy)^5\}$$

$$p = 2$$

$$b = 3$$

$$c = 2$$

$$a = 1$$

The rewriting system is confluent; therefore

$$\dim H_2(A_5; \mathbb{F}_2) = 1$$

The algorithm took about 16 milliseconds.

## Example 1

The alternating group on 5 letters,  $A_5$

$$\begin{aligned} G &= A_5 & p &= 2 \\ F &= \langle x, y \rangle & b &= 3 \\ R &= \{x^2, y^3, (xy)^5\} & c &= 2 \\ & & a &= 1 \end{aligned}$$

The rewriting system is confluent; therefore

$$\dim H_2(A_5; \mathbb{F}_2) = 1$$

The algorithm took about 16 milliseconds.

## Example 2

Special Linear Group  $SL_2(R)$ , for  $R = \mathbb{Z}[1/7, \zeta_7]$  and  $\zeta_7$  a primitive 7<sup>th</sup> root of unity.

$$\begin{array}{ll} G = SL_2(R) & p = 7 \\ F = \langle z, u_1, u_2, u_3, a, b, b_0, & b = 12 \\ & b_1, b_2, b_3, b_4, b_5, b_6, w \rangle & c = 6 \\ R = \text{set of 64 elements} & a = 6 \end{array}$$

The rewriting system is not confluent; therefore

$$\dim H_2(SL_2(R); \mathbb{F}_7) \leq 6$$

In this case, the algorithm took a total of about 5 days to finish.

- By a version of a conjecture of Quillen the map induced by inclusion,

$$H_i(GL_1^{\times j}(R); k) \rightarrow H_i(GL_j(R); k),$$

is an epimorphism. For  $i = j = 2$  this is equivalent to the surjectivity of a “transgression” map

$$H_3(GL_1(R); k) \rightarrow H_2(SL_2(R); k)_{GL_1(R)}.$$

- Our work is motivated by this conjecture.

# Thanks!

- Thank you for listening!