Homology, Homotopy and Applications, vol. 12(1), 2010, pp.27–37

AN ALGORITHM FOR LOW DIMENSIONAL GROUP HOMOLOGY

JOSHUA ROBERTS

(*communicated by Graham Ellis*)

Abstract

Given a finitely presented group G, Hopf's formula expresses the second integral homology of G in terms of generators and relators. We give an algorithm that exploits Hopf's formula to estimate $H_2(G;k)$, with coefficients in a finite field k, and give examples using $G = \mathrm{SL}_2$ over specific rings of integers. These examples are related to a conjecture of Quillen.

1. Introduction

The purpose of this note is to give an algorithm that allows us to estimate the second homology group of any finitely presented group. More precisely, given a finitely presented group G and a finite field k, the second homology group $H_2(G;k)$ with coefficients in k is a finite dimensional vector space over k, where G acts on k trivially. Our algorithm gives an upper bound for the dimension of $H_2(G;k)$ and, in particular cases, the algorithm calculates precisely this dimension. Existing algorithms such as those included with the the GAP [**7**] packages "cohomolo" [**9**] and "HAP" [**5**] are effective on finite groups and special classes of infinite groups. The algorithm presented here is novel in that it effectively finds a bound for the homology of any finitely presented group.

A motivational problem for low dimensional group homology is the study of homology for groups of the form $\mathrm{GL}_j(A)$, where GL_j is a finite rank general linear group scheme and A is a ring of arithmetic interest. An approach to this problem is to consider the diagonal matrices inside GL_j. Let D_j denote the subgroup formed by these matrices. Then the canonical inclusions $D_j \subset \mathrm{GL}_j$ for $j = 0, 1, \ldots$ induce homomorphisms on group homology with k-coefficients

$$\rho_{i,j}^{A,p} \colon H_i(D_j(A);k) \to H_i(\mathrm{GL}_j(A);k),$$

where k is the field of prime order p, i is called the homological dimension and j the rank. In this context, a celebrated conjecture of Quillen [**13**, p. 591] implies that $\rho_{i,j}^{A,p}$ is an epimorphism for $A = \mathbb{Z}[1/p, \zeta_p]$, p a regular odd prime, ζ_p a primitive p^{th} root of

Received January 23, 2009, revised December 1, 2009; published on February 19, 2010.
2000 Mathematics Subject Classification: 20J06, 20G30.
Key words and phrases: group homology, experimental mathematics, computational algebra, Hopf's formula.
This article is available at `http://intlpress.com/HHA/v12/n1/a3`

unity and any values of i and j. For a survey on the current status of this conjecture we cite [**1**].

By a spectral sequence argument applied to the group extension

$$1 \to \mathrm{SL}_j(A) \to \mathrm{GL}_j(A) \to D_1(A) \to 1$$

given by the determinant map, we can reformulate Quillen's conjecture in terms of $H_i(\mathrm{SL}_j(A); k)$. In the particular case $j = 2$, this homology has been studied extensively by using the theory of buildings. However, based on this theory we can calculate this homology only for i sufficiently large [**2**]. The problem of calculating $H_i(\mathrm{SL}_2(A); k)$ in low dimensions turns out to be highly non-trivial even when $i = 2$.

Examples in Section 4 confirm the results in [**1**], as well as give a new finding:

Theorem. *The dimension of* $H_2(\mathrm{SL}_2(\mathbb{Z}[1/7, \zeta_7]); \mathbb{F}_7)$ *as a vector space over* $\mathbb{F}_7$ *is at most* 6.

Our calculations were done with the computational algebra program GAP, and a GAP-ready text file containing this code can be found at:

`http://intlpress.com/HHA/v12/n1/a3/GAPcode.txt`

Acknowledgements

The author would like to thank Marian Anton for his enlightening discussions and Mark Dickinson for many helpful comments and recommended revisions.

2. First homology group

We consider a group given by a finite set of generators and a finite set of relators. If we denote this group by G, then there is a short exact sequence

$$1 \to R \to F \to G \to 1,$$

where F is a finitely-generated free group and R is a normal subgroup of F such that if F acts on R by conjugation then R is a finitely-generated F-module. Here if F and R are two groups not necessarily commutative then an F-module structure on R is an assignment $r \mapsto r^f$ for $r \in R$ and $f \in F$ such that

$$\begin{aligned} r^1 &= r, \\ (r_1 r_2)^f &= r_1^f r_2^f, \\ r^{f_1 f_2} &= (r^{f_1})^{f_2}, \end{aligned}$$

where, if not otherwise stated, all groups are given multiplicatively. In this context, it is well known that the first homology of a group is just another name for its abelianization [**4**, p. 8]. In particular, if we denote by $H_1(G)$ this abelian group, then there is a short exact sequence

$$1 \to R[F, F] \to F \to H_1(G) \to 1,$$

where $[F, F]$ denotes the subgroup of F generated by the commutators in F and the juxtaposition denotes the operation of taking the subgroup generated by the parts.

Letting F act on $R[F,F]$ by conjugation, we recognize that $R[F,F]$ is a finitely-generated F-module. Indeed, the commutator formula

$$[xy,z] = (xy)^{-1}z^{-1}xyz = y^{-1}x^{-1}z^{-1}xzyy^{-1}z^{-1}yz = [x,z]^y[y,z]$$

proves that since F is a finitely-generated group then $[F,F]$ is a finitely-generated F-module under conjugation and the same is assumed about R. This argument leads to a deterministic algorithm that gives the structure of $H_1(G)$. The input is a finite list of generators for F, say S, and a finite list of generators for the F-module R, say T. The output is a list of integers describing the structure of the finitely-generated abelian group $H_1(G)$.

2.1. The first homology algorithm

Algorithm 1: FirstHomology(F,R)

Input: Free Group F, Relators R

Output: List of abelian invariants of the finitely presented group F/R

1 $M :=$ Relation matrix of F/R
2 $N :=$ Smith normal form of M
3 **return** Diagonal entries of N

The GAP command `AbelianInvariants()` carries out (roughly) the above algorithm. An algorithm for reducing a matrix to a Smith Normal form is given in [**8**, p. 343]. Recall that given a finite presentation for F/R that consists of n generators S and m relators T, there is the associated $n \times m$ relation matrix M whose (i,j) entry is the sum of the exponents of all occurrences of the jth generator in the ith relator. The resulting list of diagonal entries of N is the set of entries in the (i,i) position for $i = 1 \cdots \min(n,m)$ and consists of positive integers and zeros. The number of zeros is the rank of $H_1(G)$ and each positive integer n corresponds to a copy of $\mathbb{Z}_n$ in the torsion part of $H_1(G)$.

This result can be extended to the case when the homology of G is taken with trivial coefficients in a finite field say k. In this case, the first homology group of G is denoted by $H_1(G;k)$ and is a finite dimensional vector space over k. Its dimension can be determined from the universal coefficients [**4**, p. 36] short exact sequence

$$1 \to k \otimes H_1(G) \to H_1(G;k) \to \mathrm{Tor}(H_0(G),k) \to 1,$$

where $H_0(G)$ is the free cyclic group and $\mathrm{Tor}(-,k)$ is a functor vanishing on free abelian groups. The algorithm takes as input the finite lists S and T from the previous algorithm together with the characteristic p of the finite field k. The output is an integer representing the dimension of the vector space $H_1(G;k)$.

2.2. The first homology with coefficients algorithm

Algorithm 2: FirstHomologyCoefficients(F,R,p)

Input: Free Group F, Relators R, Prime $p = \mathrm{char}(k)$

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

1 $T :=$FirstHomology(F,R)
2 $X := [\,]$
3 **for** $x \in X$ **do**

```
    if x ≡ 0 mod p then
      append x to X
    end if
  end for
  return Size(X) {number of elements in the list X}
```

3. Second homology group

Our investigation can be extended to the second homology group of G which is an abelian group that we denote $H_2(G)$. By a celebrated formula due to Hopf [**4**, p. 42] this group fits into the following exact sequence:

$$1 \to [F, R] \to R \cap [F, F] \to H_2(G) \to 1,$$

where $[F, R]$ is the subgroup of F generated by the commutators $[f, r]$ with $f \in F$ and $r \in R$. The commutator formula

$$[x, y^z] = x^{-1}(y^{-1})^z xy^z = x^{-1}z^{-1}y^{-1}zx(yy^{-1})z^{-1}yz = [zx, y][y, z]$$

proves that $[F, R]$ is a finitely-generated F-module under conjugation. However the intersection $R \cap [F, F]$ is not determined by any algorithm, and we can only estimate the group $H_2(G)$ as a subgroup of the factor group $R/[F, R]$. This factor group is abelian since $[F, R]$ contains $[R, R]$ and if we let F act on it by conjugation, then this action is trivial. In particular, since R is a finitely-generated F-module it follows that the factor group $R/[F, R]$ is a finitely-generated abelian group. Consequently, $H_2(G)$ is a finitely-generated abelian group whose structure we would like to determine.

We start with the following exact sequence

$$1 \to H_2(G) \to \frac{R}{[F, R]} \to \frac{F}{[F, F]} \to \frac{F}{R[F, F]} \to 1 \tag{1}$$

in which the last two terms are deterministically determined as explained above. Moreover, starting with a finite list of generators T for the F-module R, we can design a deterministic algorithm to find a set of generators for $H_2(G)$.

To simplify the discussion, let k denote the finite field of prime order p and start our investigation with the homology with trivial coefficients in k. By the universal coefficients theorem we have a short exact sequence

$$1 \to k \otimes H_2(G) \to H_2(G; k) \to \mathrm{Tor}(H_1(G), k) \to 1$$

whose last term can be determined as follows. For input we start with the abelian invariants of $H_1(G)$ found by the first algorithm together with the order p of the field k. The output is an integer say a representing the dimension of the vector space $\mathrm{Tor}(H_1(G), k)$ over k. The algorithm is deterministic.

3.1. The Tor Algorithm

Algorithm 3: TOR(F, R, p)
Input: Free Group F, Relators R, Prime $p = \mathrm{char}(k)$
Output: Dimension of $\mathrm{Tor}(H_1(G), k)$ over k

```
A :=FirstHomology(F, R)
X := [ ]
for x ∈ A do
   if x ≠ 0 and x ≡ 0 mod p  then
      append x to X
   end if
end for
return  Size(X)
```

The first term $k \otimes H_2(G)$ of the exact sequence is a finite dimensional vector space over k whose dimension can only be estimated from above by an algorithm that we will describe next. From the exact sequence (1) we extract the short exact sequence

$$1 \to H_2(G) \to \frac{R}{[F,R]} \to \frac{R[F,F]}{[F,F]} \to 1$$

whose last term is a subgroup of the free abelian group $F/[F,F]$. It is a standard fact that any subgroup of a finitely-generated free abelian group is free abelian and consequently the above sequence splits. In particular, by tensoring with k we obtain a short exact sequence of vector spaces over k:

$$1 \to k \otimes H_2(G) \to k \otimes \frac{R}{[F,R]} \to k \otimes \frac{R[F,F]}{[F,F]} \to 1,$$

where the last term can be rewritten as $R[F,F]/R^p[F,F]$. Here R^p denotes the subgroup of F generated by the p-powers of elements of R. In particular, there is a short exact sequence of finitely-generated abelian groups

$$1 \to k \otimes \frac{R[F,F]}{[F,F]} \to \frac{F}{R^p[F,F]} \to \frac{F}{R[F,F]} \to 1$$

whose last two terms are deterministically computable by our first algorithm.

Definition 3.1 ([**6**, p. 6])**.** For an abelian group A, define the **p-primary subgroup of** A to be

$$_{p^\infty}(A) = \{a \in A \mid a^{p^i} = 1 \text{ for some } i > 0\}.$$

The order of this subgroup is of the form p^e. Call e the **p^∞-rank of** A.

The p^∞ rank of a finitely-generated abelian group A can be calculated by taking as input the abelian invariants of A and the prime p.

By passing to p-primary subgroups, sequence 3.1 gives another short exact sequence

$$1 \to k \otimes \frac{R[F,F]}{[F,F]} \to_{p^\infty} \left(\frac{F}{R^p[F,F]}\right) \to_{p^\infty} \left(\frac{F}{R[F,F]}\right) \to 1$$

since the first term is p-torsion. We observe that while $F/R[F,F]$ can be given in terms of S and T, the factor group $F/R^p[F,F]$ can be given in the same way but replacing T by T^p, the finite list of p-powers of elements in T.

3.2. The Rank Algorithm

Algorithm 4: PrimePrimaryRank(F, R, p)

Input: Free Group F, Relators R, Prime p

Output: p^∞-rank of F/R
1 A :=FirstHomology(F, R)
2 $Y := [\,]$
3 **for** $a \in A$ **do**
4 **if** $a \neq 0$ and $a \equiv 0 \bmod p$ **then**
5 $y :=$ p-adic valuation of a
6 append y to Y
7 **end if**
8 **end for**
9 s :=Sum(Y) $\{s$ is the sum of the elements of $Y\}$
10 **return** s

The GAP command `PadicValuation(n,p)` gives the p-adic valuation of an integer n.

To summarize, let

$$\begin{aligned} a &= \text{dimension of } \mathrm{Tor}(H_1(G), k), \\ b &= p^\infty\text{-rank of } \frac{F}{R[F,F]}, \\ c &= p^\infty\text{-rank of } \frac{F}{R^p[F,F]}, \\ d &= \text{dimension of } H_2(G;k), \\ e &= \text{dimension of } k \otimes \frac{R}{[F,R]}, \end{aligned}$$

where a is determined by the Tor Algorithm, b and c by the Rank Algorithm, and e is yet to be studied. By the additive property of the dimension and the p^∞-rank, we deduce, from the exact sequences above, the following reduction formula:

$$d = a + b - c + e.$$

Since a, b, c are more or less standard, the integer e is the key difficulty we aim to approach experimentally.

We first describe an algorithm that reduces an element of a group via a rewriting system.

3.3. Reduce word algorithm

Algorithm 5: ReduceWord(F, R, Z, R', p)
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 $G := F/[F, R]R^pR'$
2 $RG :=$ Rewriting system for G
3 $x :=$ Reduced word of (z) in the rewriting system RG
4 **return** x

We use the rewriting system given by the Knuth-Bendix completion algorithm [**11**] implemented on GAP via the KBMAG package [**10**].

3.4. The find basis algorithm

Algorithm 6: FINDBASIS(F, R, p, R')
Input: Free Group F, Relators R, Prime p, Sublist R' of R
Output: Size of a generating set for $[F, R]R^pR'/[F, R]R^p$

```
1 X := R'
2 for x ∈ X do
3    x' := REDUCEWORD(F, R, x, Difference(X, [x]), p)  {Difference(A, B) is the
     complement of B in A}
4    if x' = identity then
5       X := Difference(X, [x])
6    end if
7 end for
8 return Size(X)
```

The algorithm attempts to check for linear independence of each element x of R' with respect to $R' - \{x\}$ in $[F, R]R^pR'/[F, R]R^p$. Whenever x is found by the rewriting system to be dependent of $R' - \{x\}$, it is removed from R'. The end result will be a list of potentially linearly independent generators.

We conclude this discussion with the grand scheme algorithm which takes as input a finite list of generators S and a finite list of relators T for a group G together with a prime p and gives as output an integer d representing an upper bound for the dimension of $H_2(G; k)$, where k is a field of characteristic p.

3.5. The second homology with coefficients algorithm

Algorithm 7: SECONDHOMOLOGYCOEFFICIENTS(F, R, p, R')
Input: Free Group F, Relators R, Prime p, Sublist R' of R generating $R/[F, R]R^p$
Output: An integer d such that $\dim(H_2(G; k)) \leqslant d$

```
1 a := TOR(F, R, p)
2 b := PRIMEPRIMARYRANK(F, R[F, F], p)
3 c := PRIMEPRIMARYRANK(F, R^p[F, F], p)
4 e := FINDBASIS(F, R, p, R')
5 d := a + b − c + e
6 return d
```

It is important to note that the reduction of test words in the algorithm REDUCEWORD is the word problem (for a description of the word problem see [**3**]). As such, a result of a word not being the identity is an indeterminate result. However, if G is finite, or, more generally, if the rewriting is confluent, then the reduction in the rewriting system is deterministic and a basis is achieved (the confluence for finite groups is guaranteed in theory only; in practice it may take a long time or require more space than is available). At any rate, this is not typically the case–the word problem is undecidable in general; thus the result of FINDBASIS is, in general, the cardinality of a generating set that is not necessarily a basis. Therefore in these cases we do not find the dimension of $H_2(G; k)$, only an upper bound.

4. Examples

In this section, we apply the grand scheme algorithm above to some select groups. The first example is to illustrate the effect the algorithm has on groups with small-ish presentations. The other three examples are the groups of primary interest. In Section 5 we will discuss these calculations.

Example 4.1. The symmetric groups Σ_n on n letters:

$$\begin{aligned} G &= \Sigma_5, \\ S &= [a, b], \\ T &= [a^5, b^2, (a^{-1}b)^4, (a^2ba^{-2}b)^2], \\ p &= 2, \\ d &= 2. \end{aligned}$$

Next we consider three linear groups over $\mathbb{Z}[1/p, \zeta_p]$, where ζ_p is a primitive p^{th}-root of unity. Presentations for groups of this from can be found in [**1**, pp. 447, 453].

Example 4.2.

$$\begin{aligned} G &= \mathrm{SL}_2(\mathbb{Z}[1/3, \zeta_3]), \\ S &= [z, u_1, a, b, b_0, b_1, b_2, w], \\ T &= [b_t^{-1}z^{3t}bz^{3t}a, w^{-1}z^4u_1u_2u_3, z^3, [z, u_1], [u_1, u_1], a^4, [a^2, z], [a^2, u_1], \\ &\quad a^{-1}zaz, a^{-1}u_1au_1, [b_s, b_t], b^{-3}a^2, b^{-3}b_0b_1b_2, \\ &\quad (b_0b_1^{-1}a^{-1}u_1)^3, a^{-2}b^{-1}u_1bz^{-3}b^{-1}b_0^{-1}z^3bz^{-1}u_1], \\ p &= 3, \\ d &= 0, \end{aligned}$$

where $s, t \in \{1, 2\}$.

Example 4.3.

$$\begin{aligned} G &= \mathrm{SL}_2(\mathbb{Z}[1/5, \zeta_5]), \\ S &= [z, u_1, u_2, a, b, b_0, b_1, b_2, b_3, b_4, w], \\ T &= [b_t^{-1}z^{3t}bz^{3t}a, w^{-1}z^4u_1u_2u_3, z^5, [z, u_i], [u_i, u_j], a^4, [a^2, z], [a^2, u_i], \\ &\quad a^{-1}zaz, a^{-1}u_iau_i, [b_s, b_t], b^{-3}a^2, b^{-3}b_0b_1b_2b_3b_4, \\ &\quad (b_0b_1^{-1}a^{-1}u_1)^3, (b_0b_2^{-1}a^{-1}u_2)^3, (b_0b_3^{-1}a^{-1}u_3)^3, \\ &\quad (b_0b_1^{-1}b_2^{-1}b_3a^{-1}u_1u_2)^3, (b_0b_1^{-1}b_3^{-1}b_4a^{-1}u_1u_3)^3, \\ &\quad (b_0b_2^{-1}b_3^{-1}b_5a^{-1}u_2u_3)^3, {}^{-2}\, b^{-1}u_ibz^{-3i}b^{-1}b_0^{-1}z^{3i}bz^{-i}u_i], \\ p &= 5, \\ d &= 0, \end{aligned}$$

where $i, j \in \{1, 2\}$ and $s, t \in \{1, 2, 3, 4\}$.

Example 4.4.

$$\begin{aligned}
G &= \mathrm{SL}_2(\mathbb{Z}[1/7, \zeta_7]),\\
S &= [z, u_1, u_2, u_3, a, b, b_0, b_1, b_2, b_3, b_4, b_5, b_6, w],\\
T &= [b_t^{-1} z^{3t} b z^{3t} a, w^{-1} z^4 u_1 u_2 u_3, z^7, [z, u_i], [u_i, u_j], a^4, [a^2, z], [a^2, u_i],\\
&\quad a^{-1} z a z, a^{-1} u_i a u_i, [b_s, b_t], b^{-3} a^2, b^{-3} b_0 b_1 b_2 b_3 b_4 b_5 b_6, b_t^{-7} w^{-1} b_t^{-1} w,\\
&\quad (b_0 b_1^{-1} a^{-1} u_1)^3, (b_0 b_2^{-1} a^{-1} u_2)^3, (b_0 b_3^{-1} a^{-1} u_3)^3,\\
&\quad (b_0 b_1^{-1} b_2^{-1} b_3 a^{-1} u_1 u_2)^3, (b_0 b_1^{-1} b_3^{-1} b_4 a^{-1} u_1 u_3)^3, (b_0 b_2^{-1} b_3^{-1} b_5 a^{-1} u_2 u_3)^3,\\
&\quad (b_0 b_1^{-1} b_2^{-1} b_3 b_4 b_5 b_6^{-1} a^{-1} u_1 u_2 u_3)^3, a^{-2} b^{-1} u_i b z^{-3i} b^{-1} b_0^{-1} z^{3i} b z^{-i} u_i],\\
p &= 7,\\
d &= 6,
\end{aligned}$$

where $i, j \in \{1, 2, 3\}$ and $s, t \in \{1, 2, 3, 4, 5, 6\}$.

5. Discussion and future work

Details on the above examples are as follows:

- *Example 4.1.* The rewriting system given by the KBMAG package for Σ_5 is confluent; therefore
$$\dim H_2(\Sigma_5; \mathbb{F}_2) = 2.$$
The algorithm took about 50 milliseconds to run, reflecting the relatively simple presentation.
- *Example 4.2.* The rewriting system given by the KBMAG package for $\mathrm{SL}_2(\mathbb{Z}[1/3, \zeta_3])$ is not confluent; the algorithm took about six hours to finish. In this case, the non-confluence of the system did not affect the results as the rewriting system was able to show that all elements of R reduced to identity modulo $[F, R]R^3$, so
$$\dim H_2(\mathrm{SL}_2(\mathbb{Z}[1/3, \zeta_3]; \mathbb{F}_3) = 0.$$
- *Example 4.3.* The rewriting system given by the KBMAG package for $SL_2(\mathbb{Z}[1/5, \zeta_5])$ is not confluent. As in Example 2 the non-confluence of the system did not affect the results and
$$\dim H_2(\mathrm{SL}_2(\mathbb{Z}[1/5, \zeta_5]) = 0.$$
The algorithm took about two days to finish.
- *Example 4.4.* The rewriting system given by the KBMAG package for $\mathrm{SL}_2(\mathbb{Z}[1/7, \zeta_7])$ is not confluent. In this case, the algorithm took a total of about five days to finish. Also, this is the only case tested in which the non-confluence actually mattered. Since the algorithms were not able to show that the dimension of $R/[F, R]R^7$ is 0, we only have the upper bound
$$\dim H_2(\mathrm{SL}_2(\mathbb{Z}[1/7, \zeta_7]); \mathbb{F}_7) \leqslant 6.$$

We note that for Examples 4.3 and 4.4, it was necessary to run the algorithm several times to obtain the results above since the parameters of the KBMAG package

allow a limited number of equations to be generated in the rewriting system. Each iteration eliminated elements of R from the generating list until the results stabilized. For instance, in Example 4.4, the initial iteration gave a result of $e \leqslant 16$ and $d \leqslant 10$, the second iteration gave that $e \leqslant 13$ and $d \leqslant 7$. The third and fourth iterations each gave a result of $e \leqslant 12$ and so the upper bound on d is 6.

Finally, in implementing these algorithms to find a bound on $H_2(G)$ it is useful to first perform Tietze transforms on the presentations involved to attempt to simplify the presentations. In many cases, the number of generators and relators can be reduced, thus simplifying the calculations. A description of Tietze transformations can be found in [**12**, pp. 89-99]. In Example 4.4, $\mathrm{SL}_2(\mathbb{Z}[1/7, \zeta_7])$ is given via a presentation consisting of 14 generators and 64 relators. A series of Tietze transforms, implemented via GAP, simplifies to a presentation with six generators and 34 relators. This significantly impacts the results of the algorithm.

Our future work will involve refining and improving the algorithms above. Initially we were concerned only with writing algorithms that gave results–the efficiency of these algorithms was not a concern. For the linear groups above as p increases the number of relators grows exponentially; thus the algorithms will take longer and longer to finish. Going from $p = 2$ to $p = 7$, the time required increased from several hours to several days.

We also will develop other methods for finding generators of $H_2(G)$ and $H_2(G; k)$ independent from those above. In particular, we attempt to find *lower* bounds on the dimension of $H_2(G; k)$. The strategies for both problems will be based on linear algebra involving rewriting systems for T in $F/[F, F]$ and will appear in a future paper.

References

[**1**] M. F. Anton, Homological symbols and the Quillen conjecture, *Pure Appl. Algebra* **213** (2009), no. 4, 440–453. MR2483829

[**2**] A. Borel and J.-P. Serre, Cohomologie d'immeubles et de groupes S-arithmétiques, *Topology* **15** (1976), no. 3, 211–232. MR0447474 (56 #5786)

[**3**] J. L. Britton, The word problem for groups, *Proc. London Math. Soc.* **8** (1958), no. 3, 493–506. MR0125019 (23 #A2326)

[**4**] K. S. Brown, *Cohomology of groups*, Grad. Texts in Math. **87**, Springer-Verlag, New York, 1982. MR1324339 (96a:20072)

[**5**] G. Ellis, GAP *package* HAP, 2008, `http://hamilton.nuigalway.ie/Hap/www/index.html`.

[**6**] C. Faith, *Rings and things and a fine array of twentieth century associative algebra*, Mathematical Surveys and Monographs, vol. 65, American Mathematical Society, Providence, RI, 1999. MR1657671 (99j:01015)

[**7**] The GAP group, GAP – *Groups, Algorithms, and Programming*, Version 4.4.10, 2007, `http://www.gap-system.org`.

[**8**] D. F. Holt, B. Eick and E. O'Brien *Handbook of computational group theory*, Discrete Mathematics and its Applications, Chapman & Hall/CRC Press, Boca Raton, FL, 2005. MR2129747 (2006f:20001)

[**9**] _____, GAP *package cohomolo*, 2008, `http://www.warwick.ac.uk/staff/D.F.Holt/cohomolo/`.

[**10**] _____, GAP *package* kbmag, 2009, `http://www.warwick.ac.uk/~mareg/kbmag`.

[**11**] D. E. Knuth and P. B. Bendix, Simple word problems in universal algebras, in *Computational Problems in Abstract Algebra* (Proc. Conf., Oxford, 1967), Pergamon Press, Oxford, 1970, pp. 263–297. MR0255472 (41 #134)

[**12**] Roger C. Lyndon and Paul E. Schupp, *Combinatorial group theory,* Classics in Mathematics, Springer-Verlag, Berlin, 2001. MR1812024 (2001i:20064)

[**13**] D. Quillen, The spectrum of an equivariant cohomology ring. I, II, *Ann. of Math. (2)* **94** (1971), no. 3, 549–572; *ibid. (2)* **94** (1971), no. 3, 573–602. MR0298694 (45 #7743)

Joshua Roberts `jroberts@ms.uky.edu`

Mathematics Department, University of Kentucky, 718 Patterson Office Tower, Lexington, KY 40506-0027, USA