J. Symbolic Computation (1994) **11**, 1–000

Isomorphism testing for p-groups

E.A. O'BRIEN

Centre for Mathematics and its Applications

School of Mathematical Sciences, Australian National University, Canberra, ACT 0200, Australia

E-mail address: obrien@maths.anu.edu.au

(Received)

We describe the theoretical and practical details of an algorithm which can be used to decide whether two given presentations for finite p-groups present isomorphic groups. The approach adopted is to construct a canonical presentation for each group. A description of the automorphism group of the p-group is also constructed.

1. Introduction

The isomorphism problem of determining whether two given presentations present the same group was introduced by Tietze (1908) and later formulated by Dehn in a 1911 paper. Adian (1957) and Rabin (1958) showed that the isomorphism problem for finitely presented groups is unsolvable by exhibiting its unsolvability for a particular class of examples. This work was later extended by Boone (1968). However, Segal (1990) proves that there is an algorithm available to decide the isomorphism of two polycyclic-by-finite groups given by finite presentations.

There are practical approaches available to solving the problem within particular contexts. Sometimes, the easier task is to establish that two groups are non-isomorphic by exhibiting invariants where the groups differ. However, it is frequently difficult to find "natural" invariants which distinguish among similar groups. For example, to differentiate among the 267 groups of order 64, the structure of the subgroup lattices of individual groups is on some occasions required.

In a general approach to finitely-presented groups, Holt & Rees (1992) seek to establish isomorphism by running a Knuth-Bendix procedure on the supplied group presentations, in an attempt to generate a normal form/word reduction algorithm for words in the generators. Concurrently, they attempt to establish non-isomorphism of the two groups by finding the number of finite quotients each has of a particular order.

Wursthorn has adapted modular group algebra techniques, which were developed in seeking counter-examples to the modular group algebra isomorphism conjecture, to work for p-groups. His approach is described in Wursthorn (1993).

The author thanks the Institut für Experimentelle Mathematik, Universität Essen, for its hospitality while he did part of this work.

0747–7171/90/000000 + 00 $03.00/0

In this paper, we describe an algorithm which, theoretically, provides an answer to the problem for finite p-groups. The approach adopted here is to define a canonical presentation for each p-group and to provide an algorithm which allows its construction. Hence, given two p-groups presented by arbitrary finite presentations, the determination of their isomorphism is essentially the same problem as the construction of their canonical presentations and the comparison of these presentations.

A description of the automorphism group of the p-group is constructed concurrently with the standard presentation for the group. We do not discuss the algorithm used to construct the automorphism group in detail here; the interested reader is referred to O'Brien (1994).

In Section 2, we discuss the use of power-commutator presentations in presenting p-groups and the methods available for the construction of such presentations. In the three subsequent sections, we describe the theory of the *standard presentation algorithm* used to construct the canonical presentation, provide a top-level outline of the algorithm, and give some details of a practical implementation. In Section 6, we present a detailed calculation using the algorithm. In Section 7, we propose some refinements to the original algorithm to enhance its performance. In the final section, we provide some information on the performance of the implementation.

An earlier description and partial implementation of this algorithm is given in Schultz (1988); another discussion can be found in Ascione (1979).

2. Group presentations

Let X be a non-empty set and let F be the free group on X. A group presentation is a set consisting of X and a set, $\mathcal{R}$, of words in X. The presentation is written $\{X : \mathcal{R}\}$. The normal closure of $\mathcal{R}$ in F will usually be denoted by $\langle\mathcal{R}\rangle^F$; the group defined by the presentation is $F/\langle\mathcal{R}\rangle^F$ and is written $\langle X : \mathcal{R} \rangle$.

The *generator number* (or, equivalently, the number of defining generators) of a group G is the cardinality of a smallest set X such that G is defined by a presentation $\{X : \mathcal{R}\}$, where $\mathcal{R}$ is a set of words in X. In this paper, the generator number is usually denoted by d.

2.1. Power-commutator presentations

Finite groups of prime-power order may be described uniformly using a special type of presentation known as a *power-commutator presentation.* The generating set is a finite set $\{a_1, \dots, a_n\}$. The defining relations are:

$$\begin{aligned} a_i^p &= \prod_{k=i+1}^{n} a_k^{\beta(i,k)},\ 0 \le \beta(i,k) < p\,,\ 1 \le i \le n\,, \\ [a_j, a_i] &= \prod_{k=j+1}^{n} a_k^{\beta(i,j,k)},\ 0 \le \beta(i,j,k) < p\,,\ 1 \le i < j \le n\,. \end{aligned}$$

Such presentations were first defined by Sylow (1872) who proved that every group of order p^n has a power-commutator presentation on n generators. If a presentation on n generators defines a group of order p^n, then the presentation is *consistent.*

A power-commutator presentation for a finite p-group may be constructed using a

p-quotient algorithm. The first of such algorithms was described by Macdonald (1974). Havas & Newman (1980) describe the algorithm in common usage today.

Their algorithm uses a variation of the lower central series known as the *lower exponent-p central series*. This is the descending sequence of subgroups

$$G = P_0(G) \geq \ldots \geq P_{i-1}(G) \geq P_i(G) \geq \ldots$$

where $P_i(G) = [P_{i-1}(G), G]P_{i-1}(G)^p$ for $i \geq 1$.

If $P_c(G) = 1$ and c is the smallest such integer then G has *exponent-p class c*. A group with exponent-p class c is nilpotent and has nilpotency class at most c. In this paper, the class of a group refers to its exponent-p class.

Given a description of a group G, a prime p, and a positive integer c, the p-quotient algorithm constructs a consistent power-commutator presentation for the largest p-quotient of G having class at most c.

Of course, the power-commutator presentation produced as output by a p-quotient algorithm depends on the presentation or other description supplied as input. In no sense can a power-commutator presentation produced by an arbitrary p-quotient algorithm calculation be viewed as canonical.

2.2. The p-group generation algorithm

In this section, we give a brief description of the p-group generation algorithm. A detailed description of the algorithm together with relevant proofs may be found in O'Brien (1990) and in Newman (1977).

The p-group generation algorithm calculates (presentations for) particular extensions, known as immediate descendants, of a finite p-group.

Let G be a finite p-group with generator number d and class c. A group H is a *descendant* of G if H has generator number d and the quotient $H/P_c(H)$ is isomorphic to G. A group is an *immediate descendant* of G if it is a descendant of G and has class $c+1$.

The algorithm takes as input a d-generator p-group, G, defined as a quotient, F/R, of the free group F on d generators. It also requires a description of the automorphism group of G. It produces as output a complete and irredundant list of the immediate descendants of G together with a description of their automorphism groups.

The group G is described by a power-commutator presentation computed using a p-quotient algorithm. Using this presentation, a consistent power-commutator presentation is written down for a *p-covering group*, F/R^*, of G, where $R^* = [R, F]R^p$.

Theorem 2.1. *Every immediate descendant of G is isomorphic to a factor group of F/R^*.*

The p-covering group has the following important property.

Lemma 2.2. *The isomorphism type of G^* depends only on G and not on R.*

A description of the algorithm used to construct the p-covering group, denoted by G^*, is provided in Havas & Newman (1980).

The factor group R/R^* is elementary abelian and is known as the *p-multiplicator* of

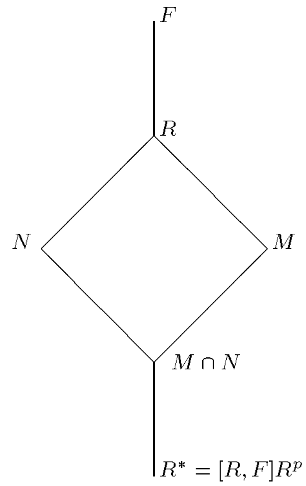

Figure 1. Various subgroups of the p-covering group

G; the *nucleus* of G is $P_c(G^*)$. An *allowable subgroup* is a subgroup of the p-multiplicator which is the kernel of a homomorphism from G^* onto an immediate descendant of G.

The allowable subgroups are characterised by the following theorem.

THEOREM 2.3. *A subgroup is allowable if and only if it is a proper subgroup of the p-multiplicator of G which supplements the nucleus.*

Figure 1 illustrates the situation, where N/R^* represents the nucleus and M/R^* is an allowable subgroup.

On taking factor groups of G^* by allowable subgroups a complete list of immediate descendants is obtained; this list usually contains redundancies. To eliminate these redundancies, an obvious equivalence relation is defined on the allowable subgroups.

DEFINITION 2.4. *Two allowable subgroups M_1/R^* and M_2/R^* are equivalent if and only if their quotients F/M_1 and F/M_2 are isomorphic.*

A complete and irredundant set of immediate descendants of G can be obtained by factoring G^* by one representative of each equivalence class. In practice, this definition is useful only because the equivalence relation can be given a different characterisation by using the automorphism group of G. An *extension* of each automorphism, α, of G to an automorphism, α^*, of G^* is defined. The action of α^* when restricted to the p-multiplicator of G is uniquely determined by α, and α^* induces a permutation of the allowable subgroups.

THEOREM 2.5. *The equivalence classes of allowable subgroups are exactly the orbits of the allowable subgroups under the action of these permutations.*

Thus, we designate one element of each orbit as its representative and factor the p-

covering group by each representative in turn to obtain a complete and irredundant list of immediate descendants of the starting group, G.

The choice of orbit representative determines the presentation obtained. Two elements from the same orbit determine different power-commutator presentations for isomorphic groups. On what basis do we choose the orbit representative? We associate with each allowable subgroup a *label* – a unique positive integer which runs from one to the number of allowable subgroups. The element with the smallest label, the *leading term*, is always chosen as the orbit representative. The methods for representing the allowable subgroups and assigning labels are described in detail in O'Brien (1990, §3.3).

An alternative view of the p-group generation algorithm is that it is a method for constructing a particular power-commutator presentation for a given p-group, G.

Assume G has generator number d and class c. Then $G/P_1(G)$ is the elementary abelian group of order p^d and, thus, G is a descendant of this elementary abelian group. It is also clear, from a consideration of properties of the lower exponent-p central series, that $G/P_{i+1}(G)$ is an immediate descendant of $G/P_i(G)$ for $i < c$.

Assume we construct the immediate descendants of $G/P_1(G)$. Among these immediate descendants is the class two quotient, $G/P_2(G)$, of G. It is now possible to calculate the immediate descendants of $G/P_2(G)$ in order to obtain a power-commutator presentation for the class three quotient of G. We may iterate this construction until we construct the class c quotient of G. Therefore, it is possible to construct G by iterating a method for calculating immediate descendants, starting with the elementary abelian group of rank d.

We designate the presentation obtained by constructing a power-commutator presentation for a given p-group using the p-group generation algorithm in this way as the *standard presentation* for this group.

3. Theory of the standard presentation algorithm

Let $\mathcal{P} = \{a_1, \ldots, a_d : \mathcal{R}\}$ be the supplied presentation. Let F be the free group on d generators; let R be the normal closure of $\mathcal{R}$ and $G = F/R$ is a p-group of Frattini quotient rank d. Since $P_1(F)R/P_1(F)$ is the identity in $F/P_1(F)$, $\mathcal{R}$ is a subset of $P_1(F)$.

Let $\mathcal{S}$ be a subset of $P_1(F)$ and let S be its normal closure in F.

Definition 3.1. *$\mathcal{R}$ and $\mathcal{S}$ are* **presentation equivalent** *if, for all $i \geq 1$,*

$$\frac{F/R}{P_i(F/R)} \cong \frac{F/S}{P_i(F/S)}.$$

Now assume that the standard power-commutator presentation for the class k quotient of the group, $G/P_k(G)$, has been constructed on power-commutator presentation generators $a_1, \ldots, a_n$. It is straightforward, using Tietze transformations, to convert this presentation into one whose relations involve only the defining generators, $a_1, \ldots, a_d$. As a result of this process, we obtain a new presentation $\{a_1, \ldots, a_d : \mathcal{R}_k\}$.

Definition 3.2. *Let $\mathcal{S}_k$ be a subset of $P_1(F)$. Then $\mathcal{S}_k$ is a* **class k standard set of defining relations** *for G if $\mathcal{S}_k$ is presentation equivalent to $\mathcal{R}$ and $\langle \mathcal{R}_k \rangle^F = P_k(F)\langle \mathcal{S}_k \rangle^F$.*

The standard presentation for the class one quotient, $G/P_1(G)$, is

$$\{a_1, \ldots, a_d : [a_j, a_i] = 1, a_i^p = 1, 1 \leq i < j \leq d\}.$$

The supplied set of defining relations, $\mathcal{R}$, is a class one standard set of defining relations.

At its kth iteration, the standard presentation algorithm takes as input the standard presentation for $G/P_k(G)$ and a class k standard set of defining relations, $\mathcal{S}_k$. It produces as output the standard presentation for $G/P_{k+1}(G)$ and a class $k+1$ standard set of defining relations, $\mathcal{S}_{k+1}$.

We introduce two sequences of subgroups. Let $P_1(F) = S_1 \geq S_2 \geq \ldots \geq S_k$ be defined by

$$S_i = P_i(F)\langle \mathcal{S}_k \rangle^F \quad \text{for } i = 1, \ldots, k.$$

Let $P_1(F) = T_1 \geq T_2 \geq \ldots$ be defined by

$$T_i = P_i(F)\langle \mathcal{S}_k \rangle^F \quad \text{for } i \geq 1.$$

Then T_i equals S_i for $i = 1, \ldots, k$.

We also define

$$R^* = [S_k, F]S_k^p.$$

Definition 3.3. *Let α be an automorphism of F/S_k which maps a_iS_k to $w_i(a_1, \ldots, a_d)S_k$ for some choice of w_i. Define a homomorphism $\tilde{\alpha} : F \longmapsto F$ by $a_i\tilde{\alpha} = w_i(a_1, \ldots, a_d)$ and a homomorphism $\alpha^* : F/R^* \longmapsto F/R^*$ by $a_iR^*\alpha^* = (a_i\tilde{\alpha})R^*$.*

Lemma 3.4. *The map α^* is an automorphism.*

A proof of this lemma appears in O'Brien (1990, Lemma 2.6). The automorphism, α^*, is an *extension* of α.

The set $\mathcal{S}_k$ is a class k standard set of defining relations for G and is, by definition, presentation equivalent to $\mathcal{R}$. Hence, $F/T_{k+1} \cong G/P_{k+1}(G)$. But $G/P_{k+1}(G)$ is an immediate descendant of $G/P_k(G)$ which is isomorphic to F/T_k.

Therefore, F/T_{k+1} is a quotient of the p-covering group, F/R^*, of F/T_k and T_{k+1}/R^* is an allowable subgroup in the p-multiplicator. Let the leading term of the orbit which contains T_{k+1}/R^* be S_{k+1}/R^*. We factor F/R^* by S_{k+1}/R^* to obtain the standard presentation for the class $k+1$ quotient.

We have completed the first part of the construction – we now discuss how to obtain a class $k+1$ standard set of defining relations.

Definition 3.5. *An automorphism, δ, of F/S_k whose extension, δ^*, maps T_{k+1}/R^* to S_{k+1}/R^* is a* **standard automorphism**.

Let $\mathcal{S}_{k+1} = \{w\tilde{\delta} : w \in \mathcal{S}_k\}$ be the result of applying $\tilde{\delta}$ to $\mathcal{S}_k$. We claim that $\mathcal{S}_{k+1}$ is a class $k+1$ standard set of defining relations for G.

To establish this, we define the following normal series:

$$P_1(F) = U_1 \geq U_2 \geq \ldots$$

where $U_i = P_i(F)\langle \mathcal{S}_{k+1} \rangle^F$ for $i \geq 1$.

We first show that $U_i = S_i$ for $i = 1, \ldots, k+1$ and then prove that $\mathcal{S}_{k+1}$ is presentation equivalent to $\mathcal{R}$.

LEMMA 3.6. *$U_i = S_i$ for $i = 1, \ldots, k+1$.*

PROOF. First, consider the action of δ^* on T_i/R^*:

$$\begin{aligned} T_i\tilde{\delta}R^* &= (P_i(F)\langle\mathcal{S}_k\rangle^F)\tilde{\delta}R^* \\ &= (P_i(F)\tilde{\delta})(\langle\mathcal{S}_k\rangle^F\tilde{\delta})R^* \\ &= P_i(F)\langle\mathcal{S}_{k+1}\rangle^F R^* \\ &= U_i R^*. \end{aligned}$$

Therefore,

$$(T_i/R^*)\delta^* = U_i/R^*.$$

Now consider $i \in \{1, \ldots, k\}$. By definition, $S_i = P_i(F)\langle\mathcal{S}_k\rangle^F$. Since $\langle\mathcal{S}_k\rangle^F$ is a subgroup of S_k which is contained in S_i, it follows that $S_i = P_i(F)S_k$. We deduce that S_i/S_k is a term of the lower exponent-p central series of F/S_k. Each term of this series is invariant; the p-multiplicator is characteristic; hence, S_i/R^* is fixed under the action of the extended automorphism, δ^*. Therefore, for $i = 1, \ldots, k$,

$$\begin{aligned} U_i/R^* &= (T_i/R^*)\delta^* \\ &= (S_i/R^*)\delta^* \quad \text{since } S_i = T_i \\ &= S_i/R^*. \end{aligned}$$

We also have

$$\begin{aligned} U_{k+1}/R^* &= (T_{k+1}/R^*)\delta^* \\ &= S_{k+1}/R^* \quad \text{by the definition of } \delta. \end{aligned}$$

This gives the desired result. □

In order to establish that $\mathcal{S}_{k+1}$ is presentation equivalent to $\mathcal{R}$, we introduce the following series of subgroups. Let $S_k = R_0^* \geq R_1^* \geq \ldots$ be defined by

$$R_i^* = [R_{i-1}^*, F](R_{i-1}^*)^p$$

for all $i \geq 1$. We now define automorphisms $\alpha_i^* : F/R_i^* \longmapsto F/R_i^*$, by analogy with Definition 3.3. Note that R_1^* is simply R^* and α_1^* is α^* as used earlier in this section.

LEMMA 3.7. *$\mathcal{S}_{k+1}$ is a class $k+1$ standard set of defining relations for G.*

PROOF. Since $P_1(F)$, a term of the lower exponent-p central series, is invariant, it contains $\mathcal{S}_{k+1}$.

We show that $\mathcal{S}_{k+1}$ is presentation equivalent to $\mathcal{S}_k$; since this is in turn presentation equivalent to $\mathcal{R}$, the result follows.

From Lemma 3.6, $F/T_i = F/U_i$ for $i = 1, \ldots, k$. In order to prove the presentation equivalence, we must demonstrate that $F/T_{k+i} \cong F/U_{k+i}$ for all $i \geq 1$.

Define a map $\gamma : F/T_{k+i} \longmapsto F/U_{k+i}$ by

$$wT_{k+i}\gamma = (w\tilde{\delta})U_{k+i}.$$

From the definition of δ, we know that $T_{k+i}\tilde{\delta} \leq U_{k+i}$. Therefore, γ is well-defined. Also, since $R_i^* \leq U_{k+i}$ and δ_i^* is onto, γ is onto.

We now establish that γ is an isomorphism. Let wT_{k+i} be in the kernel of γ. Then $w\tilde{\delta} \in U_{k+i}$ implying that

$$(wR_i^*)\delta_i^* \in U_{k+i}/R_i^*.$$

But a generalisation of the proof given in Lemma 3.6 shows that $(T_{k+i}/R_i^*)\delta_i^* = U_{k+i}/R_i^*$. Therefore, $wR_i^* \in T_{k+i}/R_i^*$, showing that $w \in T_{k+i}$. The isomorphism follows and so does the result. $\Box$

4. The standard presentation algorithm

We now summarise a procedure which can be iterated to construct the standard presentation of a given p-group.

We assume that the standard presentation for the class k p-quotient of the group has been constructed and that a generating set for the automorphism group of this quotient is known. We now wish to construct the standard presentation for the class $k+1$ quotient.

Let $H = G/P_k(G)$. The presentation $\mathcal{P} = \{X : \mathcal{S}_k\}$, where $\mathcal{S}_k$ is a class k standard set of defining relations for G, the standard presentation for H, and the generating set for the automorphism group are the inputs to the procedure.

1 Use the standard presentation for H to write down a presentation for its p-covering group, H^*.
2 Use $\mathcal{S}_k$ as input to a p-quotient algorithm to compute a presentation for the class $k+1$ quotient of G.
3 Recognise the allowable subgroup, M/R^*, which must be factored from H^* to give the presentation computed for the class $k+1$ quotient.
4 Extend the elements of the generating set for the automorphism group of H to act on the p-covering group, H^*.
5 Compute the orbit of M/R^* under the action of these automorphisms and let L/R^* be its leading term. Factor L/R^* from H^* to obtain the standard presentation for $G/P_{k+1}(G)$.
6 Compute an automorphism whose extension maps M/R^* to L/R^*.
7 Modify the relations of $\mathcal{S}_k$ by applying this standard automorphism to each. The resulting modified set is $\mathcal{S}_{k+1}$.

The output of the procedure is the modified set of defining relations, $\mathcal{S}_{k+1}$, and the standard presentation for $G/P_{k+1}(G)$.

The group presented by $\{X : \mathcal{S}_{k+1}\}$ has the standard presentation for its class $k+1$ quotient.

Note that a generating set for the automorphism group of the class k quotient is required as input to the procedure. In practice, at the kth iteration of the algorithm, a description of the automorphism group of $G/P_{k+1}(G)$ is computed at the same time as the standard presentation for this quotient is computed. This generating set is used as input to the next iteration of the procedure. For a description of the method used to construct this generating set, see O'Brien (1994).

5. An implementation of the algorithm

A detailed description of the implementation of the p-group generation algorithm is given in O'Brien (1990). Much of the subsequent discussion assumes some level of familiarity with the details of that description.

As a precursor to applying the standard presentation algorithm, the user's presentation is first supplied as input to the p-quotient algorithm, and we obtain as output the standard presentation for $G/P_1(G)$.

At the kth iteration, the standard presentation for $H = G/P_k(G)$ is used by the p-quotient algorithm to write down a presentation for the p-covering group, H^*. We next supply the class k standard set of defining relations, $\mathcal{S}_k$, as input to the p-quotient algorithm and determine a consistent power-commutator presentation for $G/P_{k+1}(G)$.

Once this presentation has been computed, it is easy to determine which allowable subgroup must be factored from H^* in order to obtain $G/P_{k+1}(G)$. The generators of the last term of the lower exponent-p central series in the presentation for $G/P_{k+1}(G)$ form the definition set for this subgroup and hence we can obtain its label.

We now carry out a partial run of the p-group generation algorithm. First, we supply a generating set for the automorphism group of $G/P_k(G)$. (When $k = 1$, we supply simply a generating set for the appropriate general linear group.) These automorphisms are extended to act on the p-covering group. The orbit of the allowable subgroup under the action of these automorphisms is then computed and the representative of the orbit noted.

An automorphism must now be determined whose extension maps an element of the orbit to the representative of that orbit. In computing the orbit of the allowable subgroup, the necessary information has already been computed which permits one to trace a word in the defining automorphisms whose extension has this property. Once the word is obtained, it is evaluated to give a standard automorphism.

The standard automorphism is now applied to $\mathcal{S}_k$ in order to obtain $\mathcal{S}_{k+1}$. A file containing the class $k+1$ standard set of defining relations together with a description of the automorphism group of this quotient is created. This file contains the necessary input for the next iteration of the algorithm.

If the algorithm is used to verify that two groups are isomorphic, it is easy to obtain an explicit isomorphism between them by recording the standard automorphisms applied at each class to bring each presentation to the single standard presentation.

A new implementation of the Havas and Newman p-quotient algorithm has been developed by the author. It builds on the previous implementation and provides a range of new facilities. Some of these are described in Newman & O'Brien (in preparation). The author's implementations of the p-group generation and standard presentation algorithms are combined with this to form the core components of the ANU p-Quotient Program.

A facility is provided by the program which allows the user to save the computed standard presentation of a group to file. The user may compare this presentation with any other to determine whether the two presentations are identical. This comparison is carried out by writing down a *compact description* of each group – a sequence whose entries are the exponents which occur in the relations of each standard presentation – and then comparing these integer sequences.

In summary, the program provides an interface which allows a user to compute the standard presentation for a given group class by class, to save all relevant data, and carry out comparisons of standard presentations constructed.

The ANU p-Quotient Program is written in C and is available on request from the author. We provide additional access to all three implementations via both GAP (see Schönert *et al.*, 1993) and MAGMA (see Butler & Cannon, 1989).

6. A sample calculation

In this section, we compute a standard presentation for the class two 3-quotient of the group, G, having presentation

$$\{x, y : (xyx)^3\}.$$

The class one 3-quotient, $H = G/P_1(G)$, has the standard power-commutator presentation:

$$\{\ a_1, a_2\ :\ a_1^3 = 1,\ a_2^3 = 1, [a_2, a_1] = 1\ \}.$$

The supplied set of defining relations is a class one standard set of defining relations, $\mathcal{S}_1$. We now apply the algorithm.

1 We use the standard power-commutator presentation as input to write down a presentation for the 3-covering group of H. Subject to the convention that all relations whose right-hand sides are trivial are not shown, this group, H^*, has presentation

$$\{\ a_1, \ldots, a_5 : [a_2, a_1] = a_3, a_1^3 = a_4, a_2^3 = a_5\ \}.$$

The nucleus is $\langle\ a_3, a_4, a_5\ \rangle$.

2 We use $\mathcal{S}_1$ as input to write down a presentation for the class two 3-quotient of G. This quotient has presentation

$$\{\ a_1, \ldots, a_4 : [a_2, a_1] = a_3, a_1^3 = a_4, a_2^3 = a_4\ \}.$$

3 The allowable subgroup, M/R^*, which must be factored from H^* to give this presentation for the class two quotient is $\langle a_4^2 a_5\rangle$.

4 A generating set for the automorphism group of H is

$$\alpha_1 : \begin{array}{lll} a_1 & \longmapsto & a_1^2 \\ a_2 & \longmapsto & a_2^2 \end{array},\quad \alpha_2 : \begin{array}{lll} a_1 & \longmapsto & a_2^2 \\ a_2 & \longmapsto & a_1 \end{array},\quad \alpha_3 : \begin{array}{lll} a_1 & \longmapsto & a_1 a_2^2 \\ a_2 & \longmapsto & a_1^2 a_2^2 \end{array},$$

$$\alpha_4 : \begin{array}{lll} a_1 & \longmapsto & a_1 \\ a_2 & \longmapsto & a_1^2 a_2 \end{array},\quad \alpha_5 : \begin{array}{lll} a_1 & \longmapsto & a_1^2 \\ a_2 & \longmapsto & a_2. \end{array}$$

The automorphism matrices representing the action of α_i^* on the 3-multiplicator of H are, respectively:

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{pmatrix}, \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 2 \\ 0 & 1 & 0 \end{pmatrix}, \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 2 \\ 0 & 2 & 2 \end{pmatrix}, \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 2 & 1 \end{pmatrix}, \begin{pmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

5 The orbit containing M/R^* is

$$\langle a_5\rangle, \langle a_4 a_5\rangle, \langle a_4^2 a_5\rangle, \langle a_4\rangle.$$

The orbit representative, L/R^*, is $\langle a_5\rangle$. We factor H^* by $\langle a_5\rangle$ to obtain the standard presentation for the class two quotient:

$$\{\ a_1, \ldots, a_4 : [a_2, a_1] = a_3, a_1^3 = a_4\ \}.$$

6 A standard automorphism whose extension maps M/R^* to L/R^* is the following:

$$\delta : \begin{array}{lll} a_1 & \longmapsto & a_1 a_2 a_3 a_4 \\ a_2 & \longmapsto & a_1 a_2^2\ . \end{array}$$

7 We now modify the relations of $\mathcal{S}_1$ by applying the standard automorphism to each. Hence $\mathcal{S}_2$ is

$$\{(xy[y,x]x^3xy^2xy[y,x]x^3)^3\}.$$

7. Improving the performance of the algorithm

The limitations on the performance of the standard presentation algorithm are essentially those inherent to the performance of the p-group generation algorithm.

Assume that we have constructed the standard presentation for the class k p-quotient of G and we now wish to construct the standard presentation of the class $k+1$ p-quotient.

Let the rank of the p-multiplicator of $G/P_k(G)$ be q and let the *width* of the kth term, s, be $\log_p |(G/P_{k+1}(G))/(G/P_k(G))|$. In constructing the standard presentation for $G/P_{k+1}(G)$, we compute an orbit of subspaces of dimension $q-s$ in the q-dimensional space. The number of such subspaces (and hence the potential size of this orbit) is largest when s is half the value of q. Of course, this number also depends on the prime, p. The space required to represent the allowable subgroups and to calculate the desired orbit is the most serious limitation. The time taken to compute the orbit is determined by its length.

In O'Brien (1990, §4), a strategy involving the use of characteristic subgroups in the p-multiplicator was described which significantly extends the range of application of the p-group generation algorithm. A further extension to the algorithm was described in O'Brien (1991).

The strategy of using characteristic subgroups has been incorporated into the standard presentation algorithm.

This involves some modifications of the algorithm described in Section 4. The discussion relies heavily on the concepts introduced in O'Brien (1990).

Steps 1 and 2 of the algorithm remain the same. Step 3 is split into two parts:

3a. We recognise the allowable subgroup which must be factored from the p-covering group and hence determine the step size required for the class $k+1$ construction.

3b. We choose the smallest initial-segment characteristic subgroup in the p-multiplicator and all of our remaining calculations are performed relative to this subgroup. The *relative allowable subgroup* is the intersection of the allowable subgroup with this initial-segment characteristic subgroup. We determine the relative allowable subgroup which must be factored from the p-covering group and hence obtain the relative step size.

As before, we now find the representative of the orbit containing the relative allowable subgroup.

If the characteristic subgroup is a proper subgroup of the p-multiplicator, then under Step 5, the orbit representative is factored from the p-covering group to give, not the standard presentation for the class $k+1$ quotient but instead, a presentation for a *reduced p-covering group*. The relations are now modified under the action of a standard automorphism. Steps 3b to 7 of the algorithm are now iterated taking as input the reduced p-covering group and the modified set of defining relations.

When the characteristic subgroup chosen is the (reduced) p-multiplicator, the standard presentation is obtained together with a class $k+1$ standard set of defining relations.

As reported in O'Brien (1990), these intermediate stage computations significantly extend the range of applicability of the p-group generation algorithm. They have a similar impact on the performance of the standard presentation algorithm.

The default strategy of the implementation is to choose the initial-segment characteristic subgroup of the smallest permitted rank at each intermediate stage of the calculations and perform all calculations relative to this subgroup.

7.1. Other strategies

In the discussion of the algorithm presented here, we have defined the standard presentation of a group as that obtained by constructing the group using the p-group generation algorithm. In checking whether two groups, say G and H, are isomorphic, we may choose another definition for their standard presentations.

As an example, we may redefine the "standard presentation" of G to be simply the power-commutator presentation obtained by using its given presentation as input to a p-quotient algorithm. We may now seek to establish that H is isomorphic to G by verifying that at each class, k, the quotient, $H/P_k(H)$, has a presentation which is identical to that of $G/P_k(G)$. Assume that this is true for the class k p-quotients. Now consider the construction of the class $k+1$ p-quotient of each. Let M_1/R^* and M_2/R^* be the allowable subgroups whose quotients provide the presentations for the class $k+1$ p-quotients, respectively. If G and H are isomorphic then these two subgroups are in the same orbit. Hence, it is sufficient to find an automorphism whose extension maps M_2/R^* to M_1/R^* and to apply this automorphism to the relations of the presentation for H. In this way, we reduce the problem of checking the isomorphism of $H/P_{k+1}(H)$ and $G/P_{k+1}(G)$ to building up the orbit of M_2/R^* until it is complete or we find M_1/R^*. In practice, the saving in time is not significant using this approach but there may be some reduction in the amount of space used.

It may be the case that the power-commutator presentations obtained by handing the finite presentations for G and H to a p-quotient algorithm are identical up to some class k. If this is so, we need only apply the standard presentation algorithm from class $k+1$ onwards; however, we must supply a generating set for the automorphism group for the class k quotient as input. If a generating set is known, the use of this feature may reduce significantly the cost of testing for isomorphism.

8. Performance data

As already mentioned, the implementation of the algorithm is most effective when the number of allowable subgroups is reasonably "small". Another limiting factor is the number of generators of the automorphism group.

A crude additional guide to the range of applicability of the implementation is obtained by considering the rank of the Frattini quotient of the group. In practice, the performance of the algorithm is best when the rank is at most five. In cases where the Frattini rank is larger, but the ratio of the width of each term of the lower exponent-p central series to the ranks of the associated smallest initial-segment characteristic subgroups is "close" to either 0 or 1, the implementation may still be useful.

Consider the following groups:

$$\begin{aligned} G_1 &= \langle\, x, y : x^4, y^4 = [y, x, x] \,\rangle; \\ G_2 &= \langle\, x, y : [x, y, y, y], [x, y, x] \,\rangle; \end{aligned}$$

Class	Order	Time
1	2^2	–
2	2^5	0.1
3	2^8	0.1
4	2^{11}	0.2
5	2^{15}	0.4
6	2^{19}	0.4
7	2^{25}	0.9
8	2^{32}	1.5
9	2^{42}	4.5
10	2^{55}	32.7

Table 1. Constructing standard presentations for 2-quotients of G_1

$$\begin{aligned} G_3 &= \langle\, x, y : x^{25}, [y,x,x,x], y^5 = [y,x,x] \,\rangle; \\ G_4 &= \langle\, x, y : x^{11}, y^{11}, z^{11}, [y,x,x,x,x,x], [z,x], [z,y], [y,x,y] \,\rangle. \end{aligned}$$

Tables 1–4 list the times taken to compute the standard presentations for certain quotients of these groups. All CPU times are in seconds and calculations were carried out on a Sparc 10/31 machine. All computations were carried out using the default implementation discussed in Section 7.

The time taken to compute the standard presentation is usually significantly greater than that taken to compute an arbitrary power-commutator presentation for a given finite p-group. For example, using the author's implementation of the p-quotient algorithm, a consistent power-commutator presentation was constructed for the largest quotient of each of the sample groups in about one second of CPU time. Hence, the standard presentation of a group should be constructed only when the canonical nature of this presentation is a desired property.

References

Adian, S.I. (1958). "On algorithmic problems in effectively complete classes of groups", *Doklady Akad. Nauk. SSR*, **123**, 13–16.

Ascione, Judith A. (1979). *On 3-groups of second maximal class*, PhD thesis. Australian National University.

Boone, W.W. (1968). "Decision problems about algebraic and logical systems as a whole and recursively enumerable degrees of unsolvability", K. Schütte (Ed.), *Contributions to Mathematical Logic*. North-Holland, Amsterdam.

Class	Order	Time
1	3^2	–
2	3^5	0.1
3	3^9	0.1
4	3^{13}	0.3
5	3^{17}	0.4
6	3^{21}	0.8
7	3^{25}	1.0
8	3^{29}	1.2
9	3^{33}	1.7
10	3^{37}	2.0

Table 2. Constructing standard presentations for 3-quotients of G_2

Class	Order	Time
1	5^2	–
2	5^4	0.2
3	5^6	0.1
4	5^7	0.2
5	5^9	0.3
6	5^{11}	0.3
7	5^{15}	0.8
8	5^{18}	0.6
9	5^{23}	3.9
10	5^{28}	4.4

Table 3. Constructing standard presentations for 5-quotients of G_3

Class	Order	Time
1	11^3	–
2	11^4	0.2
3	11^5	0.3
4	11^6	0.4
5	11^8	8.0
6	11^9	0.5
7	11^{11}	8.1
8	11^{12}	1.3
9	11^{13}	1.7
10	11^{14}	2.6

Table 4. Constructing standard presentations for 11-quotients of G_4

Butler, G., Cannon, John (1989). "Cayley, version 4: the user language". Proceedings of International Symposium on Symbolic and Algebraic Computation (Rome, 1988). Lecture Notes in Computer Science **358**, pp. 456–466. Springer-Verlag, Berlin.

Dehn, M. (1911). "Über unendliche diskontinuierliche Gruppen", *Math. Ann.*, **71**, 116–144.

Havas, George, Newman, M.F. (1980). "Application of computers to questions like those of Burnside", *Burnside Groups*, (Bielefeld, 1977). Lecture Notes in Math., **806**, pp. 211–230. Springer-Verlag, Berlin, Heidelberg, New York.

Holt, D.F., Rees, Sarah (1992). "Testing for isomorphism between finitely presented groups", *Groups and Combinatorics*, (Durham, 1989). London Math. Soc. Lecture Note Ser. **165**, pp. 459-475. Cambridge University Press, London.

Macdonald, I.D. (1974). "A computer application to finite p-groups", *J. Austral. Math. Soc. Ser. A*, **17**, 102–112.

Newman, M.F. (1977). "Determination of groups of prime-power order", *Group Theory*, (Canberra, 1975). Lecture Notes in Math., **573**, pp. 73–84. Springer-Verlag, Berlin, Heidelberg, New York.

Newman, M.F., O'Brien, E.A. (in preparation). "Application of computers to questions like those of Burnside, II".

O'Brien, E.A. (1990). "The p-group generation algorithm", *J. Symbolic Comput.*, **9**, 677–698.

O'Brien, E.A. (1991). "The Groups of Order 256", *J. Algebra*, **143**, 219–235.

O'Brien, E.A. (1994). "Computing automorphism groups of p-groups". To appear in Proceeding of CANT '92 (Sydney).

Rabin, M.O. (1958). "Recursive unsolvability of group theoretic problems", *Ann. Math.*, **67**, 172–194.

Schönert, Martin, *et al.* (1993). GAP – *Groups, Algorithms and Programming*. Lehrstuhl D. für Mathematik, RWTH, Aachen.

Schultz, Jonathan (1988). "A procedure for recognising groups of prime-power order". B.Sc. thesis, Australian National University.

Segal, Dan (1990). "Decidable properties of polycyclic groups", *Proc. London Math. Soc.* (3), **61**, 497–528.

Sylow, L. (1872). "Théorèmes sur les groupes de substitutions", *Math. Ann.*, **5**, 584–594.

Tietze, H. (1908). "Über die topologischen Invarianten mehrdimensionalen Mannigfaltigkeiten", *Monatsh. f. Math. u. Physik*, **19**, 1–118.

Wursthorn, Martin (1993). "Isomorphisms of modular group algebras: An algorithm and its application to groups of order 2^6", *J. Symbolic Comput.*, **15**, 211–227.