A Finite Soluble Quotient Algorithm

Alice C. Niemeyer

Abstract. An algorithm for computing power conjugate presentations for finite soluble quotients of predetermined structure of finitely presented groups is described. Practical aspects of an implementation are discussed.

1. Introduction

Polycyclic groups are characterised by the fact that they have a polycyclic series, which is a descending series of subgroups, such that each one is normal in the previous one and their quotient is cyclic (see Segal, 1983, or Sims, 1994). Polycyclic presentations describe polycyclic groups by exhibiting a polycyclic series of the group. They are very important for computing in the group since they allow the practical computation (by collection) of a normal word for every element in the group. Algorithms using such descriptions for finite polycyclic groups are, for example, described in Laue et al. (1984) and form an integral part of the computational group theory systems Magma (see Bosma and Cannon, 1993) and GAP (Schönert et al., 1994).

Every polycyclic group is soluble and every finite soluble group is polycyclic. However, not every soluble group is polycyclic. Baumslag, Cannonito, and Miller (1981a, 1981b) describe an algorithm which decides whether a soluble group given by a finite presentation is polycyclic. It has been partly implemented by Sims (1990). Here attention is focused on finite soluble groups. Polycyclic presentations for finite soluble groups are better known as power conjugate presentations.

The task of a finite soluble quotient algorithm is to compute power conjugate presentations for finite soluble groups described as quotients of finitely presented groups. A number of proposals for finite soluble quotient algorithms have been made, for instance by Wamsley (1977), by Leedham-Green (1984) and by Plesken (1987). The last algorithm has been implemented by Wegner (1992).

Algorithms for computing power conjugate presentations for $p$-groups or for nilpotent groups described as quotients of finitely presented groups exist. For $p$-groups see for example Havas and Newman (1980) or Celler et al. (1993) and for nilpotent groups see Sims (1994) or Nickel (submitted).
Here a new finite soluble quotient algorithm is presented in detail. For a brief description of the algorithm see Niemeyer (to appear). New features are the use of vector enumeration and the intermediate presentations considered. Leedham-Green (1984) suggested the use of a vector enumerator in this context.

We now turn to the background required for the description of the algorithm. Let $G$ be a finite soluble group and let $G = G_0 \geq G_1 \geq \cdots \geq G_n = \{1\}$ be a composition series for $G$ with factors of prime order. Choose elements $a_i \in G$ for $1 \leq i \leq n$ such that $G_{i-1} = \langle G_i, a_i \rangle$; let $p_i$ be the order of the factor $G_{i-1}/G_i$. Then $\mathcal{A} = \{a_1, \ldots, a_n\}$ is a generating set for $G$. Choose words $v_{jk}$ in the elements $a_{j+1}, \ldots, a_n$ for $1 \leq j \leq k \leq n$ such that $a_i^{p_i} = v_{ii}$ for $1 \leq i \leq n$ and $a_i^{a_j} = v_{jk}$ for $1 \leq j < k \leq n$ and let $\mathcal{R}$ be the set consisting of these relations. Then $\mathcal{R}$ is a defining set of relations for $G$. The presentation $\{\mathcal{A} \mid \mathcal{R}\}$ is a power conjugate presentation for $G$. A power conjugate presentation $\{\mathcal{A} \mid \mathcal{R}\}$ of a group $G$ exhibits the composition series $G = G_0 \geq G_1 \geq \cdots \geq G_n = \{1\}$, where $G_{i-1} = \langle a_1, \ldots, a_n \rangle$ for $1 \leq i \leq n$. The order of $G$ is at most $\prod_{i=1}^{n} p_i$ and therefore $G$ is finite. A word $w(a_1, \ldots, a_n)$ in the generators is normal if it is of the form $a_1^{e_1} \cdots a_n^{e_n}$ with $0 \leq e_i < p_i$. Note that we only consider words in the elements of $\mathcal{A}$, that is these words do not contain inverses of the generators.

In what follows “word” means semigroup word. A normal word $u$ in $\mathcal{A}$ is equivalent to a word $w$ in $\mathcal{A}$ if $u$ and $w$ are the same element of the group defined by $\{\mathcal{A} \mid \mathcal{R}\}$. The fundamental importance of power conjugate presentations arises from the observation that, given a word $w$ in the generators $\mathcal{A}$, the power conjugate presentation $\{\mathcal{A} \mid \mathcal{R}\}$ can be used to compute an equivalent normal word. It is assumed that the words $v_{jk}$ in a power conjugate presentation are normal. If the right hand side of some relation is the identity then the relation is written as a relator by just listing the left hand side.

If a word is not normal it has a non-normal subword minimal in the partially ordered set of non-normal words. This subword is of the form $a_i^p$ or of the form $a_k a_j$ with $k > j$. Collection of a word consists of a sequence of steps each of which chooses a minimal non-normal subword and replaces it. A subword of the form $a_i^p$ is replaced by $v_{ii}$ and $a_k a_j$ is replaced by $a_j v_{jk}$. For normal words the sequence of collection steps is empty. Otherwise a minimal non-normal subword is chosen and replaced. Each further step is applied to the result of the previous step. The words resulting from these steps are equivalent to the given word. For an account of collection see for example Havas and Nicholson (1976), or Leedham-Green and Soicher (1990).

A fundamental result is that every collection (independent of the choices of minimal non-normal subwords) of a non-normal word results in a normal word after a finite number of steps (see for example Sims, 1994). A normal word resulting from collecting the word $w$ will be denoted $(w)$; it may depend upon the choices made in the process. Multiplication of two elements of $G$ amounts to computing a normal word for the product given by concatenation.

In general, there may exist many normal words representing a given group element. If each element is represented by a unique normal word, then the power conjugate presentation is consistent. In this case two group elements are equal only if they are
represented by the same normal word. For the finite soluble group $G$, given as above, the order is then equal to $\prod_{i=1}^{n} p_i$.

The following result is due to Wamsley (1977). In summary it states that a power conjugate presentation is consistent if certain words, called consistency test words, can be collected in “sufficiently different” ways and still yield the same normal word. These consistency test words are $a_i a_j a_k$ with $1 \leq i < j < k \leq n$, $a_i a_k$, with $1 \leq j < k \leq n$, $a_j a_k$, with $1 \leq i < j \leq n$, and $a_i a_j a_k$ with $1 \leq i < j \leq n$.

**Theorem 1.1.** Let $G$ be a finite soluble group given by the power conjugate presentation $\{A \mid R\}$. Then the presentation $\{A \mid R\}$ is consistent if and only if the following equations hold:

\[
\begin{align*}
\left( (a_k a_j) a_i \right) &= \left( a_k (a_j a_i) \right) \quad \text{for } 1 \leq i < j < k \leq n, \\
\left( (a_k^p) a_j \right) &= \left( a_k^{p-1} (a_k a_j) \right) \quad \text{for } 1 \leq j < k \leq n, \\
\left( (a_j a_i) a_k^{p-1} \right) &= \left( a_j (a_i^p) \right) \quad \text{for } 1 \leq i < j \leq n, \\
\left( (a_k^p) a_i \right) &= \left( a_k (a_i^p) \right) \quad \text{for } 1 \leq i \leq n.
\end{align*}
\]

In practice we are interested in describing groups by consistent power conjugate presentations. In this context power conjugate presentations which exhibit a refinement of a specific series of a given group are considered. The algorithm described here can be viewed as a generalisation of the $p$-quotient algorithm described by Havas and Newman (1980) and by Celler et al. (1993). It also has features in common with the nilpotent quotient algorithm described by Nickle (submitted). Let $G$ be a group and $p$ a prime. The prime quotient algorithm works with a series

$G = P_0^p(G) \geq P_1^p(G) \geq \cdots$ with $P_i^p(G) = [P_{i-1}^p(G), G](P_{i-1}^p(G))^p$ for $i \geq 1$

called the lower exponent-$p$ central series of $G$. If there exists an integer $c \geq 0$ such that $P_c^p(G) = \langle 1 \rangle$, then $G$ is a $p$-group and the smallest such integer is called the exponent-$p$ class of $G$. It repeats a basic step which, given a consistent power conjugate presentation of $G/P_i^p(G)$, computes a consistent power conjugate presentation of $G/P_{i+1}^p(G)$.

In the next section a series is defined that takes the role of the lower exponent-$p$ central series in the context of finite soluble groups. The power conjugate presentations exhibiting this series are described.

2. The soluble $\mathcal{L}$-series

Let $G$ be a group. Let $\mathcal{L} = [(p_1, c_1), \ldots, (p_k, c_k)]$ be a list of pairs consisting of a prime, $p_i$, and a non-negative integer, $c_i$, with $p_i \neq p_{i+1}$ and $c_i$ positive for $i < k$. For $1 \leq i \leq k$ and $0 \leq j \leq c_i$ define the list $\mathcal{L}_{i,j} = [(p_1, c_1), \ldots, (p_{i-1}, c_{i-1}), (p_i, j)]$. Set $\mathcal{L}_{1,0}(G) = G$. For $1 \leq i \leq k$ and $1 \leq j \leq c_i$ let

$\mathcal{L}_{i,j}(G) = P_j^p(\mathcal{L}_{i,0}(G))$

and for $1 \leq i < k$ let

$\mathcal{L}_{i+1,0}(G) = \mathcal{L}_{i, c_i}(G)$
and \( \mathcal{L}(G) = \mathcal{L}_{k,c_k}(G) \). Note that \( \mathcal{L}_{i,j}(G) \geq \mathcal{L}_{i,j+1}(G) \) holds for \( j < c_i \).

The chain of subgroups

\[
G = \mathcal{L}_{1,0}(G) \geq \mathcal{L}_{1,1}(G) \geq \cdots \geq \mathcal{L}_{1,c_1}(G) = \mathcal{L}_{2,0}(G) \geq \cdots \geq \mathcal{L}_{k,c_k}(G) = \mathcal{L}(G)
\]

is called the \( \mathcal{L} \)-series of \( G \). If \( \mathcal{L}(G) = \langle 1 \rangle \) then \( G \) is an \( \mathcal{L} \)-group. If \( c_k > 0 \) and \( \mathcal{L}(G) \neq \langle 1 \rangle \) for \( \mathcal{L} = [(p_1, c_1), \ldots, (p_k, c_k - 1)] \) then \( G \) is a strict \( \mathcal{L} \)-group.

Note that in the definition of strict \( \mathcal{L} \)-group the exponent-\( p_k \) class of \( \mathcal{L}_{k,0}(G) \) is determined but not necessarily the exponent-\( p_1 \) class of \( \mathcal{L}_{i,0}(G) \). For every finite soluble group \( G \) there exists a (not necessarily unique) list \( \mathcal{L} \) such that \( G \) is a strict \( \mathcal{L} \)-group.

For a given \( i \) the series \( \mathcal{L}_{i,0}(G) \geq \cdots \geq \mathcal{L}_{i,c_i}(G) \) is an initial segment of the lower exponent-\( p_i \) central series of \( \mathcal{L}_{i,0}(G) \) and \( \mathcal{L}_{i,0}(G)/\mathcal{L}_{i,c_i}(G) \) is a \( p_i \)-group of exponent-\( p_i \) class at most \( c_i \).

We use the following notation.

\[
\mathcal{L}^+ = \left\{ \begin{array}{ll} 
[(p_1, c_1), \ldots, (p_k, c_k + 1)] & \text{if } p = p_k, \\
[(p_1, c_1), \ldots, (p_k, c_k), (p, 1)] & \text{if } p \neq p_k.
\end{array} \right.
\]

Then \( \mathcal{L}(G)/\mathcal{L}^+(G) \) is an elementary abelian \( p \)-group. Further

\[
\mathcal{L}^- = \left\{ \begin{array}{ll}
[(p_1, c_1), \ldots, (p_{k-1}, c_{k-1})] & \text{if } p = p_k, \\
[(p_1, c_1), \ldots, (p_k, c_k)] & \text{if } p \neq p_k.
\end{array} \right.
\]

Let \( \{\mathcal{A} \mid \mathcal{R}\} \) be a power conjugate presentation for a finite soluble group \( H \) with \( \mathcal{A} = \{a_1, \ldots, a_n\} \). Let \( d \) be the minimal number of generators in \( \mathcal{A} \) required to generate \( H \). Assume there exists a \( d \)-element subset \( \mathcal{X} \) of \( \mathcal{A} \) such that \( \mathcal{X} \) generates \( H \) and for each generator \( a \in \mathcal{A} \setminus \mathcal{X} \) there is at least one relation of \( \mathcal{R} \) having \( a \) as the last generator on the right hand side and occurring with exponent 1. Choose exactly one of these relations and call it the definition of \( a \). The fact that this relation is the definition of \( a \) is emphasised by using ’:=’ instead of ’=’ in the relation. The colon is on the same side of the relation as the generator defined by this relation. The presentation \( \{\mathcal{A} \mid \mathcal{R}\} \) together with chosen definitions for the generator in \( \mathcal{A} \setminus \mathcal{X} \) is called labelled. Let \( G \) be a group with generating set \( \{g_1, \ldots, g_b\} \) and \( \tau \) an epimorphism of \( G \) onto \( H \). For \( i = 1, \ldots, b \) let \( w_i \) be the normal word equivalent to \( \tau(g_i) \). If \( a \in \mathcal{X} \) is the last generator in at least one \( w_i \) occurring with exponent 1 and we have chosen one such \( w_i \), we write \( \tau(g_i) := w_i \) and call this the definition of \( a \). For each \( a \in \mathcal{X} \) there is a maximal \( k \) and a maximal \( c \) such that \( a \in \mathcal{L}_{k,c}(H) \). We call \( \tau \) a labelled epimorphism if each generator \( a \in \mathcal{X} \) has a definition \( \tau(g_i) =: w_i \) and \( a \) is the only generator which occurs in \( w_i \) and does lie in \( \mathcal{L}_{k,c}(H) \). If \( \{\mathcal{A} \mid \mathcal{R}\} \) is a labelled power conjugate presentation for the group \( H \) and \( \tau \) a labelled epimorphism from \( G \) to \( H \), then every generator in \( \mathcal{A} \) has a definition either as an image under \( \tau \) or as a relation in \( \mathcal{R} \). Further we can read off a preimage in \( G \) for each \( a_k \in \mathcal{A} \) under \( \tau \). Thus we can compute a preimage in \( G \) for each \( h \in H \) under \( \tau \).
The following is a labelled consistent power conjugate presentation for $S_4$, the symmetric group on 4 letters:

\[
\{ a, b, c, d \mid a^2 = c, \\
\quad b^p = b^2 c, \quad b^3, \\
\quad c^a = c, \quad c^b = d, \quad c^2, \\
\quad d^a = c d, \quad d^b = c d, \quad d^c = d, \quad d^2 \}\.
\]

The relations $a^2 = c$ and $c^b = d$ are the definitions of $c$ and $d$, respectively. Note that this notation implicitly characterises the set $\mathcal{X}$ as the subset of $\mathcal{A}$ whose elements do not occur as the last element of a right hand side of a definition. In the example, $\mathcal{X}$ is the set $\{a, b\}$.

Consider the group $G$ having the finite presentation

\[
\{ x, y \mid x^8, y^3, (x^{-1}y)^2, (yx^3y)^2 = x^4 \}
\]

and let $L$ be the list $[(2, 1), (3, 1), (2, 1)]$. Then the map $\tau$ from $G$ to $S_4$ defined by $\tau(x) := a$ and $\tau(y) := b$ is a labelled epimorphism. The elements of $\mathcal{X}$ have definitions as images of $\tau$, whereas the other elements in $\mathcal{A}$ have definitions in the relations of the power conjugate presentation for $S_4$.

### 3. The $L$-covering group

Let $L$ be the list $[(p_1, c_1), \ldots, (p_k, c_k)]$, where $c_i$ is a non-negative integer for $1 \leq i \leq k$, and let $K$ be an $L$-group with generator number $d$. Let $F$ be the free group of rank $d$ and let $\theta$ be an epimorphism of $F$ onto $K$. Let $p$ be a prime and denote $L^{-p}(K)$ by $P$ and let $F_P$ be the preimage in $F$ of $P$ under $\theta$.

Let $K$ be a finite strict $L$-group with generator number $d$. A group $H$ is a $p$-descendant of $K$ if $H$ has generator number $d$ and $H$ is an $\tilde{L}$-group, where for some non-negative integer $n$

\[
\tilde{L} = \left\{ \begin{array}{ll}
\{(p_1, c_1), \ldots, (p_k, c_k + n)\} & \text{if } p = p_k, \\
\{(p_1, c_1), \ldots, (p_k, c_k), (p, n)\} & \text{if } p \neq p_k,
\end{array} \right.
\]

and $H/\tilde{L}_{k,c_k}(H)$ is isomorphic to $K$. If $H$ is a strict $L^{+p}$-group, that is $n = 1$, then $H$ is an immediate $p$-descendant of $K$. Note that if $K$ is a $p$-group these definitions are the same as in O’Brien (1990).

**Theorem 3.1.** Let $K$ be a finite strict $L$-group with generator number $d$ and $p$ a prime. There exists an $L^{+p}$-group $\hat{K}$ with generator number $d$ such that every immediate $p$-descendant of $K$ is isomorphic to a quotient of $\hat{K}$.

**Proof.** Let $F$ be the free group of rank $d$ and let $R$ be the kernel of the epimorphism $\theta$ of $F$ onto $K$. Let $F_P$ be the preimage under $\theta$ in $F$ of $P = L^{-p}(K)$. Define $S$ to be $[R, F_P]R_p$ and define the group $\hat{K}$ to be $F/S$. Then $S \leq R$ and $\hat{K}$ is a $d$-generator group. Let $H$ be an immediate $p$-descendant of $K$ and let $\nu$ be an epimorphism of $H$ onto $K$. Then there exists an epimorphism $\hat{\theta}$ from $F$ to $H$ such that $\hat{\theta}\nu = \theta$. Since
\[ R \hat{\theta} \leq \ker \nu \] it follows that \( R \hat{\theta} \) is an elementary abelian \( p \)-subgroup of \( H \), which is central in \( \mathbb{F}_p \). Hence \( S \hat{\theta} = [R \hat{\theta}, F_p \hat{\theta}]^p(R \hat{\theta})^p \) is the identity in \( H \) and thus \( H \) is a homomorphic image of \( F/S \).

If \( p \) is not \( p_k \) then \( P \) is the identity and \( F_p \) is \( R \). Therefore \( S = [R, R]^{R_p} \) and \( R/S \) is the relation module of \( F/R \) (see Gruenberg, 1976).

The group \( K = F/S \) with \( S = [R, F_p]^{R_p} \) is the \( \mathcal{L} \)-covering group of \( K \) with respect to the prime \( p \). A consistent power conjugate presentation \( \{ A \mid R \} \) for \( K \) is an \( \mathcal{L} \)-covering presentation of \( \{ A \mid R \} \). It should always be clear from the context which prime \( p \) is chosen. Therefore \( K \) is called the \( \mathcal{L} \)-covering group of \( K \) without reference to \( p \). If \( \mathcal{L} \) is the list \( \{(p, c)\} \) then an \( \mathcal{L} \)-group \( K \) is a \( p \)-group and the \( \mathcal{L} \)-covering group \( \hat{K} \) is the \( p \)-covering group \( K^* \) of \( K \). O’Brien (1990) shows that \( K^* \) is isomorphic to \( F/[R, F]^{R_p} \).

The \( \mathcal{L} \)-covering group \( F/S \) is the largest extension of \( F/R \) by an elementary abelian \( p \)-group such that the extension has the same generator number as \( F/R \) and \( F_p/R \) acts trivially on the elementary abelian \( p \)-group. Note that the generator number of \( F_p/S \) may be larger than the generator number of \( F_p/R \). The \( p \)-covering group \( (F_p/R)^* \) of \( F_p/R \) is the largest extension of \( F_p/R \) by an elementary abelian \( p \)-group which has the same generator number as \( F_p/R \) and \( F_p/R \) acts trivially on the elementary abelian \( p \)-group.

The following theorem asserts that the isomorphism type of \( F/S \) is independent of the choice of the homomorphism \( \theta \) from \( F \) to \( K \) and thereby independent of its kernel \( R \). It is valid also for the case that \( \mathcal{L}^{-p}(F/R) \) is trivial.

**Theorem 3.2.** Let \( R_1 \) and \( R_2 \) be normal subgroups of \( F \) such that \( F/R_1 \) and \( F/R_2 \) are \( \mathcal{L} \)-groups. Furthermore, let \( U_1 \) and \( U_2 \) be subgroups of \( F \) such that for \( i \in \{1, 2\} \)

1. \( R_i \leq U_i \),
2. \( U_i/R_i \leq \mathcal{L}^{-p}(F_p/R_i) \),
3. \( U_i/R_i \) is characteristic in \( F/R_i \),

and there exists an isomorphism \( \varphi \) of \( F/R_1 \) onto \( F/R_2 \), which maps \( U_1/R_1 \) onto \( U_2/R_2 \). Define \( S_i \) to be \( [R_i, U_i]^{R_i} \) for \( i \in \{1, 2\} \). Then \( F/S_1 \) is isomorphic to \( F/S_2 \) by an isomorphism which takes \( R_1/S_1 \) to \( R_2/S_2 \) and \( U_1/S_1 \) to \( U_2/S_2 \).

**Proof.** Let \( \{a_1, \ldots, a_d\} \) be a free generating set for \( F \). Let \( \nu \) be the canonical epimorphism of \( F/S_1 \) onto \( F/R_1 \). Then \( \nu \varphi \) is an epimorphism from \( F/S_1 \) onto \( F/R_2 \). Let \( b_i \in F \) such that \( (b_iS_1)\nu \varphi = a_iR_2 \). Define a homomorphism \( \rho: F \to F/S_1 \) mapping \( a_i \) to \( b_iS_1 \). The map \( \rho \nu \varphi \) is an epimorphism from \( F \) onto \( F/R_2 \). Since \( \rho \nu \varphi \) agrees on each \( a_i \) with the natural projection of \( F \) onto \( F/R_2 \) and since the \( a_i \) generate \( F \), \( \rho \nu \varphi \) is the natural projection. Thus \( R_2\rho \nu \varphi = 1 \), and \( R_2 \rho \leq \ker \nu \varphi = R_1/S_1 \) and \( U_2 \rho \nu \varphi = U_2/R_2 \), so \( U_2 \rho \leq (U_2/R_2)\nu^{-1} \varphi^{-1} = (U_1/R_1)\nu^{-1} = U_1/S_1 \). Therefore \( S_2 \rho = ([R_2, U_2]^{R_2}) \rho \) is a subgroup of \( [R_1/S_1, U_1/S_1](R_1/S_1)^p \). Since the elements of \( R_1/S_1 \) are of order \( p \) and commute with \( U_1/S_1 \), we have that \( S_2 \leq \ker \rho \). Hence \( F/S_1 \) is isomorphic to a factor group of \( F/S_2 \). Similarly \( F/S_2 \) is isomorphic to a factor group of \( F/S_1 \), and therefore \( F/S_1 \cong F/S_2 \).
If the subgroup $U_i$ is chosen to be $L^{-p}(F_p/R_i)$ then the theorem asserts that given a $d$-generator $L$-group $G$ with $P = L^{-p}(G)$, the choice of the epimorphism $\theta : F \to G$ and thus the choice of $R = \ker\theta$ does not influence the isomorphism type of $F/S$. We can also choose $U_i = R_i$ and thus the choice of the epimorphism $\theta$ also has no impact on the isomorphism type of $F/[R,R]R^p$.

4. An $L$-covering algorithm

The task of the $L$-covering algorithm is to determine a labelled consistent power conjugate presentation for the $L$-covering group of a soluble $L$-group $K$.

- The input of the $L$-covering algorithm is a labelled consistent power conjugate presentation for a soluble $L$-group $K$ and a prime $p$.
- The output is a consistent power conjugate presentation for the soluble group $\hat{K}$, the $L$-covering group of $K$ with respect to the prime $p$.

The $L$-covering algorithm presented here first computes a finite presentation for $\hat{K}$. It is shown that one can define a normal form for the elements in $\hat{K}$ and that the finite presentation can be used like a power conjugate presentation to compute normal forms of elements of $\hat{K}$. Applying a theorem which is a generalisation of Theorem 1.1 allows the determination of a module presentation for the kernel of the natural epimorphism of $\hat{K}$ onto $\hat{K}$. A vector space basis for this $F_p\hat{K}$-module is computed by an algorithm, called vector enumeration, and this in turn enables the determination of a labelled consistent power conjugate presentation for $\hat{K}$.

The individual steps of the algorithm are illustrated by reference to the example of the symmetric group on 4 letters, $S_4$. A consistent power conjugate presentation for this group was given in Section 2. For this example let $L$ be the list $[(2,1), (3,1), (2,1)]$ and let $p$ be the prime 2.

4.1. A finite presentation for the $L$-covering group. A finite presentation $\{\tilde{A} \mid \tilde{R}\}$ for $\hat{K} = F/S$ is obtained in the following way. Let $\{A \mid R\}$ be the labelled consistent power conjugate presentation for the $L$-group $K$, where $A$ is the set $\{a_1, \ldots, a_n\}$ and

$$R = \{a_i^{p^k} = v_{ii}, a_k^{a_j} = v_{jk} \mid 1 \leq i \leq n, 1 \leq j < k \leq n\}.$$

Let $P$ be $L^{-p}(K)$. Assume that $\{A \mid R\}$ is a power conjugate presentation for $K$ with respect to a composition series which refines the soluble $L$-series. Therefore there exists an $r$ such that $\{a_1P, \ldots, a_rP\}$ generates $K/P$ and $\{a_{r+1}, \ldots, a_n\}$ generates $P$. Let $s$ be the number of relations in $R$ which are not definitions. Then $s = (n - 1)n/2 + d$.

Introduce new generators $\{y_1, \ldots, y_s\}$ and define $\tilde{A} = \{a_1, \ldots, a_n\} \cup \{y_1, \ldots, y_s\}$. We obtain $\tilde{R}$ in the following way:

1. initialise $\tilde{R}$ to contain all relations of $R$ which are definitions;
(2) modify each non-defining relation $a_i^{p_i} = v_{ii}$ or $a_j^{a_j} = v_{jk} y_t$ for some $t \in \{1, \ldots, s\}$, where different non-defining relations are modified by different $y_t$, and add the modified relation to $\tilde{R}$;

(3) add to $\tilde{R}$ all relations of the form $[y_i, y_j^g] = 1$ for all normal $g = w(a_1, \ldots, a_r)$ and $y_i^p = 1$ for $1 \leq i, j \leq s$;

(4) add to $\tilde{R}$ all relations $y_i^{a_j} = y_i$ for $j > r$ and $1 \leq i \leq s$.

We apply this to the example of $S_4$. The subgroup $L_3,0(S_4)$ is a 2-group isomorphic to the Klein 4-group and is generated by $c$ and $d$. Further, $S_4$ is generated by $a$ and $b$. The definition of $c$ is the relation with left-hand side $a^2$ and the definition of $d$ is the relation with left-hand side $c^b$. We obtain the following presentation for $\hat{S}_4$:

$$\tilde{A} = \{a, b, c, d, y_1, y_2, y_3, y_4, y_5, y_6, y_7, y_8\} \text{ and } \tilde{R} = \{a^2 = c, b^a = b^2 c y_1, b^3 = y_2, c^a = c y_3, c^b = d, c^2 = y_4, d^a = c d y_5, d^b = c d y_6, d^c = d y_7, d^2 = y_8, y_i^c = y_i, \text{ for } 1 \leq i \leq 8, y_i^d = y_i, \text{ for } 1 \leq i \leq 8, [y_i, y_j^g] = 1 \text{ for } 1 \leq i, j \leq 8 \text{ and } g \in \{a, ab, b, b^2, ab^2\} \}.$$ 

Consider the following diagram.

$$\begin{array}{c}
\text{F} \\
\downarrow \text{F}_P \\
\text{P} \\
\downarrow \\
\text{M} \\
\downarrow \text{R} \\
\text{K} \\
\text{S}
\end{array}$$

The subgroup $R/S$, denoted by $M$, is the kernel of the natural epimorphism of $\hat{K}$ to $K$ and can be characterised as follows. It is the maximal $\mathbb{F}_p K$-module by which $K$ can be extended so that $P$ acts trivially on $M$ and the extension has the same generator number as $K$. Thus $M$ is an $\mathbb{F}_p (K/P)$-module. Let $Y$ be the free $\mathbb{F}_p (K/P)$-module on $\{y_1, \ldots, y_8\}$. The module $M$ is a homomorphic image of $Y$. The kernel of the homomorphism from $Y$ onto $M$ can be computed effectively. In order to see this, we study the finite presentation for the group $\hat{K}$ in more detail.

4.2. Collecting in $\hat{K}$. One can collect in the group $\hat{K}$ relative to $\{\tilde{A} \mid \tilde{R}\}$. The definition of a normal word can be generalised for this presentation in the following way. A word in $\tilde{A}$ is normal if it is of the form $w(a_1, \ldots, a_n) \cdot \Pi_{i=1}^s f_i$, where $w(a_1, \ldots, a_n)$ is a normal word in $\{a_1, \ldots, a_n\}$ and $f_i$ is an element of $\mathbb{F}_p (K/P)$. The following steps,
referred to as “collection in \( \hat{K} \)”, can be applied to every word in \( \tilde{A} \). For \( f, f' \in F_p(K/P) \) and \( 1 \leq k, l \leq s \) and \( 1 \leq i, j \leq n \)

1. replace \( y_i^f y_i^{f'} \) by \( y_i^k y_i^{k'} \) for \( k < l \);
2. replace \( y_i^f y_i^{f'} \) by \( y_i^{f+f'} \);
3. (a) replace \( y_i^f a_i^q \) by \( a_i y_i^{f(a_i)} a_i^{q-1} \) if \( q > 1 \);
   (b) replace \( y_i^f a_i \) by \( a_i y_i^{f(a_i)} \);
4. replace \( a_i^{n} \) by \( v \), where \( a_i^{n} = v \) is a relation in \( \hat{R} \);
5. replace \( a_j a_i \) by \( a_i v \), where \( a_j^{a_i} = v \) is a relation in \( \hat{R} \) for \( i < j \).

In each step a word is replaced by another word representing the same element of \( \hat{K} \). After applying a finite number of these steps to any word it is replaced by a normal word. This can be proved in a way similar to proving that a collection process computes a normal word after applying finitely many collection steps. Rules 1), 2) and 3) use the fact that \( M \) is an \( F_p(K/P) \)-module. Note that 4) and 5) resemble collection steps in a collection algorithm, where the power conjugate presentation is used to determine the replacement.

For example the word \( b(ba) \) in collects in \( S_4 \) to \( abdy_1^{b+1}y_2y_3y_6y_7 \).

The following lemma states that two equivalent normal words can differ only by a module word in \( Y \).

**Lemma 4.1.** Let \( w \) be an arbitrary word in \( \{a_1, \ldots, a_n\} \cup \{y_1, \ldots, y_s\} \). Then there exists a unique normal word \( v \) in \( \{a_1, \ldots, a_n\} \) such that any normal word in \( \hat{K} \) equivalent to \( w \) has the form \( v \cdot \Pi_{i=1}^{n} y_i^{f_i} \).

**Proof.** The existence of the unique normal word \( v \) follows from the fact that \( \{A | R\} \) is a consistent power conjugate presentation for \( K \). \( \square \)

A consequence of this lemma is that the map \( \phi : \hat{K} \to K \) which maps a word in \( \hat{K} \) to its unique normal word in \( \{a_1, \ldots, a_n\} \) is an epimorphism.

The following theorem allows us to describe the kernel of the homomorphism from \( Y \) onto \( M = R/S \) in a manner suitable for computation. It considers certain non-normal words in \( \hat{K} \).

**Theorem 4.2.** Let \( Y \) be the free \( F_p(K/P) \)-module on \( \{y_1, \ldots, y_s\} \) and \( \{\tilde{A} | \tilde{R}\} \) the presentation for the extension \( \hat{K} \) of \( K \) as defined above. Let \( W \) be the following set of consistency test words in \( \{a_1, \ldots, a_n\} \):

\[
\begin{align*}
&(((a_k a_j) a_i) (a_k (a_j a_i))^{-1}) & \text{for} & 1 \leq i < j < k \leq n, \\
&(((a_k^p) a_j) (a_k^{p-1}(a_k a_j))^{-1}) & \text{for} & 1 \leq j < k \leq n, \\
&(((a_j a_i) a_k^{p-1}) (a_j (a_k^p))^{-1}) & \text{for} & 1 \leq i < j \leq n, \\
&(((a_i^p) a_j) (a_i (a_i^p))^{-1}) & \text{for} & 1 \leq i \leq n \\
\end{align*}
\]

and let \( T \) be the set of elements obtained by collecting the words in \( W \) with respect to \( \{\tilde{A} | \tilde{R}\} \). Then \( T \) consists of words in \( Y \) and \( M \) is isomorphic to \( Y/(TF_p(K/P)) \).
Proof. The elements of $W$ represent the identity element in $\hat{K}$. By Lemma 4.1 the elements of $T$ are words in $Y$. Denote $T\mathbb{F}_p(K/P)$ by $\langle T \rangle$. Let $\mu : Y \to M$ be the epimorphism mapping $y_i$ in $Y$ to $y_i$ in $M$. Since the elements of $W$ are the identity in $\hat{K}$ it follows that the elements of $T$, when viewed as elements of $Y$, are mapped to the identity element of $M$, hence $\langle T \rangle \subseteq \ker \mu$. Therefore $Y/\langle T \rangle$ has a factor module isomorphic to $M$.

We now define a consistent power conjugate presentation for $K$ extended by $Y/\langle T \rangle$ such that the extension is a $d$-generator group. Since $M$ is the largest $\mathbb{F}_p(K/P)$-module with these properties it follows that $Y/\langle T \rangle$ is isomorphic to $M$.

Let $\{b_1, \ldots, b_m\}$ be a vector space basis for $Y/\langle T \rangle$. Then each element $y_i\langle T \rangle$ can be expressed uniquely in the basis elements. Therefore we obtain a power conjugate presentation $\{\hat{A} \mid \hat{R}\}$ for an extension of $K$ by $Y/\langle T \rangle$ in the following way from the presentation $\{\hat{A} \mid \hat{R}\}$ where $\hat{A}$ is $A \cup \{b_1, \ldots, b_m\}$:

1. replace every occurrence of an element $y_i$ on the right hand side of a relation in $\hat{R}$ with left hand side a word in $A$, by the corresponding word for $y_i\langle T \rangle$ in the basis and add this modified relation to $\hat{R}$;
2. add to $\hat{R}$ the relations $b_i^{a_j} = w_i(b_1, \ldots, b_m)$ for all $1 \leq i \leq m$ and $1 \leq j \leq r$, where $w_i(b_1, \ldots, b_m)$ is determined by the action of $a_j$ on $b_i$;
3. add to $\hat{R}$ the relations $b_i^{a_j} = b_i$ for $1 \leq i \leq m$ and $r+1 \leq j \leq n$;
4. add to $\hat{R}$ the relations $b_j^{b_i} = b_j$ for $1 \leq i < j \leq m$ and the relations $b_i^{b_i} = 1$ for $1 \leq i \leq m$.

We now show that $\{\hat{A} \mid \hat{R}\}$ is consistent. Let $\hat{K}$ denote the group defined by $\{\hat{A} \mid \hat{R}\}$. Since the elements of $W$ collect to elements of $\langle T \rangle$ they are the trivial word in $\hat{K}$.

The consistency of $\{\hat{A} \mid \hat{R}\}$ is proved by applying Theorem 1.1. We only need to consider consistency relations which involve at least one element of the basis. Any consistency relation which involves only basis elements holds, since the basis is a basis for a vector space over $\mathbb{F}_p$.

Consider the word $b_k a_j a_i$. Applying a collection step to $(b_k a_j a_i)$ with respect to $\{\hat{A} \mid \hat{R}\}$ yields $a_j b_k^{a_i} a_i$ which collects to $a_j a_i (b_k^{a_i})^{a_i}$ and finally to $a_i v_{ij} (b_k^{a_i a_i})$. On the other hand $b_k (a_j a_i)$ collects to $b_k a_i v_{ij}$ which in turn collects to $a_i v_{ij} b_k^{a_i - v_{ij}}$. Since $(b_k^{a_i a_i})$ and $b_k^{a_i - v_{ij}}$ are the same module element they have the same normal form in the basis.

Consider the word $b_k b_j a_i$. Applying a collection step to $(b_k b_j a_i)$ with respect to $\{\hat{A} \mid \hat{R}\}$ yields $b_j b_k a_i$ which collects to $a_j b_j^{a_i} b_k^{a_i}$. On the other hand $b_k (b_j a_i)$ collects to $b_k a_i b_j^{a_i}$ which collects to $a_i b_k^{a_i} b_j^{a_i}$. Therefore the first consistency relation in Theorem 1.1 holds. Similarly one can prove that the other relations also hold and therefore $\{\hat{A} \mid \hat{R}\}$ is a consistent power conjugate presentation.

In collecting the words in $W$ with respect to the presentation $\{\hat{A} \mid \hat{R}\}$ we obtain a set $T$ which generates the kernel of the epimorphism of the free $\mathbb{F}_p(K/P)$-module $Y$ onto
the module $M$. In the example $T$ is the set:

\[
\{ y_2, \ y_1^{(1+b+b^2)}y_3y_5y_6y_7, \ y_3^{(a+1)}, \ y_3y_6y_7^{(a+1)}, \\
y_3^by_7, \ y_3y_6y_7^{(1+b)}, \ y_2, \ y_2y_3y_4^{(a+1)}y_6, \\
y_1^{(1+a+b^2)}y_2y_3y_4^{(1+b)}, \ y_5^{(1+b)}y_7, \\
y_5^{(1+b)}y_6y_7, \ y_5^{(a+b)}y_6y_7^{b}, \ y_6y_7^{(1+a)}, \\
y_2y_3y_4^{(1+b+2)}y_5^{(a+b)}y_6^{(1+b)}y_7^{b}, \ y_6^{(1+b)} \}.
\]

In the proof of the previous theorem it was assumed that we have

1. vector space basis $\{ b_1, \ldots, b_m \}$ for the $\mathbb{F}_p (K/P)$-module $Y/\langle T \rangle$;
2. an expression in the basis for $b_i^a$ for $1 \leq i \leq m$ and $1 \leq j \leq r$;
3. an expression in the basis for $y_i\langle T \rangle$ for $1 \leq i \leq s$.

Where such information is available the proof yields a constructive method to obtain a consistent power conjugate presentation $\{ \hat{A} \mid \hat{R} \}$ for $\hat{K}$. We now describe an algorithm which may be used to obtain this information.

4.3. Computing a vector space basis for a module. The technique of vector enumeration is used to compute a basis for the $\mathbb{F}_p (K/P)$-module $M$ needed to obtain a power conjugate presentation for $F/S$. A vector enumeration algorithm is described in Linton (1991) and Linton (1993). Its use in this context has been suggested by Leedham-Green (private communication, 1991).

It is used with the following input:

1. a consistent power conjugate presentation for $K$;
2. the set of free generators for $Y$;
3. a set $T$.

The output is:

1. an $\mathbb{F}_p$-basis $\{ b_1, \ldots, b_m \}$ for $M$;
2. the matrix action of each generator of $K/P$ in the power conjugate presentation of $K/P$ on $M$ with respect to the computed basis;
3. an expression in the computed basis for $y_i\langle T \rangle$ for $1 \leq i \leq s$.

This output is used to obtain a consistent power conjugate presentation for the extension $\hat{K}$ of $K$ by $M$ using the method described in the proof of Theorem 4.2.

We illustrate the technique by reference to our example. The vector enumerator with input $\{ A \mid R \}$, the set $\{ y_1, y_2, y_3, y_4, y_5, y_6, y_7 \}$ and $T$ as above computes a module basis for the module $M$. The basis has five elements $\{ e, f, g, h, i \}$ defined by $e = y_1$, $f = y_2$, $g = y_3$, $h = e^a$ and $i = e^b$. Further the vector enumerator gives the action of $a$ and $b$ on the module basis, while the elements $c$ and $d$ act trivially. The information returned by the vector enumerator can be used to construct the following consistent power conjugate presentation for the $L$-covering group $\hat{S}_4$:
\{a, b, c, d, e, f, g, h, i, j \mid a^2 = c, b^a = b^2 c e, b^3, c^a = c, c^b = d, c^2 = f, d^a = cdg, d^b = cdh, d^c = dgh, d^2 = i, e^a = j, e^b = j, e^c = e, e^d = e, e^2, f^a = f, f^b = i, f^c = f, f^d = f, f^e = f, f^2, g^a = fh, g^b = hi, g^c = g, g^d = g, g^e = g, g^f = g, g^2, h^a = fg, h^b = gi, h^c = h, h^d = h, h^e = h, h^f = h, h^g = h, h^2, i^a = fghi, i^b = fghi, i^c = i, i^d = i, i^e = i, i^f = i, i^g = i, i^h = i, i^2, j^a = e, j^b = efghi, j^c = j, j^d = j, j^e = j, j^f = j, j^g = j, j^h = j, j^i = j, j^2\).

From the presentation we can read off that \(S_4\) has order \(2^9 3 = 1536\). The group \(L^{-2}(S_4)\) has order \(2^8\) and is generated by \(\{c, d, e, f, g, h, i, j\}\). It is the direct product of the normal subgroups \(\langle c, d \rangle\) and \(\langle e, j, fgi \rangle\) of \(S_4\). In general, the complete preimage \(\hat{P}\) of \(P = L^{-p}(K)\) is a normal subgroup of \(L^{-p}(\hat{K})\). It contains a normal subgroup, namely \(\hat{P} \cap M\). Note that \(r\) was defined such that the subset \(\{a_{r+1}, \ldots, a_n\}\) of \(A\) generates \(P\). A generating set for \(\hat{P} \cap M\) is the union of the set \(\{v_{ii} \mid a^p = v_i\} is a relation in \(\hat{R}\) for \(i > r\}\) and the set \(\{v_{ij} \mid a^p_i = a, v_{ij}\} is a relation in \(\hat{R}\) for \(i, j > r\}\). It can thus be obtained from the power conjugate presentation for \(\hat{K}\).

The group ring \(\mathbb{Z}_p(K/P)\) in the example is isomorphic to \(\mathbb{Z}_2 S_3\). We can investigate the module structure of \(M\) as an \(S_3\)-module further. The submodule \(\hat{P} \cap M\) is the direct sum of \(\langle fi, ghi \rangle\) and \(\langle gh \rangle\). Its module complement \(\langle e, j, fgi \rangle\) is a direct sum of a one dimensional and a two dimensional module. It has the decomposition \(\langle fgi \rangle \oplus \langle ej, efghi \rangle\).

### 4.4. Obtaining a labelled presentation

In some cases additional work is necessary to transform the consistent power conjugate presentation \(\{\hat{A} \mid \hat{R}\}\) of \(\hat{K}\) into a labelled presentation. It is possible that a basis vector \(b_i\) does not occur as the last element of the right hand side of a relation in \(\hat{R}\) and thus no relation can be chosen as the definition of \(b_i\). In this case we proceed as follows. For each basis vector \(b_i\) choose a relation which contains \(b_i\) in its right hand side as the definition of \(b_i\), ensuring that this relation is not chosen as the definition of any other basis vector. Assume that for the element \(b_i\) the right hand side of its defining relation has the form

\[w_1(a_1, \ldots, a_n) \cdot w_2(b_1, \ldots, b_{i-1}) \cdot w_3(b_i, \ldots, b_m)\).

Define the element \(\overline{b_i}\) to be \(w_3(b_i, \ldots, b_m)\). Obviously \(\{\overline{b_1}, \ldots, \overline{b_m}\}\) is again a vector space basis for \(M\) and the action of the generator \(a_j\) of \(K/P\) on this basis can be computed as the action of \(a_j\) on \(w_3(b_1, \ldots, b_m)\) and then expressing the result in the new basis. A labelled consistent power conjugate presentation for \(\hat{K}\) is obtained by performing a base change.
The power conjugate presentation for \( \hat{S}_4 \) given above is already a labelled power conjugate presentation, where \( c, d, e, f, g, h, i \) and \( j \) are defined by the relations with left hand sides \( a^2, c^b, b^a, c^2, d^a, d^b, d^2 \) and \( e^a \), respectively. Every extension of \( S_4 \) by an elementary abelian 2-group \( M \) such that the Klein 4-group acts trivially on \( M \) and the extension has generator number 2 is isomorphic to a quotient of the group defined by this presentation.

5. A soluble quotient algorithm

The soluble quotient algorithm presented here computes a power conjugate presentation for a quotient \( G/L(G) \) of a finitely presented group \( G \), where the presentation exhibits a composition series of the quotient group which is a refinement of the soluble \( L \)-series. It takes as input:

1. A finite presentation \( \{g_1, \ldots, g_b \mid r_1(g_1, \ldots, g_b), \ldots, r_m(g_1, \ldots, g_b)\} \) for \( G \);
2. A list \( \mathcal{L} = [(p_1, c_1), \ldots, (p_k, c_k)] \), where each \( p_i \) is a prime, \( p_i \neq p_{i+1} \), and each \( c_i \) is a positive integer.

The output is:

1. A labelled power conjugate presentation for \( G/L(G) \) exhibiting a composition series refining the soluble \( L \)-series of this quotient;
2. A labelled epimorphism \( \tau : G \rightarrow G/L(G) \).

The algorithm proceeds by computing power conjugate presentations for the quotients \( G/L_{i,j}(G) \) in turn. Without loss of generality assume that a power conjugate presentation for \( G/L_{i,j}(G) \) has been computed for \( j < c_i \). The basic step computes a power conjugate presentation for \( G/L_{i,j+1}(G) \). The group \( L_{i,j}(G)/L_{i,j+1}(G) \) is a \( p_i \)-group. The basic step takes as input:

1. The finite presentation for \( G \);
2. A labelled consistent power conjugate presentation for the finite soluble quotient \( K \cong G/L_{i,j}(G) \) of \( G \) with \( j < c_i \) which refines the \( L \)-series of \( K \);
3. A labelled epimorphism \( \theta : G \rightarrow K \).

The output is:

1. A labelled consistent power conjugate presentation for the finite soluble group \( H \cong G/L_{i,j+1}(G) \), exhibiting a composition series refining the \( L \)-series of \( H \);
2. An epimorphism \( \phi : H \rightarrow K \);
3. A labelled epimorphism \( \tau : G \rightarrow H \) with \( \tau \phi = \theta \).

If during the basic step it is discovered that \( L_{i,j}(G) = L_{i,j+1}(G) \), then \( L_{i+1,0}(G) \) is set to \( L_{i,j}(G) \).

The basic step is illustrated by the following diagram, where the input is described on the left and the output is described on the right. Put \( p = p_i \), let \( P \) denote \( L_{i,0}(K) \), and \( \hat{P} \) denote \( L_{i,0}(H) \). If \( j = 0 \) then \( P \) is trivial. The elementary abelian \( p \)-group \( \ker \phi \) is denoted by \( N \). The group \( \hat{P} \) acts trivially on \( N \); thus \( \hat{P} \) is a central extension of \( P \) by
$N$, and $\hat{P}$ is a $p$-group of exponent-$p$ class at most one larger than the exponent-$p$ class of $P$.

The subgroup $N = \ker \phi$ plays a role similar to that of the subgroup $M$ in the $L$-covering algorithm. It is the maximal $\mathbb{F}_p K$-module by which $K$ can be extended so that $P$ acts trivially on $N$ and the extension is an epimorphic image of $G$. Thus $N$ is an $\mathbb{F}_p (K/P)$-module. If $P$ is non-trivial the extension of $K$ by $N$ has the same generator number as $K$, because, by Burnside’s Basis Theorem (Huppert I, Satz 3.15, 1967), the generator number of the extension of $P$ by $N$ is already determined by the generator number of $P$. If $P$ is non-trivial the module $M$ is the largest $\mathbb{F}_p K$-module by which $K$ can be extended such that the extension has the same generator number as $K$ and $P$ acts trivially on $M$. Therefore $N$ is a factor module of $M$. If $P$ is trivial and $N$ is the largest $\mathbb{F}_p K$-module by which $K$ can be extended such that the extension is a homomorphic image of $G$, it does not follow that $N$ is isomorphic to a factor module of $M$, since the extension may have a larger generator number than $K$. However, in both cases we can write down a finite presentation for the extension.

This presentation is obtained as follows. Let $\{A \mid R\}$ be the supplied consistent power conjugate presentation for $K$, where $A = \{a_1, \ldots, a_n\}$ and

$$R = \{a_i^{p_i} = v_{ii}, a_k^{a_j} = v_{jk} \mid 1 \leq i \leq n, 1 \leq j < k \leq n\}.$$ 

Let $\{\tilde{A} \mid \tilde{R}\}$ with $\tilde{A} = \{a_1, \ldots, a_n, y_1, \ldots, y_s\}$ be the finite presentation for $\hat{K}$ as calculated by the $L$-covering algorithm. Then $G/L_{i,j+1}(G)$ is isomorphic to a quotient of $\hat{K}$, if $P$ is nontrivial. If $P$ is trivial, the generator number of $G/L_{i,j+1}(G)$ may be larger than the generator number of $G/L_{i,j}(G)$. Since $G/L_{i,j+1}(G)$ has a consistent power conjugate presentation refining its $L$-series it follows that any additional generators lie in $L_{i,j}(G)/L_{i,j+1}(G)$. Therefore $N$ is isomorphic to a quotient of the direct product $Z$ of the free $\mathbb{F}_p (K/P)$-module $Y$ and the free $\mathbb{F}_p (K/P)$-module on the additional generators. Let $t$ be the number of generators of $G$ whose images under $\theta$ are not definitions, then $t = b - d$, where $b$ is the number of generators of $G$ in the finite presentation and $d$ is the generator number of $K$. Add new generators $\{z_1, \ldots, z_t\}$ to $\tilde{A}$. The set of relations $\tilde{R}$ is modified in the following manner.
(1) add to \( \tilde{R} \) all relations of the form \([z_i, z_j^g] = 1\), \([y_k, z_j^g] = 1\) and \([z_i, y_k^g] = 1\) for all normal \( g = w(a_1, \ldots, a_r) \) for \( 1 \leq i, j \leq t \) and \( 1 \leq k \leq m \) and all relations \( z_i^p = 1 \) for \( 1 \leq i \leq t \);

(2) add to \( \tilde{R} \) all relations \( z_i^{g_j} = z_i \) for \( j > r \) for \( 1 \leq i \leq t \).

The group \( \tilde{K} \) defined by \( \{ \tilde{A} \mid \tilde{R} \} \) has \( G/L_{i,j+1}(G) \) as a factor group. It is called the \textit{extended} \( L \)-covering group of \( K \) and \( \{ \tilde{A} \mid \tilde{R} \} \) is the \textit{extended} \( L \)-covering presentation. Define a map \( \sigma \) from \( \{ g_1, \ldots, g_b \} \) to the group \( \tilde{K} \) by \( g_i^g = g_i^g z_k \) if \( g_i^g \) is non-defining and \( g_i^g = g_i^g \) if \( g_i^g \) is defining. The map \( \sigma \) is called the \textit{extended} map.

The basic step is illustrated by an example. Consider the group \( G \) defined by the following finite presentation \( \{ x, y \mid x^8, y^3, (x^{-1}y)^2, (yx^3yx)^2 = x^4 \} \). Let \( L \) be the list \([ (2, 1), (3, 1), (2, 2) \)\]. Then it can be shown that \( G/L_{3,1}(G) \) is isomorphic to \( S_4 \). A labelled consistent power conjugate presentation for \( S_4 \) was given above. The input for the basic step is the finite presentation for \( G \), the labelled consistent power conjugate presentation for \( S_4 \) and the epimorphism \( \theta : G \to G/L_{3,1}(G) \) defined by \( x \mapsto a \) and \( y \mapsto b \). The images of \( \theta \) are the definitions of \( a \) and \( b \), respectively. We have previously determined a presentation for \( S_4 \). This is also the extended \( L \)-covering presentation since both images of \( \theta \) are definitions. The map \( \sigma \) is the map from \( G \) to \( \tilde{K} \) which maps \( x \) to \( a \) and \( y \) to \( b \). Using the map \( \sigma \) the kernel of the homomorphism from \( Z \) onto \( N \) can be computed effectively.

**Theorem 5.1.** Let \( T \) be the set of elements defined in Theorem 4.2. Let \( U \) be the set \( \{ r_i(g_i^g, \ldots, g_b^g) \mid 1 \leq i \leq m \} \) of elements of \( Z \) obtained by evaluating the relators of \( G \) in the images of the generators of \( G \) under the map \( \sigma \). Then \( N \) is isomorphic to \( Z/((T \cup U) \mathbb{F}_p(K/P)) \).

**Proof.** Consider the factor group \( H \) of \( \tilde{K} \) obtained by extending the group \( K \) by \( Z/((T \cup U) \mathbb{F}_p(K/P)) \). Then \( H \) is generated by \( g_i^g, \ldots, g_b^g \). Since the relations of \( G \) hold in \( H \) it follows that it is a homomorphic image of \( G/L_{i,j}(G) \). By construction \( H \) has \( G/L_{i,j}(G) \) as a homomorphic image, hence \( H \) is isomorphic to \( G/L_{i,j}(G) \).

In our example \( U \) is the set \( \{ y_1^3 y_3^{(bab)} y_1^b y_3^b y_7^{(1+b)} \} \).

The vector enumerator was used to compute a vector space basis for the module \( M \) in the \( L \)-covering algorithm. Here it is employed to compute a vector space basis for the module \( N = Z/((T \cup U) \mathbb{F}_p(K/P)) \). It takes as input

1. a consistent power-conjugate presentation for \( K \);
2. the set of generators for \( Z \);
3. the set of relations \( T \cup U \).

The output is

1. an \( \mathbb{F}_p \)-basis for \( N \);
2. an expression in this basis for the image under the generators of \( K/P \) of every basis element;
expressions for the images of the $\mathbb{F}_p(K/P)$-generators of $Z$ in terms of the basis elements.

This output is used to obtain a consistent power conjugate presentation for the extension $H$ of $K$ by $N$, an epimorphism $\tau$ from $G$ to $H$ and an epimorphism $\phi$ from $H$ to $K$. The method for constructing the consistent power conjugate presentation is again the method described in the proof of Theorem 4.2. The homomorphism $\tau$ from $G$ to $H$ is obtained by replacing the elements $z_i$ in the map $\sigma$ by the corresponding word in the basis for $N$. This yields an epimorphism by Theorem 5.1. As pointed out earlier a base change for the vector space basis of the module $N$ may be necessary in order to obtain a labelled consistent power conjugate presentation for $H$. A base change may also be necessary in order to transform $\tau$ into a labelled homomorphism. The map $\tau$ is an epimorphism, since all the generators of $H$ are either defined as images of the generators of $G$ under $\tau$ or by definitions in the power conjugate presentation of $H$ on the images of those generators.

The vector enumerator is employed to compute a vector space basis for the module $Z/(T \cup U)\mathbb{F}_2(K/P)$ in the previous example. The vector enumerator returns the basis $\{e, f, g\}$ defined by $e = y_3, f = y_4$ and $g = y_5$. Again, the vector enumerator gives the action of $a$ and $b$ on this basis, while $c$ and $d$ act trivially. The information is used to construct for the quotient $H = G/\mathcal{L}_{4,0}(G)$ the following labelled consistent power conjugate presentation:

\[
\{ a, b, c, d, e, f, g \mid \\
a^2 = c, \\
b^a = b^a c, b^b = c, \quad c^a = d, \quad c^2 = e, \\
d^a = c d f, d^b = c d g, d^c = d f g, \quad d^2 = e f, \\
e^a = e, e^b = e f, e^c = e, e^d = e, \quad e^2 = e, \\
f^a = e g, f^b = e f g, f^c = f, f^d = f, f^e = f, f^2, \\
g^a = e f, g^b = e, g^c = g, g^d = g, g^e = g, g^f = g, g^2 \}
\]

and the labelled epimorphism $\tau$ from $G$ onto $G/\mathcal{L}_{4,0}(G)$ defined by $x \mapsto a$ and $y \mapsto b$.

Hence in this example $G$ has a homomorphic image isomorphic to a factor group of $\hat{S}_4$. In fact $G$ is an extension of $S_4$ by a group $N$ of order $2^3$. The group $N$ is generated by $\{c^2, d^2, [d, c]\}$ and therefore $\mathcal{L}^{-2}(H)$ is isomorphic to the 2-covering group of the Klein-4 group.

6. Practical aspects

This section focuses on practical aspects of our algorithm. First we highlight some features of an implementation.

6.1. The implementation. The soluble quotient algorithm as described in Section 5 has been implemented in C and this first implementation is known as the ANU Soluble Quotient Program (SQ). It uses Version 3 of Linton’s implementation of a vector
enumerator (see Linton 1991 and 1993) and Nickel’s parser for finite presentations as used in his Nilpotent Quotient Program, (see Nickel 1992).

The basic step of the algorithm first computes a finite presentation \( \tilde{A} | \tilde{R} \) for the extended \( L \)-covering group \( \tilde{K} \) of \( K \) (given by the power conjugate presentation \( A | R \)), where \( \tilde{A} = A \cup \{ y_1, \ldots, y_s \} \cup \{ z_1, \ldots, z_t \} \). The number \( s + t \) of extra generators for \( \tilde{A} \) has a serious impact on the performance of the algorithm. Related quotient algorithms face the same problem. Methods to reduce the number of extra generators in the context of a \( p \)-quotient algorithm and a nilpotent quotient algorithm are described in Celler et al. (1993) and Nickel (1993 and submitted), respectively. Similar ideas can be used in our algorithm when repeatedly computing \( L \)-covering groups for a fixed prime \( p \). These features can be invoked by an option given to the program (SQ 1).

Theorem 6 uses the set of normal words \( T \cup U \). Recall that a word in \( \tilde{A} \) is normal if it is of the form \( w(a_1, \ldots, a_n) \cdot \Pi_{i=1}^{s} y_{f_i}^{f_i} \), where \( f_i \) is an element of \( F_p(K/P) \). We implement the steps referred to as “collection in \( \tilde{K} \)” in Section 4.2 following the strategy of collection from the left (see Leedham-Green and Soicher, 1990). In representing \( f_i \) one may write it as a sum over \( F_p \) of either arbitrary or normal words in the generators of the consistent power conjugate presentation of \( K/P \). This can again be determined by an option given to the program (SQ 2). The influence of the options is demonstrated in Section 6.4.

6.2. The test examples. We highlight certain aspects of the behaviour of SQ using the following example presentations.

\[ P_1 \{a, b \mid (ab)^2b^{-6}, a^4b^{-1}ab^{-9}a^{-1}b\}, \]
\[ P_2 \{a, b \mid a^2ba^{-1}ba^{-1}b^{-1}ab^{-2}, a^2b^{-1}ab^{-1}aba^{-2}b^2\}, \]
\[ P_3 \{a, b \mid ab^2(ab^{-1})^2, (a^2b)^2a^{-1}ba^2(bab)^{-1}\}, \]
\[ P_4 \{a, b \mid ab^2a^{-1}b^{-1}ab^2, ba^2b^{-1}a^{-1}ba^2\}, \]
\[ P_5 \{a, b \mid ab^3a^{-1}b^{-1}ab^3, ba^3b^{-1}a^{-1}ba^3\}, \]
The first seven presentations appear in Wegner (1992); presentations $P_1$, $P_2$, and $P_3$ were constructed by Kenne (1990) and $P_4$, $P_5$ and $P_6$ appear in Campbell (1975). Presentation $P_8$ arose during Neubüser's study of the Heineken group (Neubüser and Sidki, 1988). Presentations $P_9$, $P_{10}$, and $P_{11}$ were constructed in the study of $B(2,6)$ (see Havas, Newman and Niemeyer (in preparation)). Presentations $P_{12}$, $P_{13}$ and $P_{14}$ define soluble groups of derived length 3. They arise as semidirect products of the multiplicative and additive groups of finite fields extended by the Frobenius automorphism. They have been suggested by Pasechnik.

6.3. The $L$-series of the test examples. An important problem in applying the soluble quotient algorithm is to choose an appropriate $L$-series to determine a factor group of a finitely presented group. The program Quotpic, by Holt and Rees (1992), has proved to be valuable in solving this problem. It is an interactive, graphical tool which, among other things, incorporates very practical routines for manipulating presentations and provides access to various other programs. It was employed to compute the $L$-series for some of the test examples.

One possible method of determining an $L$-series for a finite soluble quotient of a finitely presented group $G$ using Quotpic is the following. A prime $p$ in the list of abelian invariants of $G$ is chosen and the ANU $p$-Quotient Program (see Newman and O'Brien, in
preparation) is called to compute a power conjugate presentation for a $p$-quotient $G/K$ of desired exponent $p$-class. A presentation for $K$ can be obtained by Reidemeister-Schreier rewriting. Now the process can be repeated with $K$ taking the role of $G$.

Obviously it is only possible to compute presentations for the kernel if the quotient of the group over the kernel is small. This is the case in presentations $P_8$, $P_9$, $P_{10}$ and $P_{11}$, where the 3-quotient has a kernel of index at most $3^2$ in the group defined by the presentation. For presentations $P_{12}$, $P_{13}$ and $P_{14}$ the $L$-series were known beforehand.

### 6.4. Performance

Table 1 gives the CPU time in seconds taken on a DEC 3000/600 AXP (with 96Mb RAM) to compute consistent power conjugate presentations for the test examples. For each presentation we list the $L$-series that determines the finite soluble quotient computed and its order.

| $P$ | Order | $L$-series | SQ | SQ 1 | SQ 2 |
|-----|-------|-------------|-----|------|------|
| 1   | $2^1 \cdot 3^4$ | [(2,1),(3,1),(2,2),(3,2)] | 2.1 | 7.6  | 4.0  |
| 2   | $2^5 \cdot 3^5$ | [(2,1),(3,1),(2,2),(5,1)] | 2.9 | 29.6 | 8.7  |
| 3   | $2^3 \cdot 3^5$ | [(3,1),(2,2),(5,2)] | 4.1 | 4.0  | 2.4  |
| 4   | $2^8 \cdot 3^3$ | [(3,1),(2,2),(5,1),(11,1)] | 27.9 | 129.6 | 60.0 |
| 5   | $2^3 \cdot 6^3$ | [(2,3),(3,2)] | 0.5 | 0.4  | 0.4  |
| 6   | $2^3 \cdot 5^3 \cdot 11$ | [(3,2),(2,2)] | 0.6 | 0.5  | 0.6  |
| 7   | $2^9 \cdot 3^4$ | [(3,1),(2,2),(3,2),(2,1)] | +1220 | +1260 | +1411 |
| 8   | $2^{26} \cdot 3^3$ | [(3,1),(2,11)] | 988.3 | 348.3 | 1021.1 |
| 9   | $2^{252} \cdot 3^3$ | [(3,2),(2,3)] | 1453.5 | 347.7 | 2646.2 |
| 10  | $2^{182} \cdot 3^3$ | [(3,2),(2,2)] | 77.8 | 58.3 | 81.6  |
| 11  | $2^{398} \cdot 3^3$ | [(3,2),(2,2)] | 473.1 | 336.6 | 675.2 |
| 12  | $2^5 \cdot 31$ | [(5,1),(31,1),(2,1)] | 0.3 | 0.3  | 0.4  |
| 13  | $2^{17} \cdot 7 \cdot 127$ | [(7,1),(127,1),(2,1)] | 31.5 | 31.4 | 36.4  |
| 14  | $2^6 \cdot 3^5$ | [(2,2),(3,1),(5,1),(2,1),(3,1)] | 0.6 | 0.6  | 0.7  |

Some entries contain the symbol ‘+’. In these cases the computations were not completed. The vector enumerator invoked by SQ, SQ 1 and SQ 2 ran out of space after the time listed for presentation $P_7$.

Some remarks on the space requirements of SQ and the vector enumerator are in order. SQ 1 allocated less than 1 Mb for almost all presentations; it needed 1.19 Mb for $P_{10}$, 2.6 Mb for $P_9$, and 4.8 Mb for $P_{11}$. These space requirements are for the SQ 1 only and do not list the requirements for the vector enumerator. The vector enumerator, for example, allocated 3.6 Mb for the computation of the module of dimension 163 in $P_{10}$ when called from SQ 1, 54.8 Mb for the module of dimension 205 in $P_9$ when called from SQ 2, but only 9.7 Mb when called from SQ 1.

### 6.5. Comparisons

Let us first compare the different versions of our algorithm. SQ 1 performs best when the presentation describes a group with large $p$-quotients; for example for presentations $P_8$, $P_9$, $P_{10}$ and $P_{11}$. If the order of the computed quotient involves many different primes, and each occurs to a small power, SQ appears to perform...
better; for example for presentations $P_1$, $P_2$ and $P_4$. This shows that the number of extra generators has an influence on the performance of the vector enumerator. When repeatedly computing $\mathcal{L}$-covering groups for a fixed prime the presentations generated by SQ 1 seem to be more suitable for the vector enumerator than those computed by SQ. However, in examples where SQ 1 performs worse than SQ it spends additional time in the vector enumerator. This indicates that the time is not lost in the methods to reduce the number of extra generators, but that the presentations computed are worse for the vector enumerator.

The performance of SQ 2 seems to lie between that of SQ and SQ 1. An interesting feature is that where it performs slower than SQ the additional time is spent in the vector enumerator and not in collection. Thus handing the vector enumerator normal words does not necessary seem to be better.

If one wants to show that a finitely presented group with a soluble quotient $G$ does not have a larger soluble quotient as an extension of $G$ by a module $N$ over $\mathbb{F}_p$, then the theoretical bound on the dimension of $N$ is generally very large. Presentation $P_{14}$ describes a group $G$ of order $2^6 \cdot 3 \cdot 5$. Hence it does not describe an extension of $G$ by a module over $\mathbb{F}_3$. Here SQ can prove that this is the case in 0.6 seconds.

6.6. Conclusions. The performance of the vector enumerator is critical for the soluble quotient algorithm presented here. As pointed out earlier, its performance depends on the module presentations supplied as input. It seems promising to investigate more thoroughly how a presentation should be supplied as input. Further it seems promising to develop methods for reducing the number of generators in the module presentations. For example, Leedham-Green (1984) has suggested using special power conjugate presentations. Recently various aspects of these power conjugate presentations were investigated by Eick (1993) and Cannon and Leedham-Green (in preparation). Special power conjugate presentations can be used to reduce the number of new generators if the soluble quotient is calculated by computing maximal nilpotent factors. These suggestions need to be tested. In fact, when extending a soluble group by an elementary abelian $p$-group, it is not necessary to introduce new generators for those relations whose left hand sides only involve generators whose power relations involve a prime not equal to $p$. In that case, however, one has to add new generators to all images of the epimorphism of the finitely presented group onto the soluble quotient so far computed and use Fox derivatives, see Leedham-Green (1984). The method he describes computes a module larger than the one required. In fact, when extending a soluble group $K$ this method computes a module which has a direct summand $\mathbb{Z}_p^{[K]} - 1$, which seems difficult for large $K$.

Verifying that a certain quotient group of a finitely presented group is indeed the largest finite soluble quotient seems to be a very difficult problem in practice. Leedham-Green (1984) suggested using an integer vector enumerator to determine the primes in the next nilpotent factor. Linton has written a version of the vector enumerator that works over the integers. Plesken (1987) also describes a method to determine the primes that
can occur in a soluble quotient of a finitely presented group. Again, these ideas need to tested.

6.7. The Code. The ANU Solvable Quotient Program (Version 1.0) is available via email from the author (alice@maths.uwa.edu), by anonymous ftp from maths.anu.edu.au in the directory pub/SQ, and as a share library with GAP 3.4. It might also become available as part of MAGMA.

ACKNOWLEDGMENTS

I thank my PhD supervisor Dr M.F. Newman for his generous support and assistance; I thank Dr L.G. Kovács, Dr C.R. Leedham-Green, Dr Werner Nickel and Dr E.A. O’Brien for many encouraging discussions and generous help. I thank Dmitrii V. Pasechnik for many helpful discussions on the last section.

I acknowledge the support of an OPRSA and an ANU PhD scholarship during which this work was carried out. Part of the writing was supported by ARC Grant A69230241.

References

[1] Baumslag, G., Cannonito, F.B., Miller, C.F. III (1981a). “Some recognizable properties of solvable groups”, Math. Z., 178, 289–295.

[2] Baumslag, G., Cannonito, F.B., Miller, C.F. III (1981b). “Computable algebra and group embeddings.”, J. Algebra, 69, 186–212.

[3] Campbell, C.M. (1975). Applications of the Todd-Coxeter Coset Enumeration Algorithm, PhD thesis. University of St Andrews.

[4] Bosma, W., Cannon, J.J. (1993). “Handbook of MAGMA functions” Department of Pure Mathematics, Sydney University.

[5] Cannon, J.J., Leedham-Green, C.R. (in preparation).

[6] Celler, F., Newman, M.F., Nickel, W., Niemeyer, A.C. (1993). “An algorithm for computing quotients of prime-power order for finitely presented groups and its implementation in GAP”, Research Report.

[7] Eick, B. (1993). “Spezielle PAG-Systeme im Computeralgebrasystem GAP”, Diplomarbeit. RWTH-Aachen.

[8] Gruenberg, K.W. (1976). Relation modules of finite groups, Published for the Conference Board of the Mathematical Sciences by the American Mathematical Society.

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

[10] Havas, G., Newman, M.F., Niemeyer, A.C. (in preparation). “Finiteness proofs for groups of exponent six”, in preparation.

[11] Havas, G., Nicholson, T. (1976). “Collection”, SYMSAC ’76 (Proc. ACM Sympos. on Symbolic and Algebraic Computation, Yorktown Heights, New York, 1976), 9–14.

[12] Holt, D.F., Rees, S. (1994). “A graphics system for displaying finite quotients of finitely presented groups.”, Proceedings of the DIMACS Workshop on Groups and Computation, AMS-ACM.

[13] Huppert, B. (1967). Endliche Gruppen I. Grundlehren Math. Wiss., 134, Springer-Verlag, Berlin, Heidelberg, New York.

[14] Kenne, P.E. (1990). “Some new efficient soluble groups”, Comm. Algebra, 14, 2747–2753.

[15] Laue, R., Neubüser, J., Schoenwaelder, U. (1984). “Algorithms for Finite Soluble Groups and the SOGOS System”, Computational Group Theory, (Durham, 1982), pp. 105–135. Academic Press, London, New York.
[16] Leedham-Green, C.R. (1984). “A Soluble Group Algorithm”, *Computational Group Theory*, (Durham, 1982), pp. 85–101. Academic Press, London, New York.

[17] Leedham-Green, C.R., Soicher, L.H. (1990). “Collection from the left and other strategies”, *J. Symbolic Comput.*, 9, 665–675.

[18] Linton, S.A. (1990). “On Vector Enumeration”, *Linear Algebra and Applications*, 192, 235–248.

[19] Linton, S.A. (1991). “Constructing Matrix Representations of Finitely Presented Groups”, *J. Symbolic Comput.*, 12(4 & 5), 427–438.

[20] Neubüser, J., Sidki, S. (1988). “Alguns Procedimentos Computacionais Para Grupos Dados Por Uma Apresentação Finita”, *Mathemática Universitaria*, No. 7, 77–120, Rio de Janeiro

[21] Newman, M.F., O’Brien, E.A. (in preparation). “Application of computers to questions like those of Burnside II”.

[22] Nickel, W. (1992). “A Nilpotent Quotient Program (Version 1.1d)”, available by anonymous ftp from maths.anu.edu.au in the directory pub/NQ, and as a share library with GAP 3.4.

[23] Nickel, W. (1993). *Central Extensions of Polycyclic Groups*, PhD thesis. The Australian National University.

[24] Nickel, W. (submitted). “Computing Nilpotent Quotients of Finitely Presented Groups”, Proceedings DIMACS Workshop, March 1994.

[25] Niemeyer, A.C. (to appear). “Computing Finite Soluble Quotients”, *Proceedings of CANT ’92*

[26] Niemeyer, A.C. (1993). *Computing Presentations For Finite Soluble Groups*, PhD thesis. The Australian National University.

[27] O’Brien, E.A. (1990). “The p-group generation algorithm”, *J. Symbolic Comput.*, 9, 677–698.

[28] Plesken, W. (1987). “Towards a Soluble Quotient Algorithm”, *J. Symbolic Comput.*, 4, 111–122.

[29] Schönert, M. *et al.* (1994). *GAP – Groups, Algorithms and Programming*. RWTH, Aachen: Lehrstuhl D für Mathematik.

[30] Segal, D. (1983). *Polycyclic Groups*. Cambridge University Press, New York.

[31] Sims, C.C. (1990). “Implementing the Baumslag-Cannonito-Miller Polycyclic Quotient Algorithm”, *J. Symbolic Comput.*, 9(5 & 6), 707–723.

[32] Sims, C.C. (1994). *Computation with finitely presented groups*. Cambridge University Press.

[33] Wamsley, J.W. (1977). “Computing soluble groups”, A. Dold, B. Eckmann (Ed.), *Group Theory*, Lecture Notes in Math., 573, (Canberra, 1975), pp. 118–125. Springer-Verlag.

[34] Wegner, A. (1992). *The Construction of Finite Soluble Factor Groups of Finitely Presented Groups and its Application*, PhD thesis. St. Andrews.

DEPARTMENT OF MATHEMATICS, UNIVERSITY OF WESTERN AUSTRALIA, NEDLANDS, WA 6009, AUSTRALIA