

Some remarks on the computation of complements and normalizers in soluble groups

F. Celler, J. Neubüser, C.R.B. Wright

Dedicated to Wolfgang Gaschütz, to whom we owe so much of our understanding of soluble groups, on the occasion of his seventieth birthday

1. Introduction.

The three lectures “Computing in Soluble Groups” given by the second author as part of the “Seminar on Computational Algebra” at the Department of Mathematics of the University of Rome “Tor Vergata” on March 22./23., 1990 embedded a report on some recent work on the topic mentioned in the title of this paper into a survey of the state of the art in that wider field. In this paper a reminiscence of that overview will be confined to some very sketchy historical remarks in this introduction, but the bibliography should still be sufficient to obtain a reasonable coverage of the literature on the field by tracing back.

For investigations that involve calculation with the elements of a finite group the elements must be represented in a form that allows efficient multiplication, inversion and, last but not least, comparison. The first programs of that kind, for the determination of the subgroup lattice, already used both permutations for arbitrary groups and, for p -groups, ordered words in a generating set corresponding to a central series with cyclic factors [Neu 61]. The systematic use in such programs of basic notions from permutation group theory, in particular the stabilizer chain, was initiated by C. Sims in three fundamental papers in the late sixties and made generally available with many refinements and extensions in the precursors and early versions of the CAYLEY system (cf. [Can 84] for a comprehensive overview). On the other hand a systematic use of the properties of composition and chief series of finite soluble groups for the investigation of their structure only started with an algorithm for the determination of conjugacy classes of p -groups [FeN 79] and the subsequent development of the SOGOS system [LNS 84], preceded by the invention [Mac 74], further development, and very successful application [New 76] of the Nilpotent Quotient Algorithm, which constructs central series of p -factor groups of a finitely presented group. Computing methods for soluble groups have recently received

much interest: collection methods for the multiplication of the ordered words mentioned above that had been believed to be fairly well understood for some time [HaN 76] have been reinvestigated with remarkable success [L-GS 90][Va-L 90], a number of rather powerful new algorithms have been described [GIS 90], [MeN 89], [Con 90], applications to explicit classifications of p -groups have been made [O’Br 90], p -group and soluble group techniques have entered into more general situations [Hol 84], [Sim 90b], and a number of proposals have been made for the generalisation of the NQA to a “Soluble Quotient Algorithm” [Ple 87], [Sim 90a].

In this paper we describe methods for calculating the classes of complements of a given normal subgroup of a soluble group and their connection to the calculation of the normalizer of a given subgroup. These methods can in some sense, on which we comment at the end of the paper, be viewed as counterparts to the algorithms for the determination of conjugacy classes of elements given in [MeN 89]. They supplement the algorithm for the determination of the normalizer described in [GIS 90] in those cases in which the latter has to fall back on the general “orbit-stabilizer” algorithm outlined in section 3 of [LNS 84]. The algorithms described in this paper have been implemented by the first author as part of the GAP system [NNS 88], which, among many other things, contains in a much better organized form almost all that had previously been incorporated into SOGOS.

2. Basic definitions.

To make the paper reasonably self-contained we repeat once more the basic definitions, referring to [LNS 84] for details.

Let $G = G_0 > \dots > G_n = \langle 1 \rangle$ be a composition series of the finite soluble group G with factors $G_{i-1}/G_i = \langle g_i G_i \rangle$ of order some prime p_i . Then $\langle g_1, \dots, g_n \rangle$ is a generating sequence, called a *PAG sequence*, with a defining set of relations

$$\begin{aligned} g_i^{p_i} &= w_{ii}(g_{i+1}, \dots, g_n) && \text{for } 1 \leq i \leq n \\ [g_i, g_j] &= w_{ij}(g_{j+1}, \dots, g_n) && \text{for } 1 \leq j < i \leq n. \end{aligned}$$

We may assume that in addition a subsequence $N_1 = G_0, N_2 = G_{i_2}, \dots, N_j = G_{i_j}, \dots, N_m = G_n$ of the composition series forms a normal series with elementary abelian factors, or even a chief series of G .

Each element of G can be uniquely expressed in the form

$$g = g_1^{\nu_1} \dots g_n^{\nu_n} \quad \text{with } 0 \leq \nu_i < p_i,$$

and multiplication can be performed by a collection process using the relations.

If $\nu_i = 0$ for $i = 1, \dots, k-1$ and $\nu_k \neq 0$, we call $\nu_k =: \lambda(g)$ the *leading exponent* and $k =: w(g)$ the *weight* of g . With respect to (g_1, \dots, g_n) each nontrivial subgroup $U \leq G$ has a unique *canonical generating sequence* (u_1, \dots, u_s) , abbreviated CGS, with the following properties

- (1) (u_1, \dots, u_s) is a PAG sequence for U ,
- (2) $w(u_i) > w(u_j)$ for $i > j$,
- (3) $\lambda(u_i) = 1$ for $i = 1, \dots, s$,
- (4) $\nu_{w(u_i)}(u_j) = 0$ for $i \neq j$.

Given any generating set of a subgroup $U \leq G$, its CGS can be determined by a “noncommutative Gauß algorithm”.

The use of “homomorphism principles” has been recommended in section 3 of [LNS 84]. We shall apply them in the following form.

Let G be a group acting on a set Ω and let $N \triangleleft G$. Let $\omega \in \Omega$. Then an orbit ω^N is a block for G . Let $Stab_G(\omega)$ be the stabilizer of ω and $Stab_G(\omega^N) = \{g \mid g \in G, \omega^g \in \omega^N\}$ the block stabilizer of ω^N . For each $b \in Stab_G(\omega^N)$ there exists $n_b \in N$ such that $\omega^b = \omega^{n_b}$, that is, $bn_b^{-1} \in Stab_G(\omega)$. Hence with $Stab_{G/N}(\omega^N) = \langle b_1N, \dots, b_kN \rangle$ and $Stab_N(\omega) = \langle n_1, \dots, n_l \rangle$ we have

$$Stab_G(\omega) = \langle b_1n_{b_1}^{-1}, \dots, b_kn_{b_k}^{-1}, n_1, \dots, n_l \rangle.$$

In our applications we will be able to compute $Stab_{G/N}(\omega^N)$ and for given $bN \in Stab_{G/N}(\omega^N)$ compute an element n_b such that $bn_b^{-1} \in Stab_G(\omega)$.

3. The first cohomology group.

Let G be an arbitrary group with finite presentation

$$G = \langle g_1, \dots, g_n \mid R_j(g_1, \dots, g_n) = 1, j = 1, \dots, r \rangle,$$

let M be an elementary abelian p -group of rank d on which $g \in G$ acts by $g : m \rightarrow m^g$. As is well known, the group of 1-cocycles can be defined as

$$Z^1 = \{\gamma : G \rightarrow M \mid \gamma(gg') = \gamma(g)^{g'} \gamma(g'), \quad \forall g, g' \in G\},$$

the group of 1-coboundaries as

$$B^1 = \{\gamma_m : G \rightarrow M \mid \exists m \in M, \forall g \in G, \gamma_m(g) = m m^{-g}\},$$

and the 1-cohomology group $H^1(G, M) := Z^1/B^1$.

The definition of Z^1 makes clear that γ is uniquely determined by its values on the generators g_1, \dots, g_n of G , i.e. that $\beta : \gamma \rightarrow (\gamma(g_1), \dots, \gamma(g_n))$ is a monomorphism of Z^1 into the n -th direct power M^n of M , on which G acts componentwise. We will in fact calculate $\beta(Z_1)$ and $\beta(B_1)$ using the well-known interpretation of H^1 in a split extension. Let

$$G \bowtie M = \{(g, m) \mid g \in G, m \in M, (g_1, m_1)(g_2, m_2) = (g_1 g_2, m_1^{g_2} m_2)\}.$$

Then the set of complements of $M^* = \{(1, m) \mid m \in M\}$ is described as

$$\mathcal{K} = \{K_\gamma = \{(g, \gamma(g)) \mid g \in G\} \mid \gamma \in Z^1(G, M)\}$$

and two complements K_{γ_1} and K_{γ_2} are conjugate in $G \bowtie M$ if and only if $\gamma_1 \in B^1(G, M)\gamma_2$ so that there is a bijection between the conjugacy classes of complements and the elements of $H^1(G, M)$.

On the other hand, if

$$1 \rightarrow M \xrightarrow{id} H \xrightarrow{\pi} G \rightarrow 1$$

is an exact sequence and τ a mapping from $\{g_1, \dots, g_n\}$ into H with $g_i^{\tau\pi} = g_i$, then this extension splits if and only if there exists a mapping $f : \{g_1, \dots, g_n\} \rightarrow M$ such that for all $j = 1, \dots, n$

$$(1) \quad R_j(g_1^\tau f(g_1), \dots, g_n^\tau f(g_n)) = 1,$$

a complement being given by

$$\langle g_i^\tau f(g_i) \mid i = 1, \dots, n \rangle.$$

We want to determine these functions f by solving (1) for $(f(g_1), \dots, f(g_n)) \in M^n$. Since $M \triangleleft H$, we may in each relation “collect” the elements g_i^τ to the left, without changing their order and we get

$$(2) \quad R_j(g_1^\tau, \dots, g_n^\tau) R_j^*(f(g_1), \dots, f(g_n)) = 1$$

where $R_j(g_1^\tau, \dots, g_n^\tau) \in M$ and where R_j^* is a map from M^n into M with $R_j^*(f(g_1), \dots, f(g_n))$ a product of conjugates of the elements $f(g_i)$ under the elements g_j^τ . If we write M additively, i.e. as a $\mathbf{Z}_p G$ -module of dimension d , conjugation by an element g_j^τ becomes application of a linear map G_j corresponding to g_j , and after introduction of a basis of M becomes right-multiplication of rows by a matrix that we also denote by G_j . R_j^* becomes a linear map from the sum of n copies of this module into M , and hence with respect to suitable bases is represented by a $dn \times d$ matrix. If we express each $f(g_i)$ as a linear combination of the basis with unknown coefficients x_{i1}, \dots, x_{id} , then we see that each relation (2) yields an inhomogeneous system of d equations in the nd unknown x_{ij} , so the r relations yield a system of rd such equations. This fact puts a premium on getting a presentation with a small number of relations. Unfortunately, no general methods seem to be known for producing such presentations.

It should be mentioned, that the ‘‘collecting process’’ producing R_j^* can be described with the help of ‘‘Fox derivatives’’; in the case of a soluble group G described by a pc presentation the data structure for the implementation is particularly neat.

Any two solutions $(f(g_1), \dots, f(g_n))$ and $(f'(g_1), \dots, f'(g_n))$ of this system of equations differ by some $(\gamma_m(g_1), \dots, \gamma_m(g_n)) \in \beta(Z^1(G, M))$ determined by a cocycle $\gamma_m \in Z^1(G, M)$. I.e., if $L = f_0 + L_{Hom}$ is the solution set of the system of equations, then L_{Hom} describes $\beta(Z^1(G, M))$. In certain cases the number of unknowns in the system of equations for Z^1 can further be reduced. If P' is a p' -subgroup of G then it will split over M , so for generators g_i of G with $g_i \in P'$ we may without loss assume $f(g_i) = 1$.

The computation of $\beta(B^1(G, M))$ follows directly from the definition: for each $\gamma_m \in B^1(G, M)$ there exists $m \in M$ such that $\gamma_m(g) = m m^{-g}$. Using additive notation and identifying M with $\mathbf{Z}_p^{1 \times d}$ via a basis of M again, and using XG_j also to denote the multiplication of the row X by the $d \times d$ matrix corresponding to g_j , we see that if $S := (I - G_1, \dots, I - G_n)$ is a $d \times dn$ matrix, then $\beta(B^1(G, M))$ is represented by the row space of S . A cocycle γ lies in $B^1(G, M)$ if and only if there exists a solution X of the inhomogeneous linear system $XS = B_\gamma$, where B_γ is the dn -row describing $\beta(\gamma) = (\gamma(g_1), \dots, \gamma(g_n))$ with respect to the chosen basis of M . Solving this system yields the d -row X describing the element $m \in M$ with $\gamma_m = \gamma$, with respect to the basis. The space of solutions of $XS = 0$ de-

scribes $C_M(K)$, where K is an arbitrary complement of M in H . (This space of solutions is of course equal to the space of common eigenvectors of G_1, \dots, G_n with eigenvalue 1).

The computation of H^1 is demonstrated by a simple example:

Let $S_4 := \langle a, b, c, d \rangle$ with $a = (12)$, $b = (123)$, $c = (13)(24)$, $d = (12)(34)$. Let $H := S_4 \times Z_2 = S_4 \times \langle e \rangle$, let $M := \langle c, d, e \rangle \triangleleft H$ and $G := \langle g_1, g_2 \mid g_1^2 = g_2^3 = (g_2 g_1)^2 = 1 \rangle$. Then $\pi : a \rightarrow g_1, b \rightarrow g_2, c \rightarrow 1, d \rightarrow 1, e \rightarrow 1$ is an epimorphism with kernel M , and τ with $g_1^\tau = ac$, $g_2^\tau = b$ fulfills $g_i^{\tau^\pi} = g_i$, $i = 1, 2$.

“Collecting” the three relations in H and writing $f_i^{g_j}$ for $(g_j^\tau)^{-1} f(g_i) g_j^\tau$, we get

$$\begin{aligned} 1 &= (g_1^\tau f(g_1))^2 &= (g_1^\tau)^2 f_1^{g_1} f_1 &= d f_1^{g_1} f_1, \\ 1 &= (g_2^\tau f(g_2))^3 &= (g_2^\tau)^3 f_2^{g_2^2} f_2^{g_2} f_2 &= f_2^{g_2^2} f_2^{g_2} f_2, \\ 1 &= (g_2^\tau f(g_2) g_1^\tau f(g_1))^2 &= (g_2^\tau g_1^\tau)^2 f_2^{g_1 g_2 g_1} f_1^{g_2 g_1} f_2^{g_1} f_1 &= c d f_2^{g_1 g_2 g_1} f_1^{g_2 g_1} f_2^{g_1} f_1. \end{aligned}$$

Introducing additive notation and basis vectors $(1\ 0\ 0) \leftrightarrow c$, $(0\ 1\ 0) \leftrightarrow d$,

$$(0\ 0\ 1) \leftrightarrow e \quad \text{we have } G_1 = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad G_2 = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

With $f(g_1) \leftrightarrow X_1 = (x_{11}, x_{12}, x_{13})$, $f(g_2) \leftrightarrow X_2 = (x_{21}, x_{22}, x_{23})$ we get

$$(0\ 0\ 0) = (0\ 1\ 0) + X_1 G_1 + X_1 = (0\ 1\ 0) + X_1 \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix},$$

$$(0\ 0\ 0) = X_2 G_2^2 + X_2 G_2 + X_2 = X_2 \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix},$$

$$\begin{aligned} (0\ 0\ 0) &= (1\ 1\ 0) + X_2 G_1 G_2 G_1 + X_1 G_2 G_1 + X_2 G_1 + X_1 \\ &= (1\ 1\ 0) + X_1 \begin{pmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} + X_2 \begin{pmatrix} 0 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}. \end{aligned}$$

So with $X = (x_{11}, x_{12}, x_{13}, x_{21}, x_{22}, x_{23})$ and $V = (0\ 1\ 0\ 0\ 0\ 0\ 1\ 1\ 0)$

$$X \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix} = V.$$

The set of solutions X is

$$(1\ 0\ 0\ 0\ 0\ 0) + \left\langle \begin{pmatrix} 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix} \right\rangle.$$

$B^1(G, M)$ is determined by the rowspace of

$$S = (G_1 - I, G_2 - I) = \begin{pmatrix} 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

So $H^1(G, M)$ is isomorphic to $(0\ 0\ 1\ 0\ 0\ 0) + B^1(G, M)$ and M has two classes of complements, represented by $\langle a, b \rangle$ and $\langle ae, b \rangle$.

4. Complements.

We are next going to describe a general method for finding the conjugacy classes of complements of an arbitrary normal subgroup M of a finite soluble group G , using a similar kind of homomorphism principle as for the classes of elements.

If M is elementary abelian, we can of course compute $H^1(G, M)$ as just described. Otherwise, using the non-commutative Gauß algorithm, it is not difficult to construct a series of normal subgroups with elementary abelian factors passing through M , that is:

$$G = N_1 > N_2 > \cdots > N_i = M > \cdots > N_m = \langle 1 \rangle$$

with $N_j \triangleleft G$ and N_j/N_{j+1} elementary abelian for $j = 1, \dots, m-1$. Starting with $j = i$ and stepping down for $j = i+1, \dots, m$, we will construct inductively the classes of complements of M/N_j in G/N_j . It suffices to describe a typical step from G/N_j to G/N_{j+1} . I.e., we may assume that

$N = N_j$ is elementary abelian and that the results are known for G/N_j . We shall need the following well-known

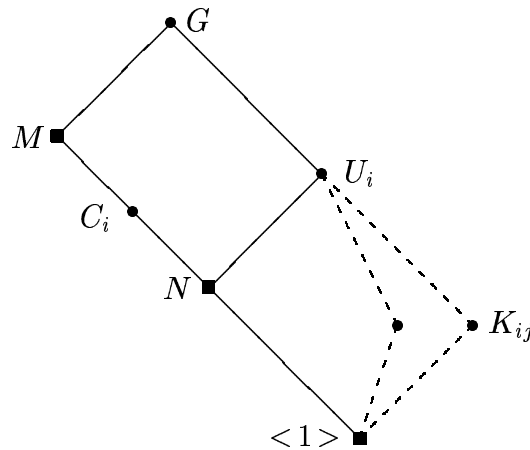
4.1 Lemma: Let G be a group, let $M \triangleleft G$ and let $K_1, K_2 < G$ be complements of M in G . Then K_1 is conjugate to K_2 in G if and only if K_1 is conjugate to K_2 under M . Furthermore $N_M(K) = C_M(K)$ for each complement K of M and $N_G(K) = K C_M(K)$.

From this we get a reduction scheme, the proof of which is immediate from the figure.

4.2 Theorem: Let G be a group with $M, N \triangleleft G$ and $M > N$.

- (i) Let $\{U_1/N, \dots, U_k/N\}$ be a system of representatives of the conjugacy classes of complements of M/N in G/N .
- (ii) For each $i = 1, \dots, k$ let $\overline{C}_i = C_{M/N}(U_i/N)$ and let C_i be its complete preimage in G , that is, $C_i/N = \overline{C}_i$.
- (iii) Let \mathcal{K}_i be the set of complements of N in U_i , (the empty set, if none exist), let $\mathcal{L}_i = \{K_{i1}, \dots, K_{is_i}\}$ be a set of representatives of the orbits of C_i acting on \mathcal{K}_i by conjugation (or again the empty set), and let $C_{ij} = C_{C_i}(K_{ij})$.

Then $\bigcup_{i=1}^k \mathcal{L}_i$ is a set of representatives of the conjugacy classes of complements of M in G , and $C_{ij} = C_M(K_{ij})$.



To apply Theorem 4.2 inductively in the above mentioned “typical step”, we have to perform the following tasks for each U_i :

- (a) We have to decide whether N has a complement in U_i . Since N is elementary abelian in our situation, one way this can be done is by investigating the solvability of an inhomogeneous system of linear equations, as explained in section 3.
- (b) Let us then assume that there is a complement K of N in U_i . Then each $K_{ij} \in \mathcal{K}_i$ can be written as

$$K_{ij} = \{ k\gamma_j(k) \mid k \in K \}$$

with a cocycle $\gamma_j \in Z^1(K, N) \cong Z^1(U_i/N, N)$. The group C_i operates on \mathcal{K}_i by conjugation. Let $c \in C_i$, then

$$c^{-1}K_{ij}c = \{ c^{-1}k\gamma_j(k)c \mid k \in K \} = \{ k \cdot [k, c] \cdot c^{-1}\gamma_j(k)c \mid k \in K \},$$

and since $c^{-1}K_{ij}c$ is also a complement of N in U_i , we see that $\alpha_c(\gamma_j)$ defined by

$$(\alpha_c(\gamma_j))(k) := [k, c]c^{-1}\gamma_j(k)c \in N$$

is also a cocycle in $Z^1(K, N)$.

Since $cN \in C_{G/N}(U_i/N)$, we see that the element $[k, c]$ of N is independent of γ_j , while $\gamma_j(k) \rightarrow c^{-1}\gamma_j(k)c$ defines a linear mapping of $Z^1(K, N)$ again considered as a \mathbf{Z}_p -vector space. Hence

$$\alpha_c : \gamma_j \rightarrow \alpha_c(\gamma_j)$$

is an affine mapping of the \mathbf{Z}_p -vector space $Z^1(K, N)$ onto itself. Now let $n \in N \triangleleft C_i$. Then

$$(\alpha_n(\gamma_j))(k) = [k, n]n^{-1}\gamma_j(k)n = [k, n]\gamma_j(k).$$

But $k \rightarrow [k, n] = n n^{-k}$ is just the coboundary γ_n , that is, $(\alpha_n(\gamma_j))(k) = \gamma_j(k) \gamma_n(k)$. Hence $N \triangleleft C_i$ acts on $Z^1(K, N)$ as the group of “translations” by the coboundaries, and the orbits of N on Z^1 are the

cosets of B^1 . Therefore C_i/N and hence C_i act on $H^1(K, N) = Z^1(K, N) / B^1(K, N)$ again as a group of affine mappings

$$\bar{\alpha}_c = \bar{\alpha}_{cN} : H^1(K, N) \rightarrow H^1(K, N)$$

with

$$\bar{\alpha}_{cN}(\gamma_j B^1(K, N)) = \alpha_c(\gamma_j) B^1(K, N).$$

- (c) It remains to find $C_{ij} := C_{C_i}(K_{ij})$ for each $K_{ij} \in \mathcal{L}_i$, the set of representatives of the C_i -orbits on \mathcal{K}_i . $C_{C_i}(K_{ij})$ can be described as the stabilizer of the cocycle γ_j belonging to K_{ij} in the affine action via α of C_i on $Z^1(K, N)$. The orbit-stabilizer algorithm applied to the action via $\bar{\alpha}$ of C_i on $H^1(K, N)$ will give the stabilizer \bar{C}_{ij} of $\gamma_j B^1(K, N) \in H^1$. In terms of the action of C_i via α on $Z^1(K, N)$, of course, \bar{C}_{ij} is the block stabilizer of the block $\gamma_j B^1(K, N)$ which was obtained as the orbit of γ_j under the action of N . Hence for each $c \in \bar{C}_{ij}$ there exists $n_c \in N$ such that

$$\alpha_c(\gamma_j) = \alpha_{n_c}(\gamma_j) = \gamma_j \gamma_{n_c}.$$

Therefore the values of the coboundary γ_{n_c} are determined as

$$\gamma_{n_c}(k) = (\gamma_j(k))^{-1} (\alpha_c(\gamma_j))(k).$$

The element n_c can be determined by a solution of an inhomogeneous system of linear equations and the space of solutions of the homogeneous system determines $C_N(K_{ij})$ as explained before the example in section 3. This allows to compute C_{ij} using the homomorphism principle described at the end of section 2 by solving such an inhomogeneous system of linear equations for each $c \in \{c_1, \dots, c_k\}$ with $\langle c_1 N, \dots, c_k N \rangle = \bar{C}_{ij}/N$.

5. Special cases and further reductions.

5.1 The central case.

The method described in section 4 reduces the use of the rather time-consuming orbit-stabilizer algorithm to the action of C_i on $H^1(K, N)$; the

rest is all done by solving systems of linear equations. So the reduction is best if $H^1(K, N)$ is small, i.e., if B^1 is close to Z^1 . On the other hand, if N is a central subgroup, then B^1 is trivial, and hence no reduction of this kind is possible. However, in this case linear methods are again possible, in a way analogous to those described in [MeN 89] (see also [LNS 84]).

Using the notation of the last section, let us again assume that there exists a complement K of N in U_i . The complete preimage of $N_{G/N}(U_i/N)$ is equal to $C_i U_i$ by lemma 4.1. Let N be central in $C_i U_i$. Since $K < C_i U_i$ centralizes N , we have $B^1(K, N) = 1$ and $H^1(K, N) = Z^1(K, N)$. For $\gamma \in Z^1(K, N)$ and $c \in C_i$

$$(\alpha_c(\gamma))(k) = [k, c]c^{-1}\gamma(k)c = [k, c]\gamma(k) \quad \text{for all } k \in K.$$

Applying this equation to the trivial cocycle γ_0 , defined by $\gamma_0(k) = 1$ for all $k \in K$, we see that each $c \in C_i$ defines a cocycle $\gamma_c : k \rightarrow [k, c]$. Because $[k, c][k, c'] = [k, cc']$ these form a subgroup S_i of $Z^1(K, N)$ and the orbits of C_i on $Z^1(K, N)$ are the cosets of this subgroup, so can easily be described. It remains to find $C_{C_i}(K_{ij})$ for each K_{ij} . Since N is centralized by C_i , all these centralizers are equal: $C_{C_i}(K_{ij}) = C_{C_i}(K)$ for all j and this subgroup can be found by the following lemma which is an exact analogue of 3.1 in [MeN 89]:

Lemma: Let $\{g_1, \dots, g_m, g_{m+1}, \dots, g_n\}$ be a PAG sequence for C_i , with $\langle g_{m+1}, \dots, g_n \rangle = N$. Then $\langle \gamma_1 = \gamma_{g_1}, \dots, \gamma_m = \gamma_{g_m} \rangle = S_i < Z^1(K, M)$. Adopting additive notation, choose a basis $\{\gamma_{i_s}, \dots, \gamma_{i_1} \mid m \geq i_s > \dots > i_1 \geq 1\}$ for S_i running from γ_m to γ_1 in this order. Then for each $k \in \{1, \dots, m\} \setminus \{i_1, \dots, i_s\}$ we can express γ_k as

$$\gamma_k = \gamma_{i_1}^{e_{k, i_1}} \cdots \gamma_{i_s}^{e_{k, i_s}},$$

and $C_{C_i}(K)$ has the PAG sequence

$$\{g_k(g_{i_1}^{e_{k, i_1}} \cdots g_{i_s}^{e_{k, i_s}})^{-1} \mid k \in \{1, \dots, m\} \setminus \{i_1, \dots, i_s\}\} \cup \{g_{m+1}, \dots, g_n\}.$$

In practical implementation, of course, a PAG sequence of G is passed through N and a CGS is used for each subgroup that occurs.

5.2 Normal complements.

If one only wants to get normal complements one is able to treat an even more general situation. Let G be a soluble normal subgroup of the group H and let $M \triangleleft H$ with $M < G$. Then we can obtain all $K \triangleleft H$ with $MK = G$ and $M \cap K = \langle 1 \rangle$ by inductive use of the following

Lemma: Let H be a group and $G, M, N \triangleleft H$ with $G > M > N$. Let N be elementary abelian and $U \triangleleft H$ with $MU = G$ and $M \cap U = N$.

Let $U/N = \langle u_1N, \dots, u_nN \mid R_j(u_1N, \dots, u_nN) = 1, j = 1, \dots, r \rangle$,

let $M = \langle m_1, \dots, m_s \rangle$, and

let $H/G = \langle h_1G, \dots, h_tG \rangle$ and $(u_iN)^{h_lN} = W_{il}(u_1N, \dots, u_nN)$ for $i = 1, \dots, n$ and $l = 1, \dots, t$.

Then each function $f : \{u_1, \dots, u_n\} \rightarrow N$, satisfying

- (i) $R_j(u_1f(u_1), \dots, u_nf(u_n)) = 1$ for $j = 1, \dots, r$,
- (ii) $[m_k, u_if(u_i)] = 1$ for $k = 1, \dots, s$ and $i = 1, \dots, n$, and
- (iii) $(u_if(u_i))^{h_l} = W_{il}(u_1f(u_1), \dots, u_nf(u_n))$ for $i = 1, \dots, n$ and $l = 1, \dots, t$

determines a subgroup $V \triangleleft H$ with $MV = G$ and $M \cap V = \langle 1 \rangle$ as $V = \langle u_1f(u_1), \dots, u_nf(u_n) \rangle$, and each $V \triangleleft H$ with $MV = G$ and $M \cap V = \langle 1 \rangle$ is determined by such an f in this way.

The proof uses the fact that V will centralize M , and the computational use of conditions (i)-(iii) for the determination of all complements V is analogous to the procedure described in section 4.

5.3 Further reductions.

In this section we give a brief sketch of some reduction methods for finding complements. Consider again the general situation described in section 4. We have $M \triangleleft G$ and want complements to M in G . Assuming that we have $N \triangleleft G$ with $N \triangleleft M$ and we have found a complement U/N to M/N in G/N , we aim to choose $N_1 \triangleleft G$ with N/N_1 elementary abelian, and

then find complements to N/N_1 in U/N_1 . For simplicity of exposition, we will not pay attention here to the management of conjugacy classes; the basic tool for that is Lemma 4.1. In the notation of section 4, everything takes place inside $C_i U_i$, so we may as well assume here that $C_i = M$, $U_i = U \triangleleft G = MU$ and $M \cap U = N > N_1 \geq \langle 1 \rangle$.

The basic idea is to reduce the problem of finding complements to N/N_1 in U/N_1 to a problem for smaller groups. If we limit our computations to the linear algebra methods of section 3, the number of linear equations from a PAG presentation of U/N (i.e., of G/M) grows quadratically with the composition length of U/N , and the number of coefficients in the equations grows as the cube of the composition length. For moderately large groups, it may be sensible or necessary to cut down the size before computing. Of course there are many groups, an easy example is the quaternion group, in which the number of classes of complements is bigger in some factor groups than it is in the whole group, and it is not always clear a priori that reduction in size is not buying too many classes in exchange. This problem is, in fact, already present for the general scheme described in section 4.

One way to reduce the size of G/N_1 is to replace N_1 by $N_2 \triangleleft G$ with $N_1 < N_2 < N$ and to consider G/N_2 first. The idea is to choose N_1 dynamically as the computation proceeds and to be willing to change the choice to N_2 in the light of experience. We will shortly give an example of this sort of replacement.

Another way to reduce G/N_1 is to replace N_1 by $L \triangleleft G$ with $N \cap L = N_1$ and $L \triangleleft U$, and then to complement NL/L in U/L . This device is particularly effective if there is a subgroup L , necessarily normal in G , that is contained in every complement to N/N_1 in U/N_1 .

Here is an illustration, with $N_1 = \langle 1 \rangle$ for convenience and with N an elementary abelian p -group. Let $C_U(N) =: D$ and let $[D, D]D^p =: E$. Then E is in every complement to N in U . If $\langle 1 \rangle < E$, then we can recur to G/E to find complements to NE/E in U/E . If $E = \langle 1 \rangle$, then D is itself a \mathbf{Z}_p -vector space on which U acts. In this case, complements to N in U intersect D in $\mathbf{Z}_p U$ -submodules W that complement N in D . One can, in effect, recur to U/W unless $W = \langle 1 \rangle$, i.e., unless $N = C_U(N)$.

Suppose that $N = C_U(N)$. Then we can perhaps use Sylow theory as well as linear algebra. If $N = U$, then of course we are done. Otherwise, suppose that $N < Q \triangleleft U$ with Q/N elementary abelian, which is more than we need, but easy to produce. If Q is also a p -group, then $1 < [N, Q]$

because $N = C_U(N)$, so we can recur to $G/[N, Q]$ to get subgroups Y that complement $N/[N, Q]$ in $U/[N, Q]$. Then replace N by $[N, Q]$ and recur to get complements to $[N, Q]$ in each Y . If, on the other hand, Q/N is a p' -group, then the complements to N in Q are the (conjugate) p -complements R of Q . By the Frattini argument, $U = N_U(R)N$, and $N \cap N_U(R) = C_N(R) = C_N(Q) \triangleleft G$. If $C_N(Q) \neq \langle 1 \rangle$, we can recur to $G/C_N(Q)$, while if $C_N(Q) = \langle 1 \rangle$, the subgroup $N_U(R)$ is a representative complement to N in U .

This account gives only a hint of some of the kinds of reductions that are possible and the tools that they require. Plainly, this sort of attack would only make sense in a setting in which the size of G/M made solving systems of linear equations unattractive or impractical. The general design of recursive algorithms for finding complements is a fairly complex question that we must leave for another paper.

6. Normalizers.

The ideas described in the previous sections can also be used to supplement a method for the calculation of the normalizer $N_G(U)$ of an arbitrary subgroup of a soluble group, which has been proposed and implemented in CAYLEY by S. Glasby and M. Slattery [GIS 90]. While their method works very efficiently in many cases, there are others in which they have to resort to explicit calculation of large orbits of subgroups under conjugation. It is in these cases that we can offer improvement. The case distinction will become clear in the sequel and examples will be given in section 7.

As with the tasks discussed in the last sections, the computation of $N_G(U)$ can be split up inductively by virtue of the following easily proved facts.

6.1 Lemma: Let H be a group, $V < H$, $N \triangleleft H$ and $N < S \leq H$ such that $\bar{S} := S/N = N_{H/N}(VN/N)$. Then $N_H(V) \leq S$ and hence $N_H(V) = N_S(V)$.

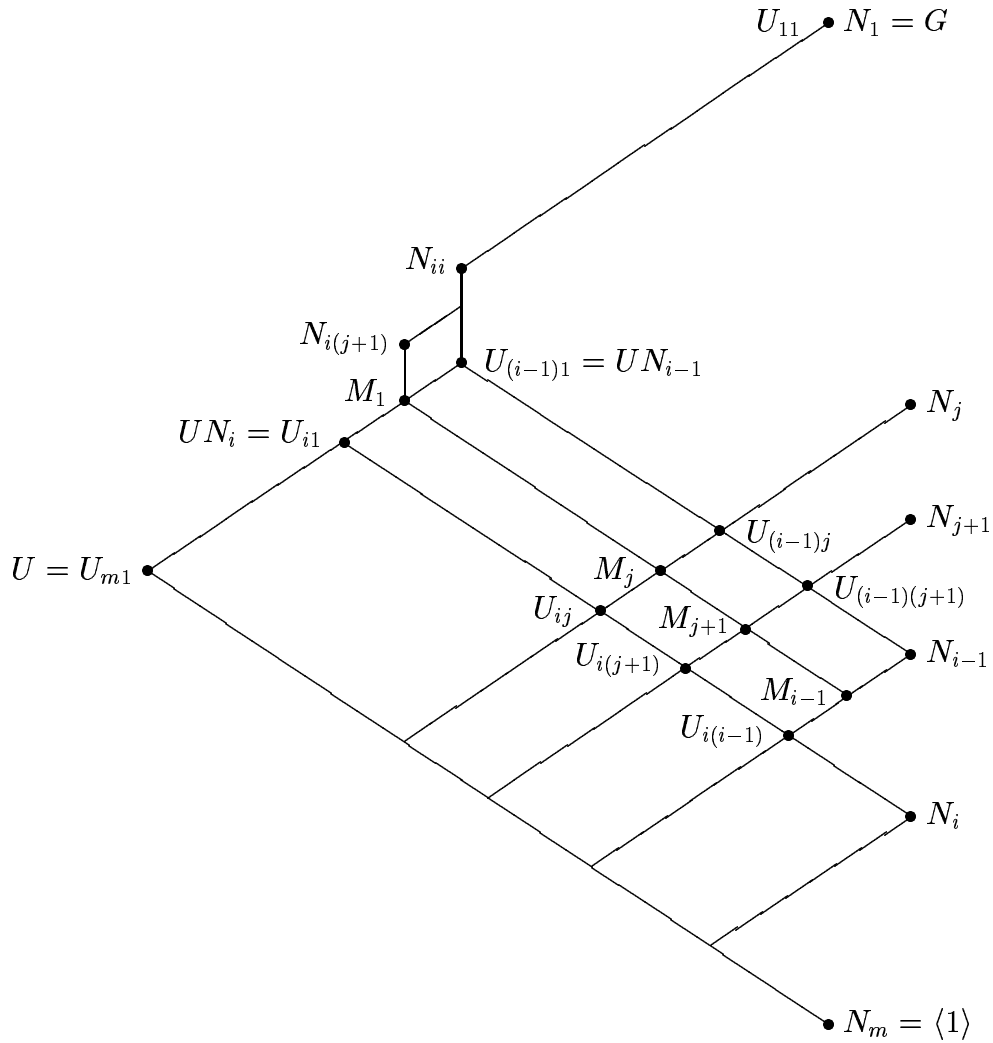
6.2 Lemma: Let H be a group, $V < H$, $M \triangleleft H$ and $T = N_H(V \cap M)$. Then $N_H(V) \leq T$ and hence $N_H(V) = N_T(V)$.

Glasby and Slattery combine these two relations in the following way: Let

$$G = N_1 \triangleright N_2 \triangleright \dots \triangleright N_m = \langle 1 \rangle$$

be a series of normal subgroups $N_i \triangleleft G$. For $1 \leq j < i \leq m$, that is, for $N_i < N_j$, let

$$U_{ij} := (UN_i \cap N_j) = (U \cap N_j)N_i \quad \text{and} \quad \bar{U}_{ij} := U_{ij}/N_i.$$



Define $N_{11} := G$, $\overline{N}_{11} := G/N_1 = \langle 1 \rangle$ and by induction for each $i = 2, \dots, m$:

(for $j = i$) let N_{ii} be the complete preimage in G
of $\overline{N}_{(i-1)1} = N_{G/N_{i-1}}(\overline{U}_{(i-1)1})$ and $\overline{N}_{ii} := N_{ii}/N_i$
for $j = i-1, \dots, 1$ let $\overline{N}_{ij} := N_{\overline{N}_{i(j+1)}}(\overline{U}_{ij})$ and N_{ij} be the complete
preimage of \overline{N}_{ij} in G .

Then $N_G(U)$ is obtained as N_{m1} . The statement is seen by twofold induction:

For fixed i by definition $\overline{N}_{i(i-1)} = N_{\overline{N}_{ii}}(\overline{U}_{i(i-1)})$, so (by lemma 6.2 with $H := \overline{N}_{ii}$) we have $\overline{N}_{ij} = N_{\overline{N}_{ii}}(\overline{U}_{ij})$ for $j = i-2, \dots, 1$, and hence in particular $\overline{N}_{i1} = N_{\overline{N}_{ii}}(\overline{U}_{i1})$. By induction on i , using lemma 6.1, this proves the claim.

Computationally there are two different kinds of steps in this induction:

1. $\overline{N}_{i(i-1)} := N_{\overline{N}_{ii}}(\overline{U}_{i(i-1)})$ has indeed to be found by an ‘‘orbit-stabilizer’’ algorithm (cf. [LNS 84], 3.1, 3.2, p. 111). However one can take advantage of the fact that $\overline{U}_{i(i-1)} = (UN_i \cap N_{i-1})/N_i \leq N_{i-1}/N_i$ which is an elementary abelian normal subgroup of $\overline{N}_{ii} = N_{ii}/N_i$, and hence can be considered as a \mathbf{Z}_p -vector space. The matrices describing the action of the generators of \overline{N}_{ii} on N_{i-1}/N_i can be computed once and for all, the non-commutative Gauß algorithm for the images of $\overline{U}_{i(i-1)}$ reduces to an ordinary one and only vector addition and matrix multiplication are needed instead of the much more time-consuming collection algorithm.
2. On the other hand, for the determination of $\overline{N}_{ij} := N_{\overline{N}_{i(j+1)}}(\overline{U}_{ij})$ with $j = i-2, \dots, 1$ we can use the methods developed in the previous sections.

Since by definition $U_{i(j+1)} \triangleleft N_{i(j+1)}$ and $U_{i(j+1)} \leq U_{ij} \leq N_{ij} \leq N_{i(j+1)}$, we have

$$\overline{N}_{ij}/\overline{U}_{i(j+1)} \cong N_{ij}/U_{i(j+1)} = N_{N_{i(j+1)}/U_{i(j+1)}}(U_{ij}/U_{i(j+1)}).$$

To determine this group we introduce $M_1 := N_{i(j+1)} \cap U_{(i-1)1}$ and $M_k := N_{i(j+1)} \cap U_{(i-1)k} = M_1 \cap U_{(i-1)k}$ for $k = 2, \dots, i-1$. As we have seen, $U_{i1} \leq N_{i1} \leq N_{i(j+1)} \leq N_{ii}$ and $U_{i1} \leq U_{(i-1)1} \triangleleft N_{ii}$. Therefore

$U_{(i-1)1}$ and further all $U_{(i-1)k} = U_{(i-1)1} \cap N_k$, $k = 2, \dots, i-1$ and all M_k , $k = 1, \dots, i-1$ are normalized by $N_{i(j+1)}$. Finally, since

$$U_{(i-1)k} = U_{ik}U_{(i-1)l}, \text{ and}$$

$$U_{il} = U_{ik} \cap U_{(i-1)l}$$

for $k = 1, \dots, i-1$ and $l = k+1, \dots, i-1$, we also get

$$M_k = U_{ik}M_l \text{ and}$$

$$U_{il} = U_{ik} \cap M_l$$

for $k = 1, \dots, i-1$ and $l = k+1, \dots, i-1$. From this we see that if for some j we have $U_{ij} = M_j \triangleleft N_{i(j+1)}$ then for all $k < j$ also $U_{ik} = M_k$, in particular $U_{i1} = M_1 \triangleleft N_{i(j+1)}$ and hence $N_{i1} = N_{i(j+1)}$, so that we can immediately proceed from G/N_i to G/N_{i+1} .

Otherwise we have

$$U_{ij}M_{j+1} = M_j \triangleleft N_{i(j+1)} \text{ and}$$

$$U_{ij} \cap M_{j+1} = U_{i(j+1)} \triangleleft N_{i(j+1)} .$$

That is, $M_j/U_{i(j+1)}$ is a semidirect product of the elementary abelian normal subgroup $M_{j+1}/U_{i(j+1)}$ by $U_{ij}/U_{i(j+1)}$ and since M_j and M_{j+1} are normal in $N_{i(j+1)}$ we have that $N_{N_{i(j+1)}/U_{i(j+1)}}(U_{ij}/U_{i(j+1)})$ is the stabilizer of $U_{ij}/U_{i(j+1)}$ in the operation of $N_{i(j+1)}/U_{i(j+1)}$ on the set of complements. The orbits of M_{j+1} , which are also orbits of M_j , are blocks in this operation, and $N_{i(j+1)}/U_{i(j+1)}$, or equivalently $N_{i(j+1)}/M_j$, operates on these blocks, which can be described by the elements of $H^1(U_{ij}/U_{i(j+1)}, M_{j+1}/U_{i(j+1)})$.

We have to find the stabilizer of the block containing U_{ij} , that is the stabilizer of the element $B^1(U_{ij}/U_{i(j+1)}, M_{j+1}/U_{i(j+1)}) \in H^1$ by the orbit-stabilizer method, but we can then get from this the desired $N_{ij}/U_{i(j+1)}$ as the stabilizer of the trivial cocycle in the same way as described at the end of section 4(c).

The group $M_{j+1}/U_{i(j+1)}$ is isomorphic to a factor of N_{i-1}/N_i which is an elementary abelian p -group, and $U_{ij}/U_{i(j+1)}$ is isomorphic to a subgroup of N_j/N_{j+1} which is an elementary abelian q -group.

If $p \neq q$, $H^1(U_{ij}/U_{i(j+1)}, M_{j+1}/U_{i(j+1)}) = 1$, so there is only one block of complements and $N_{i(j+1)}/U_{i(j+1)}$ is its stabilizer, no orbit-stabilizer algorithm is necessary, and only the task to solve the inhomogeneous sets of linear equations remains in order to determine $N_{ij}/U_{i(j+1)}$. For each inhomogeneous set of equations one solution can be written down directly as described in [GIS 90].

7. Performance.

Comparison with other implementations is only possible for the normalizer routine for soluble groups in CAYLEY described in [GIS 90]. As is to be expected, a gain is reached if orbit-stabilizer calculations can be replaced by “linear” methods in the “non-coprime” situation, as in the following examples suggested by M. Slattery.

Let p and q be primes, $n \in \mathbf{N}$ and $p \mid n \mid q - 1$. Let Z_p , Z_q and $Z_n = \langle a \rangle$ be cyclic, $Z_q \rtimes Z_n$ be a nonabelian semidirect product represented as a permutation group of degree q and $G = Z_p \text{wr}_q (Z_q \rtimes Z_n)$. We list for some values of p , q and n and subgroups $U < \langle a \rangle$ the order of $N_G(U)$ and as indication of the length of the longest orbit occurring in an orbit-stabilizer method the order of some B^1 , and we give computing times for the determination of $N_G(U)$: t_1 for the GAP-implementation on a MASSCOMP 5500 PEP, t_2 for the CAYLEY implementation on an APOLLO 10000, which is 5-10 times faster.

p	q	n	$ G $	U	$ N(U) $	$ B^1 $	t_1	t_2
2	31	30	$2^{32} \cdot 3 \cdot 5 \cdot 31$	$\langle a^{15} \rangle$	$2^{17} \cdot 3 \cdot 5$	2^{15}	25 sec	103 sec
3	13	12	$3^{14} \cdot 2^2 \cdot 15$	$\langle a^4 \rangle$	$2^2 \cdot 3^6$	3^8	10 sec	10 sec
5	11	10	$5^{12} \cdot 2 \cdot 11$	$\langle a^2 \rangle$	5^4	5^8	6 sec	> 400 sec

We have further compared implementations, both in GAP, of the Glasby-Slattery method of the “conjugating element” to which we referred at the end of section 6, i.e. without orbit-stabilizer computation, with our method of solving inhomogeneous linear equations instead in the case of a certain group of order $5^6 \cdot 313^5$ and have obtained computing times of 104 sec and 20 sec respectively, indicating that with elements of higher order the transition to linear methods pays, even in comparison to highly efficient methods involving collection.

8. Concluding remarks

The only parts of the methods described in this paper that have not been implemented yet are the further reductions described in section 5.3. They are actually only part of a more general recursive scheme proposed by the third author in [Wri 88a,b].

It should further be remarked that not only is there a formal analogy between our methods and those of [MeN 89], the latter can in fact be interpreted as using H^0 instead of H^1 . The analogy raises the question of the possibility of using similar methods with H^2 for the classification of soluble groups; Zassenhaus' space groups algorithm should provide a model.

Bibliography

- [Can 84] Cannon, J., *A Computational Toolkit for Finite Permutation Groups*. M. Aschbacher et al., ed., Proc. Rutgers Group Theory Year 1983/84, Cambridge U.P. 1984, 1–18.
- [Con 90] Conlon, S.B., *Calculating characters of p -groups*. J. Symbolic Comp. to appear.
- [FeN 79] Felsch, V., Neubüser, J., *An algorithm for the computation of conjugacy classes and centralizers in p -groups*. Lecture Notes in Computer Science 72 (1979) 452–465.
- [GIS 90] Glasby, S.P., Slattery, M.C., *Computing intersections and normalizers in soluble groups*. J. Symbolic Comp. to appear.
- [HaN 76] Havas, G., Nicholson, T., *Collection*. R.D. Jenks, ed., SYMSAC'76, Assoc. Comp. Mach., New York (1976) 9–14.
- [Hol 84] Holt, D.F., *The calculation of the Schur multiplier of a permutation group*. M.D. Atkinson, ed., Computational Group Theory, Acad. Press (1984) 307–319.
- [L-GS 90] Leedham-Green, C.R., Soicher, L.H., *Collection from the Left and other Strategies*. J. Symbolic Comp. to appear.
- [LNS 84] Laue, R., Neubüser, J., Schoenwaelder, U., *Algorithms for finite soluble groups and the SOGOS system*. M.D. Atkinson, ed., Computational Group Theory, Acad. Press (1984) 105–135.

- [Mac 74] Macdonald, I. D., *A computer application to finite p -groups*. J. Austral. Math. Soc. 17 (1974) 102–112.
- [MeN 89] Mecky, M., Neubüser, J., *Some remarks on the computation of conjugacy classes of soluble groups*. Bull. Austral. Math. Soc. 40 (1989) 281–292.
- [Neu 61] Neubüser, J., *Bestimmung der Untergruppenverbände endlicher p -Gruppen auf einer programmgesteuerten elektronischen Dualmaschine*. Numer. Math. 3 (1961) 271–278.
- [New 76] Newman, M.F., *Calculating presentations for certain kinds of quotient groups*. R.D. Jenks, ed., SYMSAC'76, Assoc. Comput. Mach., New York (1976) 2–8.
- [NNS 88] Niemeyer, A., Nickel, W., Schönert, M., *GAP. Getting started and Reference Manual*. Lehrstuhl D für Mathematik, RWTH Aachen, 1988.
- [O'Br 90] O'Brien, E.A., *The p -group generation algorithm*. J. Symbolic Comp. to appear.
- [Ple 87] Plesken, W., *Towards a soluble quotient algorithm*. J. Symbolic Comp. 4 (1987) 111–122.
- [Sim 87] Sims, C.C., *Verifying Nilpotence*. J. Symbolic Comp. 3 (1987) 231–247.
- [Sim 90a] Sims, C.C., *Implementing the Baumslag-Cannonito-Miller Polycyclic Quotient Algorithm*. J. Symbolic Comp. to appear.
- [Sim 90b] Sims, C.C., *Computing the Order of a Solvable Permutation Group*. J. Symbolic Comp. to appear.
- [Va-L 90] Vaughan-Lee, M.R., *Collection from the Left*. J. Symbolic Comp. to appear.
- [Wri 88a] Wright, C.R.B., *A Pseudo-Cayley complementation procedure for soluble groups*. Manuscript 1988, University of Oregon.
- [Wri 88b] Wright, C.R.B., *Recursive Algorithms for computing complements in Finite Groups*. Manuscript 1988, University of Oregon.