

# ALGORITHMIC USE OF THE MAL'CEV CORRESPONDENCE

BJÖRN ASSMANN<sup>1</sup>

Centre for Interdisciplinary Research in Computational Algebra (CIRCA),  
University of St. Andrews, North Haugh, St. Andrews, KY16 9SS Fife, Scotland

## Abstract

Mal'cev showed in the 1950s that there is a correspondence between radicable torsion-free nilpotent groups and rational nilpotent Lie algebras. In this paper we show how to establish the connection between the radicable hull of a finitely generated torsion-free nilpotent group and its corresponding Lie algebra algorithmically. We apply it to fast multiplication of elements of polycyclically presented groups.

## 1 Introduction

The connection between groups and Lie rings, respectively Lie algebras, is a well-known and mathematically very useful concept. For example, a typical way to solve a problem in a Lie group is to transfer the problem to the Lie algebra of the group, study it there with the help of tools from linear algebra and transfer the result back into the Lie group.

Mechanisms of this kind have also been shown to be useful for algorithmic applications. For instance, Vaughan-Lee and O'Brien used Lie ring techniques to construct a consistent polycyclic presentation of  $R(2, 7)$ , the largest 2-generator group of exponent 7 [15].

In this paper we demonstrate the algorithmic usefulness of the Mal'cev correspondence between torsion-free radicable nilpotent groups and rational nilpotent Lie algebras. We use it to develop an algorithm for fast multiplication in polycyclic groups. To be more precise, for a infinite polycyclic group  $G$ , given by a polycyclic presentation, we show how to determine a subgroup  $H$  of finite index in  $G$ , and construct an algorithm which carries out fast multiplication in  $H$ .

For the setup of this algorithm we determine a finitely generated torsion-free nilpotent subgroup  $N$  of  $H$  and compute a matrix representation of  $N$ . Using the latter we calculate a Lie algebra  $\mathcal{L}$  which corresponds to the radicable hull of  $N$ . Multiplication in  $H$  is first reduced to some calculations with automorphisms of  $N$ , which correspond to automorphisms of  $\mathcal{L}$ , which, being linear, are much easier to handle.

An alternative approach would be to compute a matrix representation for the whole group  $G$ . That this is possible for any polycyclic group was shown by Auslander [1]. Lo and Ostheimer constructed an algorithm for this task [11]. Such a representation could then be applied to carry out multiplications in  $G$ . However it seems to be much harder to compute a representation for  $G$  than for  $N$ . Also,

---

<sup>1</sup>I gratefully acknowledge support from the "Gottlieb Daimler und Karl Benz-Stiftung" and the EPSRC.

the author is not aware of a practical implementation of the algorithm of Lo and Ostheimer.

## 2 Mal'cev correspondence

A group  $G$  is called *radicable* if for all  $g \in G$  and for all  $m \in \mathbb{N}$  there exists an  $h \in G$  such that  $h^m = g$ . In this section we recall some well known facts about the connection between radicable torsion-free nilpotent groups and nilpotent Lie algebras, the Mal'cev correspondence, discovered by Mal'cev in 1951 [12].

We denote by  $\mathrm{Tr}_1(n, \mathbb{Q})$  the group of all upper unitriangular matrices over  $\mathbb{Q}$  and by  $\mathrm{Tr}_0(n, \mathbb{Q})$  the set of all upper triangular rational matrices with zeros on the diagonal. Although this can be done in a more general way, we will restrict ourself in this section to the description of the connection between subgroups of  $\mathrm{Tr}_1(n, \mathbb{Q})$  and Lie subalgebras of  $\mathrm{Tr}_0(n, \mathbb{Q})$ . For proofs of the stated results and more background see [16, Chapter 6].

Let  $g \in \mathrm{Tr}_1(n, \mathbb{Q})$ . The *logarithm* of  $g = 1 + u$  is

$$\log g = u - \frac{1}{2}u^2 + \cdots + \frac{(-1)^n}{(n-1)}u^{n-1}$$

and for  $x \in \mathrm{Tr}_0(n, \mathbb{Q})$ , the *exponential* of  $x$  is

$$\exp x = 1 + x + \frac{1}{2}x^2 + \cdots + \frac{1}{(n-1)!}x^{n-1}.$$

Note that this coincides with the definition of  $\log$  and  $\exp$  on the complex numbers, since  $u^n = x^n = 0$ . These mappings are mutually inverse bijections and moreover, for commuting matrices  $x, y \in \mathrm{Tr}_0(n, \mathbb{Q})$ , we have  $(\exp x)(\exp y) = \exp(x + y)$ .

The vector space  $\mathrm{Tr}_0(n, \mathbb{Q})$  with the Lie bracket  $[x, y] = xy - yx$  has the structure of a Lie algebra. For longer Lie brackets we use the left norm convention, i.e.  $[x_1, \dots, x_r] = [[x_1, \dots, x_{r-1}], x_r]$ , and for a vector of positive integers  $e = (e_1, \dots, e_r)$  we define

$$[x, y]_e = [x, \underbrace{y, \dots, y}_{e_1}, \underbrace{x, \dots, x}_{e_2}, \dots].$$

Since  $[x_1, \dots, x_n] = 0$  for all  $x_i \in \mathrm{Tr}_0(n, \mathbb{Q})$ , the latter is a nilpotent Lie algebra of class  $n - 1$ .

**Theorem 2.1 (Baker-Campbell-Hausdorff formula)** *There exist universal constants  $q_e \in \mathbb{Q}$  not depending on  $n$  such that for all  $x, y \in \mathrm{Tr}_0(n, \mathbb{Q})$  and  $z(x, y) = \log((\exp x)(\exp y))$  we have*

$$z(x, y) = x + y + \sum_e q_e [x, y]_e,$$

where we take the sum over all vectors  $e = (e_1, \dots, e_r)$ , with positive integer entries, such that  $e_1 + \dots + e_r < n - 1$ . In particular this means that  $z(x, y)$  is an element of the Lie subalgebra of  $\mathrm{Tr}_0(n, \mathbb{Q})$  generated by  $x$  and  $y$ .

A similar formula holds for  $\log([\exp(x), \exp(x)]) = [x, y] + \sum_e p_e [x, y]_e$ , where the  $p_e \in \mathbb{Q}$  are again universal constants. Using the BCH-formula we can define a group multiplication  $x * y = z(x, y)$  on  $\mathrm{Tr}_0(n, \mathbb{Q})$ . The exponential map is then an isomorphism of groups between  $(\mathrm{Tr}_0(n, \mathbb{Q}), *)$  and  $\mathrm{Tr}_1(n, \mathbb{Q})$ . The following theorem explains the interplay of subgroups of  $\mathrm{Tr}_1(n, \mathbb{Q})$  and Lie subalgebras of  $\mathrm{Tr}_0(n, \mathbb{Q})$  via this mechanism.

**Theorem 2.2** *Let  $G \leq \mathrm{Tr}_1(n, \mathbb{Q})$  and let  $\mathcal{L}(G) = \mathbb{Q} \log G$  be the  $\mathbb{Q}$ -vector space spanned by  $\log G = \{\log g | g \in G\}$ . Let  $L$  be a Lie subalgebra of  $\mathrm{Tr}_0(n, \mathbb{Q})$ . Then the following holds:*

- $\exp L$  is a radicable torsion-free nilpotent subgroup of  $\mathrm{Tr}_1(n, \mathbb{Q})$ .
- $\mathcal{L}(G)$  is a Lie subalgebra of  $\mathrm{Tr}_0(n, \mathbb{Q})$ .
- $G \leq \exp \mathcal{L}(G)$  and every element of  $\exp \mathcal{L}(G)$  has some positive power lying in  $G$ .

A group  $N$  is called a  $\mathcal{T}$ -group if it is finitely generated torsion-free nilpotent. It is well-known that every  $\mathcal{T}$ -group can be embedded in  $\mathrm{Tr}_1(n, \mathbb{Q})$  for some  $n \in \mathbb{N}$  [16, Chapter 5]. Therefore, given a faithful representation  $\beta : N \rightarrow \mathrm{Tr}_1(n, \mathbb{Q})$ , we can construct the Lie algebra  $\mathcal{L}(N\beta)$  of  $N\beta$ . In the following we will denote by  $\mathcal{L}(N)$  the Lie algebra  $\mathcal{L}(N\beta)$  and for  $g \in N$  by  $\mathrm{Log}(g)$  the element  $\log(g\beta)$ , i.e. we identify the groups  $N$  and  $N\beta$ . This is justified by the fact that for two embeddings  $\beta_1, \beta_2$  the Lie algebras  $\mathcal{L}(N\beta_1)$  and  $\mathcal{L}(N\beta_2)$  are isomorphic, see [16, Chapter 6].

A group  $\hat{N}$  is said to be a *radicable hull* of a  $\mathcal{T}$ -group  $N$ , if it is a radicable torsion-free nilpotent group, that contains  $N$  and has the property that every element in  $\hat{N}$  has some positive power lying in  $N$ . It can be shown that  $\hat{N}$  is unique up to isomorphism [16, Chapter 6]. Theorem 2.2 shows that  $\exp \mathcal{L}(G)$  is a radicable hull of a finitely generated subgroup  $G \leq \mathrm{Tr}_1(n, \mathbb{Q})$ . Using a faithful representation  $\beta : N \rightarrow \mathrm{Tr}_1(n, \mathbb{Q})$  and identifying  $N$  and  $N\beta$  we can construct the radicable hull of any  $\mathcal{T}$ -group  $N$ .

**Theorem 2.3** *Let  $G \leq \mathrm{Tr}_1(n, \mathbb{Q})$  and let  $\Gamma$  be the group of automorphisms of the Lie algebra  $\mathcal{L}(G)$  which map  $\log G$  onto itself. Then the function  $\phi : \mathrm{Aut}(G) \rightarrow \Gamma$ ,  $\alpha \mapsto \exp \circ \alpha \circ \log$  is an isomorphism.*

$$\begin{array}{ccc}
 G & \xrightarrow{\alpha} & G \\
 \exp \uparrow & & \downarrow \log \\
 \mathcal{L}(G) & \xrightarrow{\quad} & \mathcal{L}(G)
 \end{array}$$

### 3 Polycyclic presentations

Let  $G$  be a polycyclic group. A *polycyclic sequence* of  $G$  is a sequence of elements  $\mathcal{G} = (g_1, \dots, g_n)$  of  $G$  such that the subgroups  $G_i = \langle g_i, \dots, g_n \rangle$  form a subnormal series  $G = G_1 > \dots > G_n > G_{n+1} = \{1\}$  with non-trivial cyclic factors.

Let  $r_i = [G_i : G_{i+1}]$  and  $I = \{i \in \{1, \dots, n\} \mid r_i < \infty\}$ . Then each element  $g \in G$  has a unique normal form with respect to the polycyclic sequence:

$\text{nf}(g) = g_1^{e_1} \cdots g_n^{e_n}$  with  $e_i \in \mathbb{Z}$  and  $0 \leq e_i < r_i$  for  $i \in I$ . Thus  $g$  can be represented by the exponent vector  $(e_1, \dots, e_n)$  with respect to  $\mathcal{G}$ .

Each polycyclic sequence  $(g_1, \dots, g_n)$  of  $G$  defines a presentation of  $G$  on the generators  $g_1, \dots, g_n$  with relations of the form

$$\begin{aligned} g_i^{g_j} &= g_{j+1}^{e(i,j,j+1)} \cdots g_n^{e(i,j,n)} & \text{for } 1 \leq j < i \leq n, \\ g_i^{g_j^{-1}} &= g_{j+1}^{f(i,j,j+1)} \cdots g_n^{f(i,j,n)} & \text{for } 1 \leq j < i \leq n, \\ g_i^{r_i} &= g_{i+1}^{\ell(i,i+1)} \cdots g_n^{\ell(i,n)} & \text{for } i \in I, \end{aligned}$$

where the right hand sides in these relations are the normal forms of the elements on the left hand sides. It is well-known that these relations define  $G$ . Such a presentation is called a *consistent polycyclic presentation* for  $G$ . The term consistent refers to the fact that every element in this finitely presented group on the abstract generators  $g_1, \dots, g_n$  has a unique normal form  $g_1^{e_1} \cdots g_n^{e_n}$  with  $e_i \in \mathbb{Z}$  and  $0 \leq e_i < r_i$  for  $i \in I$ . Throughout this paper all polycyclic presentations are consistent.

Let  $H$  be a  $\mathcal{T}$ -group, i.e. a finitely generated torsion-free nilpotent group. A polycyclic sequence  $(h_1, \dots, h_n)$  is called a *Mal'cev basis* for  $H$  if the subgroups  $\langle h_i, \dots, h_n \rangle$  form a central series with infinite cyclic factors. Since the upper central series of a  $\mathcal{T}$ -group has torsion-free factors, every  $\mathcal{T}$ -group has a Mal'cev basis.

#### 4 Algorithmic realization of the Mal'cev correspondence

The aim of this section is to show how to realize algorithmically the correspondence between elements of a  $\mathcal{T}$ -group  $N$ , represented by their exponent vectors with respect to a certain polycyclic sequence, and their counterparts in the Lie algebra  $\mathcal{L}(N)$ , represented as coefficient vectors with respect to a certain basis.

**Lemma 4.1** *Let  $(g_1, \dots, g_l)$  be a Mal'cev basis for a  $\mathcal{T}$ -group  $N$ . Then  $\mathcal{B} = \{\text{Log}(g_1), \dots, \text{Log}(g_l)\}$  is a basis for the Lie algebra  $\mathcal{L}(N)$ . In particular, the dimension of  $\mathcal{L}(N)$  is equal to the Hirsch length of  $N$ .*

**Proof** Let  $g = g_1^{a_1} \cdots g_l^{a_l} \in N$ . Then  $\text{Log}(g) = a_1 \text{Log}(g_1) * \cdots * a_l \text{Log}(g_l)$ . Thus it is sufficient to show that  $\mathcal{B}$  is basis for the Lie algebra  $L$  generated by  $\mathcal{B}$ . We will show this via induction over  $l$ , the Hirsch length of  $N$ .

If  $N = \langle g_1 \rangle$  then  $\{\text{Log}(g_1)\}$  is a basis for  $\mathcal{L}(N)$ . Assume that the lemma is true for all  $\mathcal{T}$ -groups of Hirsch length  $l-1$ . First we show that  $\mathcal{B}$  is a generating set for the  $\mathbb{Q}$ -vector space  $\mathcal{L}(N)$ . By assumption the vector spaces  $\langle \text{Log}(g_2), \dots, \text{Log}(g_l) \rangle_{\mathbb{Q}}$  and  $\langle \text{Log}(g_1), \text{Log}(g_3), \dots, \text{Log}(g_l) \rangle_{\mathbb{Q}}$  are closed under taking Lie brackets. Thus we have to show that  $[\text{Log}(g_1), \text{Log}(g_2)] \in \langle \mathcal{B} \rangle_{\mathbb{Q}}$ . By Corollary 3 of [16, Chapter 6] we have that  $[\text{Log}(g_1), \text{Log}(g_2)] = \text{Log}([g_1, g_2]) + \sum_i \alpha_i \text{Log}(\chi_i(g_1, g_2))$ , where  $\alpha_i \in \mathbb{Q}$  and  $\chi_i$  is a repeated group theoretic commutator in  $g_1, g_2$  of length  $\geq 3$ . Since  $[g_1, g_2], \chi_i(g_1, g_2) \in \langle g_3, \dots, g_l \rangle$  the right hand side of the last equation is contained in  $\mathcal{L}(\langle g_3, \dots, g_l \rangle)$  and thus in the  $\mathbb{Q}$ -vector space spanned by  $\mathcal{B}$ .

It remains to show that the elements of  $\mathcal{B}$  are linearly independent. So assume that  $\text{Log}(g_1) \in \langle \text{Log}(g_2), \dots, \text{Log}(g_l) \rangle_{\mathbb{Q}} = L_2$ . Therefore  $g_1 \in \exp(L_2)$  (recall that we identify  $N$  and  $N\beta \leq \text{Tr}_1(n, \mathbb{Q})$ ) and, according to Theorem 2.2, there must be an  $m \in \mathbb{N}$  such that  $g_1^m \in \langle g_2, \dots, g_l \rangle$ . Since  $(g_1, \dots, g_l)$  is a Mal'cev basis this is a contradiction.  $\square$

Lo/Ostheimer [11] and de Graaf/Nickel [2] describe practical algorithms to compute a faithful representation of a  $\mathcal{T}$ -group in  $\text{Tr}_1(n, \mathbb{Q})$ . The latter has been implemented in GAP [18] and is part of the Polycyclic package [5]. We use this implementation for the computation of representations of  $\mathcal{T}$ -groups.

A polycyclic sequence  $\mathcal{M} = (M_1, \dots, M_l)$  for a group  $H \leq \text{Tr}_1(n, \mathbb{Q})$  is called *constructive* if there exists a practical algorithm, which, given any  $h \in H$ , determines the normal form  $\text{nf}(h)$  with respect to  $\mathcal{M}$ . It is well-known how to compute a constructive polycyclic sequence for a given finitely generated group  $H \leq \text{Tr}_1(n, \mathbb{Q})$  which is also a Mal'cev basis of  $H$ , see for example [17, Chapter 9].

We now summarize the algorithms which set up the Mal'cev correspondence between a  $\mathcal{T}$ -group  $N$  and the Lie algebra  $\mathcal{L}(N)$ .

#### **SetupMalcevCorrespondence( $N$ )**

- 1: compute a faithful representation  $\beta : N \rightarrow \text{Tr}_1(n, \mathbb{Q})$ .
- 2: compute a constructive pc-sequence  $\mathcal{M} = (M_1, \dots, M_l)$  for  $N\beta$ .
- 3: determine the basis  $\mathcal{B} = \{\log M_1, \dots, \log M_l\}$  of  $\mathcal{L}(N)$ .
- 4: compute a polycyclic presentation for  $N$  with respect to  $\beta^{-1}(\mathcal{M})$ .

Given  $g \in N$ , the following algorithm computes the corresponding element in  $\mathcal{L}(N)$ . We represent elements in  $N$  by their normal form with respect to the polycyclic sequence  $\beta^{-1}(\mathcal{M}) = (n_1, \dots, n_l)$  of  $N$ , and the elements of  $\mathcal{L}(N)$  as coordinate vectors with respect to the canonical basis  $\log(\mathcal{M})$  of Lemma 4.1.

#### **Logarithm( $g$ )**

- 1: let  $(e_1, \dots, e_l)$  be the exponent vector of  $g$  with respect to  $\beta^{-1}(\mathcal{M})$ .
- 2: determine  $\beta(g) = M_1^{e_1} \cdots M_l^{e_l}$ .
- 3: determine  $\log \beta(g)$ .
- 4: determine  $\gamma = (\gamma_1, \dots, \gamma_l)$  such that  $\log \beta(g) = \sum \gamma_i \log M_i$ .
- 5: return  $\gamma$ .

Finally, given an element in  $\log(N\beta)$ , represented as coordinate vector with respect to  $\log(\mathcal{M})$ , the following algorithm computes its counterpart in  $N$ .

#### **Exponential( $(\gamma_1, \dots, \gamma_l)$ )**

- 1: determine  $N = \sum \gamma_i \log M_i$ .
- 2: determine  $\exp N$ .
- 3: compute  $(e_1, \dots, e_l)$  such that  $M_1^{e_1} \cdots M_l^{e_l} = \exp N$ .
- 4: return  $n_1^{e_1} \cdots n_l^{e_l}$ .

## 5 Collection in polycyclic groups

Let  $G$  be an infinite polycyclically presented group. In this section we show that the Mal'cev correspondence can be used for a efficient multiplication algorithm in a certain subgroup  $H$  of finite index in  $G$ .

A polycyclic presentation (pcp) is a good computer representation of a polycyclic group. Algorithms for finite solvable groups given by a pcp were developed in the 1980s by Laue, Neubüser and Schoenwaelder [8]. More recently, calculations with infinite polycyclically presented groups have been shown to be practical, see for example [4, 17].

The efficiency of calculations with polycyclic presentations depends on the ability to compute quickly the normal form of the product of two elements given in normal form. The original methods for doing this were based on rewriting using pcp and called *collection*. Nowadays, collection is used as a general term for an algorithm which computes the normal form of an element of a polycyclically presented group. In the recent past, various strategies for collection algorithms have been discussed [6, 9, 19]. “Collection from the left” is the current state of the art.

The collection process in  $\mathcal{T}$ -groups is much better understood than in the more general class of polycyclic groups. The following theorem states that collection with respect to a Mal'cev basis can be done symbolically. For a proof see [7].

**Theorem 5.1 (Hall)** *Let  $G$  be a  $\mathcal{T}$ -group with Mal'cev basis  $(g_1, \dots, g_l)$ . Define the functions  $\zeta_1, \dots, \zeta_l$  in  $2l$  integer variables  $x_1, \dots, x_l, y_1, \dots, y_l$  such that*

$$g_1^{x_1} \cdots g_l^{x_l} g_1^{y_1} \cdots g_l^{y_l} = g_1^{\zeta_1} \cdots g_l^{\zeta_l}.$$

*Then the functions  $\zeta_i$  are rational polynomials.*

This result can be used for computational applications. In the 90s Leedham-Green and Soicher developed the algorithm “Deep Thought” [10], which computes these polynomials and uses them for collection in  $\mathcal{T}$ -groups. An implementation of “Deep Thought” by Merkwitz [13] is part of the GAP system.

Having said that collection in  $\mathcal{T}$ -groups is easier than in the general case, it is interesting to note that  $\mathcal{T}$ -groups can be regarded as main ingredients of polycyclic groups. Namely, if  $G$  is a polycyclic group, then it has a normal series  $G \geq K \geq N \geq 1$ , where  $N$  is a  $\mathcal{T}$ -group,  $K/N$  is free-abelian and  $G/K$  is finite. Further the next theorem shows that, up to a finite index, the group  $G$  is made out of two  $\mathcal{T}$ -groups. A proof can be found in [16, Chapter 3].

**Theorem 5.2 (Newell)** *Let  $K$  be a polycyclic group and  $N$  a  $\mathcal{T}$ -group which is normal in  $K$  and has the property that  $K/N$  is nilpotent. Then there exists a  $\mathcal{T}$ -group  $C$  such that  $CN$  is of finite index in  $K$ .*

The group  $C$  from Theorem 5.2 is said to be a *nilpotent almost-supplement* for  $N$  in  $K$ . ‘Almost’ because  $C$  and  $N$  generate a subgroup of finite index, and ‘supplement’ because  $C$  may intersect  $N$  non-trivially. Since  $[G : K] < \infty$  the group  $C$  is also a nilpotent almost-supplement for  $N$  in  $G$ . This structure of

polycyclic groups can also be explored algorithmically. In [4, Chapter 9] Eick describes a practical algorithm to compute a nilpotent-by-abelian-by-finite series  $G \geq K \geq N \geq 1$  and methods to determine a nilpotent almost-supplement  $C$  for  $N$  in  $G$ .

Therefore, up to a finite index, collection in infinite polycyclic groups can be reduced to the case of a polycyclic group  $H$  generated by two  $\mathcal{T}$ -groups  $C$  and  $N$  such that  $N \triangleleft H$  and  $H/N$  is free-abelian. We explain how collection in  $H$  can be realized efficiently. Let  $\mathcal{N} = (n_1, \dots, n_l)$  be a Mal'cev basis of  $N$  and let  $(c_1N, \dots, c_kN)$  be a basis for the free abelian group  $CN/N$ . Then  $\mathcal{H} = (c_1, \dots, c_k, n_1, \dots, n_l)$  is a polycyclic sequence for  $H$ .

**Lemma 5.3** *The list  $(c_1, \dots, c_k)$  can be extended to a Mal'cev basis*

$$\mathcal{C} = (c_1, \dots, c_k, c_{k+1}, \dots, c_{k+m})$$

*of the  $\mathcal{T}$ -group  $C$ .*

**Proof** The upper central series of  $C \cap N$  has torsion-free factors and is invariant under the action of  $(c_1, \dots, c_k)$ . This series can be refined to a central series with torsion-free factors which are centralized by  $(c_1, \dots, c_k)$ . To see this, let  $M$  be one of the torsion-free factors. Denote by  $A$  the centralizer of  $(c_1, \dots, c_k)$  in  $M$ . Then  $A$  is non-trivial, since  $C$  acts nilpotently on  $M$ , and furthermore  $M/A$  is torsion-free, since for  $m \in M, z \in \mathbb{N}$ , the equality  $zm = (zm)^{c_i} = z(m^{c_i})$  implies  $m^{c_i} = m$ . Thus, by induction on the dimension of  $M$ , we get a strictly ascending series of submodules of  $M$  with torsion-free factors, which are by construction centralized by  $(c_1, \dots, c_k)$ .  $\square$

Denote by  $c^x n^{\bar{x}}$  the element in  $H$  given by the exponent vector  $(x_1, \dots, x_k, \bar{x}_1, \dots, \bar{x}_l)$  with respect to  $\mathcal{H}$ . For two elements  $c^x n^{\bar{x}}, c^y n^{\bar{y}} \in H$  we have

$$c^x n^{\bar{x}} c^y n^{\bar{y}} = c^x c^y (n^{\bar{x}})^{(c^y)} n^{\bar{y}}.$$

Since  $CN/N$  is free abelian, the normal form of  $c^x c^y$  with respect to  $\mathcal{C}$  is of the form  $c^{x+y} c_{k+1}^{z_{k+1}} \dots c_{k+m}^{z_{k+m}}$ . The computation of the tail  $t = c_{k+1}^{z_{k+1}} \dots c_{k+m}^{z_{k+m}}$  is easy in the sense that we can apply standard methods for  $\mathcal{T}$ -groups, such as Deep Thought. We can also compute the normal form of the tail  $t \in C \cap N$  with respect to  $\mathcal{N}$ . The efficient computation of the normal form of

$$(n^{\bar{x}})^{(c^y)} = (n^{\bar{x}})^{(c_1^{y_1} \dots c_k^{y_k})}$$

is the crucial step of our method. For this, we will describe an algorithm which applies powers of automorphisms using the Mal'cev correspondence. Finally, the multiplication of  $\text{nf}(t)$ ,  $\text{nf}((n^{\bar{x}})^{(c^y)})$  and  $n^{\bar{y}}$  in  $N$  can be done by standard techniques.

It remains to describe an algorithm for the computation of the normal form of

$$n^{(\varphi^q)},$$

where  $n \in N$ ,  $\varphi \in \text{Aut}(N)$  and  $q \in \mathbb{Z}$ . As described in Section 4, let  $\beta : N \rightarrow \text{Tr}_1(m, \mathbb{Q})$  be a faithful representation of  $N$  and  $\mathcal{M} = (M_1, \dots, M_l)$  be a constructive polycyclic sequence for  $N\beta$ , which is also a Mal'cev basis for  $N\beta$ . Denote by  $n_i$  the element  $\beta^{-1}(M_i) \in N$ . We define a  $l \times l$  matrix  $\Phi$  by

$$\text{Log}(n_i^\varphi) = \sum_{j=1}^l \Phi_{ij} \text{Log}(n_j).$$

Then by Theorem 2.3 the matrix  $\Phi$  is a representation of the Lie algebra isomorphism  $\exp \circ \varphi \circ \log$ , with respect to the basis  $\{\log M_1, \dots, \log M_l\}$ . This yields the following algorithm.

**ApplyPowerOfAutomorphism**(  $n, \varphi, q$  )

- 1: determine  $\gamma = \mathbf{Logarithm}(n)$ .
- 2: compute  $\bar{\gamma} = \gamma \cdot \Phi^q$ .
- 3: compute  $g = \mathbf{Exponential}(\bar{\gamma})$ .
- 4: return  $g$ .

If we want to apply several powers of automorphisms, as in the computation of the normal form of  $(n^{\bar{x}})^{(c^y)}$ , we switch only once from  $n^{\bar{x}}$  to the corresponding element  $\gamma$  in the Lie algebra, then multiply  $\gamma$  with  $\Phi(c_1)^{y_1} \dots \Phi(c_k)^{y_k}$ , where  $\Phi(c_i)$  is the matrix representation of the Lie algebra isomorphism corresponding to the conjugation with  $c_i$ , and then switch back to the representation with respect to  $(n_1, \dots, n_l)$ .

Given that this algorithm works for arbitrary automorphisms, it can also be used as a partial step for collection in arbitrary polycyclic groups. If  $G$  is a polycyclic group,  $N$  a  $\mathcal{T}$ -group which is normal in  $G$  and  $C$  a nilpotent almost-supplement for  $N$  in  $G$ , then  $G$  has a polycyclic sequence of the form

$$\mathcal{G} = (g_1, \dots, g_j, c_1, \dots, c_k, n_1, \dots, n_l),$$

where  $\langle g_1 CN, \dots, g_j CN \rangle = G/CN$  and  $\mathcal{H} = (c_1, \dots, c_k, n_1, \dots, n_l)$  is as before. The action of  $g^z = g_1^{z_1} \dots g_j^{z_j}$  on an element  $n^{\bar{x}} \in N$  can be computed in the same way as the action of  $c^y$  on  $n^{\bar{x}}$ .

The development of this algorithm was motivated by a recent result of du Sautoy. In [3] he proves that symbolic collection in so-called *splittable* polycyclic groups is possible. For definition and background of a semi-simple splitting of a polycyclic group see [16, Chapter 7]. Every polycyclic group has a splittable subgroup of finite index, which can be constructed as a subgroup of finite index of the group  $CN$  mentioned in the last paragraph. However, for practical collection algorithms it seems to be better to avoid this concept. In our approach we avoid having to pass to a subgroup of finite index of  $CN$ , and we also avoid computations with elements of finite extension of  $\mathbb{Q}$ , which are used by du Sautoy to describe symbolic collection in splittable groups.

Group	Hirsch length	HL( $N$ )	Class $N$	Dimension $N\beta$	Time
$G(\mathrm{Tr}_3(\mathcal{O}_1))$	9	6	2	6	00:00.229
$G(\mathrm{Tr}_4(\mathcal{O}_1))$	16	12	3	16	00:03.752
$G(\mathrm{Tr}_5(\mathcal{O}_1))$	25	20	4	46	05:46.230
$G(\mathrm{Tr}_3(\mathcal{O}_2))$	12	9	2	9	00:00.669
$G(\mathrm{Tr}_4(\mathcal{O}_2))$	22	18	3	27	00:32.664
$G(\langle \bar{\varphi}_1 \rangle \rtimes F_{2,4})$	9	8	4	10	00:00.801
$G(\langle \bar{\varphi}_1 \rangle \rtimes F_{2,5})$	15	14	5	20	00:34.531
$G(\langle \bar{\varphi}_2 \rangle \rtimes F_{3,4})$	33	32	4	43	12:22.355

Table 1. Setup of the Mal'cev correspondence: The third column displays the Hirsch Length of the  $\mathcal{T}$ -group  $N$  for which the Lie algebra  $\mathcal{L}(N)$  is computed. In the fifth column the dimension  $m$  of the vector space on which  $N\beta \leq \mathrm{Tr}_1(m, \mathbb{Q})$  acts is indicated. The sixth column contains the time which is needed by the algorithm `SetupMalcevCorrespondence( $N$ )` of Section 4.

## 6 Runtimes

In this section we compare the performance of the collection algorithm presented in Section 5 with the classical “Collection from the left” method, as implemented in the GAP package `Polycyclic` [5] Version 1.1. We also specify the time required to set up the Mal'cev correspondence.

We construct our first class of examples of polycyclically presented groups with the help of matrices over algebraic integers. Let  $\mathbb{Q}(\theta)$  be an algebraic extension of  $\mathbb{Q}$  and  $\mathcal{O}$  its maximal order. We denote by  $\mathrm{Tr}_n(\mathcal{O})$  the group of upper-triangular matrices in  $GL_n(\mathcal{O})$ , by  $\mathrm{Tr}_1(n, \mathcal{O})$  the subgroup of matrices in  $\mathrm{Tr}_n(\mathcal{O})$  with 1s on the diagonal and by  $D_n(\mathcal{O})$  the group of diagonal matrices in  $GL_n(\mathcal{O})$ . Note that every polycyclic group has a subgroup of finite index which can be embedded in some  $\mathrm{Tr}_n(\mathcal{O})$  [16, page 132].

Let  $U(\mathcal{O})$  be the group of units of the maximal order  $\mathcal{O}$ . By Dirichlet's Units Theorem,  $U(\mathcal{O})$  is polycyclic and therefore also  $D_n(\mathcal{O})$ . Using the torsion unit and fundamental units of  $U(\mathcal{O})$ , it is straightforward to obtain a polycyclic presentation for  $D_n(\mathcal{O})$ . In a similar way to  $\mathrm{Tr}_1(n, \mathbb{Q})$ , we can compute a constructive polycyclic sequence for  $\mathrm{Tr}_1(n, \mathcal{O})$ , which then yields a polycyclic presentation for  $\mathrm{Tr}_1(n, \mathcal{O})$ . Using the fact that  $\mathrm{Tr}_n(\mathcal{O}) = D_n(\mathcal{O}) \rtimes \mathrm{Tr}_1(n, \mathcal{O})$ , we obtain a polycyclically presented group  $G(\mathrm{Tr}_n(\mathcal{O}))$  which is isomorphic to  $\mathrm{Tr}_n(\mathcal{O})$ .

We use the irreducible polynomials  $p_1(x) = x^2 - 3$  and  $p_2(x) = x^3 - x^2 + 4$  for our examples. By  $\mathcal{O}_i$  we denote the maximal order of  $\mathbb{Q}(\theta_i)$  where  $\theta_i$  is a zero of the polynomial  $p_i$ .

The second class of examples is constructed as follows: Let  $F_n$  be the free group on  $n$  generators  $f_1, \dots, f_n$ . Then  $F_{n,c} = F_n / \gamma_{c+1}(F_n)$  is the free nilpotent group on  $n$  generators of class  $c$ . It is a  $\mathcal{T}$ -group and we use the nilpotent quotient algorithm in the GAP package `NQ` [14] to compute a polycyclic presentation for it. An automorphism  $\varphi$  of  $F_n$  naturally induces an automorphism  $\bar{\varphi}$  of  $F_{n,c}$ .

Group	range = 2		range = 4		range = 8	
	Cftl	Malcev	Cftl	Malcev	Cftl	Malcev
$G(\text{Tr}_3(\mathcal{O}_1))$	0.001	0.002	0.951	0.003	*	0.004
$G(\text{Tr}_4(\mathcal{O}_1))$	0.011	0.013	22:24.238	0.017	*	0.025
$G(\text{Tr}_5(\mathcal{O}_1))$	0.035	0.148	*	0.174	*	0.223
$G(\text{Tr}_3(\mathcal{O}_2))$	0.002	0.004	00:31.017	0.006	*	0.009
$G(\text{Tr}_4(\mathcal{O}_2))$	5.808	0.049	*	0.061	*	0.088
$G(\langle \bar{\varphi}_1 \rangle \rtimes F_{2,4})$	0.001	0.004	0.001	0.005	7.099	0.008
$G(\langle \bar{\varphi}_1 \rangle \rtimes F_{2,5})$	0.001	0.020	0.221	0.027	*	0.043
$G(\langle \bar{\varphi}_2 \rangle \rtimes F_{3,4})$	0.001	0.179	0.531	0.221	*	0.304

Table 2. Runtimes for the multiplication of two random elements: This table specifies the average runtime of 100 computations of the normal form of  $gh$  where  $g, h$  are randomly chosen group elements of range  $r$ . The two compared methods are Cftl, i.e. ‘Collection from the left’ as implemented in the GAP package Polycyclic [5] Version 1.1, and Malcev, i.e. the algorithm explained in Section 5. The symbol \* means that the average runtime is more than 1 hour.

We use the automorphism  $\varphi_1$  of  $F_2$  which maps  $f_1$  to  $f_2^{-1}$  and  $f_2$  to  $f_1 f_2^3$  and the automorphism  $\varphi_2$  of  $F_3$  mapping  $f_1$  to  $f_2^{-1}$ ,  $f_2$  to  $f_3^{-1}$  and  $f_3$  to  $f_2^{-3} f_1^{-1}$  for our examples.

An automorphism  $\psi$  of  $F_{n,c}$  can be used to construct a polycyclically presented group  $G(\langle \psi \rangle \rtimes F_{n,c})$  which is isomorphic to  $\langle \psi \rangle \rtimes F_{n,c}$ .

The collection algorithm of Section 5 can be applied to these examples by setting  $N = \text{Tr}_1(n, \mathcal{O})$  (respectively  $N = F_{n,c}$ ) and  $C = D_n(\mathcal{O})$  (respectively  $C = \langle \psi \rangle$ ). All calculation in  $N$  were carried out via the matrix representation  $\beta : N \rightarrow \text{Tr}_1(m, \mathbb{Q})$ .

In Table 1 we give an overview of the example groups and specify the time required to set up the Mal’cev correspondence between the normal nilpotent  $\mathcal{T}$ -group  $N$  and the Lie algebra  $\mathcal{L}(N)$ , as described in Section 4. Most of the time for this set up is used to construct a faithful matrix representation for  $N$ .

In Table 2 we indicate the average runtime for the multiplication of two random elements. We generate random elements of a group  $G$  with polycyclic sequence  $(g_1, \dots, g_k)$  in the following way. Let  $r$  be a natural number. A random element  $g \in G$  of range  $r$  is of the form  $g = g_1^{e_1} \cdots g_k^{e_k}$ , where  $e_i$  is a randomly chosen integer in  $[-r, \dots, r]$ .

All computations were carried out in GAP Version 4.4.5 on a Pentium 4 machine with 3.2 gigahertz and 1 gigabyte of memory. The display format for runtimes is minutes:seconds.milliseconds.

### Discussion of the results:

The runtimes displayed in Table 2 show that Collection from the left (Cftl) is faster than the collection algorithm of Section 5 (Malcev) only for the multiplication of group elements with very small exponents, like integers between -2 and 2. In these cases little rewriting has to be done and so Cftl is quicker.

For slightly bigger exponents Malcev outperforms Cftl. For example the multi-

plication of two random elements of range 5 in the group  $G(\text{Tr}_5(\mathcal{O}_1))$  using Cftl took at least 1 hour on average. In fact, we found frequently examples of random elements  $g, h \in G(\text{Tr}_5(\mathcal{O}_1))$  of range 5, whose multiplication using Cftl did not terminate after 24 hours. The average runtime for the same kind of computation using Malcev was 189 milliseconds.

The reason why Cftl fails in these situations is that the exponents arising in the normal form of the product become too big. For instance, the size of the exponents of  $\text{nf}(gh)$  for two random elements  $g, h \in G(\text{Tr}_5(\mathcal{O}_1))$  of range 5 can easily be bigger than  $10^{10}$ . Malcev can handle these cases, since the algorithm is based on arithmetic with the exponents rather than classical rewriting.

It is interesting to note that the size of the exponents of  $\text{nf}(gh)$  varies a lot for random elements  $g, h$  of same range. As a consequence the runtime of Cftl for several multiplications differs considerably. For example, the average runtime for 100 computations of  $\text{nf}(gh)$ , where  $g, h \in G(\text{Tr}_4(\mathcal{O}_1))$  were random elements of range 3, was 4.296 seconds with a standard deviation of 29.901 seconds. Malcev showed a stable behavior in this respect in all experiments. In the cited example the average runtime was 0.014 seconds with a standard deviation of 0.009 seconds.

Moreover, Malcev is able to deal with multiplications of random elements of much higher range. For example in the group  $G(\text{Tr}_3(\mathcal{O}_2))$  it is possible to multiply two random elements of range 1000 on average in 466 milliseconds. It seems that the average runtime of Malcev as function in the range of the input elements is linear, see Figure 1.

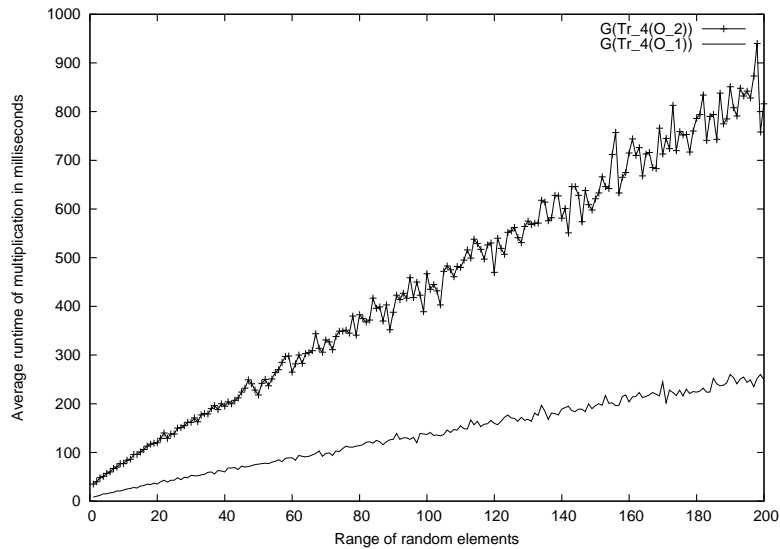


Figure 1. For the groups  $G(\text{Tr}_4(\mathcal{O}_1))$  and  $G(\text{Tr}_4(\mathcal{O}_2))$  the average runtime of the collection algorithm Malcev as a function in the range of the multiplied random elements is displayed. The graph suggests a linear relationship. For comparison, Collection from the left is not able to carry out multiplications of random elements of range 5 within 1 hour in either group.

The results make it clear that it is worthwhile to set up the Mal'cev correspondence and to use the collection algorithm Malcev, if at least one non-trivial multiplication is going to be carried out. However for more complicated groups it can be very hard to set up this correspondence. The most time consuming step is the calculation of the matrix representation  $\beta : N \rightarrow \text{Tr}_1(m, \mathbb{Q})$ . For example, we were not able to compute a matrix representation of  $F_{2,7}$  in less than 1 hour. Future work will explore alternatives to this calculation.

### Acknowledgment

I would like to thank Steve Linton for various inspiring discussions and Colva Roney-Dougal for helpful comments on an earlier version of this article.

### References

- [1] L. Auslander. On a problem of Philip Hall. *Ann. of Math. (2)*, 86:112 – 116, 1967.
- [2] W. de Graaf and W. Nickel. Constructing faithful representations of finitely-generated torsion-free nilpotent groups. *J. Symb. Comput.*, 33:31–41, 2002.
- [3] M. du Sautoy. Polycyclic groups, analytic groups and algebraic groups. *Proc. London Math. Soc. (3)*, 85:62–92, 2002.
- [4] B. Eick. Algorithms for polycyclic groups. Habilitationsschrift, Universität Kassel, 2001.
- [5] B. Eick and W. Nickel. *Polycyclic - computing with polycyclic groups*, 2000. A refereed GAP 4 package, see [18].
- [6] V. Gebhardt. Efficient collection in infinite polycyclic groups. *J. Symbolic Comput.*, 34 (3):213–228, 2002.
- [7] P. Hall. Nilpotent groups. In *The collected works of Philipp Hall*, pages 415 – 462. Clarendon Press, Oxford, 1988. Notes of lectures given at the Canadian Mathematical Congress 1957 Summer Seminar.
- [8] R. Laue, J. Neubüser, and U. Schoenwaelder. Algorithms for finite soluble groups and the SOGOS system. In *Computational Group Theory*, pages 105 – 135, London, New York, 1984. (Durham, 1982), Academic Press.
- [9] C. R. Leedham-Green and L. H. Soicher. Collection from the left and other strategies. *J. Symbolic Comput.*, 9:665 – 675, 1990.
- [10] C. R. Leedham-Green and L. H. Soicher. Symbolic collection using Deep Thought. *LMS J. Comput. Math.*, 1:9 – 24, 1998.
- [11] E. H. Lo and G. Ostheimer. A practical algorithm for finding matrix representations for polycyclic groups. *J. Symbol. Comput.*, 28:339 – 360, 1999.
- [12] A. J. Mal'cev. On certain classes of infinite soluble groups. *Mat. Sb.*, 28:567 – 588, 1951.
- [13] W. Merkwitz. Symbolische Multiplikation in nilpotenten Gruppen mit Deep Thought. Diplomarbeit, RWTH Aachen, 1997.
- [14] W. Nickel. *NQ*, 1998. A refereed GAP 4 package, see [18].
- [15] E. O'Brien and M. Vaughan-Lee. The 2-generator restricted burnside group of exponent 7. *Internat. J. Algebra Comput.*, 12:575–592, 2002.
- [16] D. Segal. *Polycyclic Groups*. Cambridge University Press, Cambridge, 1983.
- [17] C. C. Sims. *Computation with finitely presented groups*. Cambridge University Press, Cambridge, 1994.
- [18] The GAP Group. *GAP – Groups, Algorithms and Programming*. <http://www.gap-system.org>, 2005.
- [19] M. Vaughan-Lee. Collection from the left. *J. Symbolic Comput.*, 9:725–733, 1990.