

# Constructing faithful representations of finitely-generated torsion-free nilpotent groups

WILLEM A. DE GRAAF<sup>1</sup> AND WERNER NICKEL<sup>2</sup>

<sup>1</sup> *Mathematical Institute, University of Utrecht, PO BOX 80.010, 3508 TA Utrecht, The Netherlands*

<sup>2</sup> *Fachbereich Mathematik, TU Darmstadt, Schloßgartenstraße 7, 64289 Darmstadt, Germany*

## Abstract

We formulate an algorithm for calculating a representation by unipotent matrices over the integers of a finitely-generated torsion-free nilpotent group given by a polycyclic presentation. The algorithm works along a polycyclic series of the group, each step extending a representation of an element of that series to the next element.

## 1. Introduction

A finitely-generated, nilpotent and torsion-free group is called a  $\mathcal{T}$ -group. An example of a  $\mathcal{T}$ -group is the subgroup  $U_r(\mathbb{Z})$  of the general linear group  $\mathrm{GL}(r, \mathbb{Z})$  consisting of all upper-triangular matrices with all diagonal entries 1. By a theorem of S. A. Jennings (see [5]) every  $\mathcal{T}$ -group can be embedded into  $U_r(\mathbb{Z})$ . In this paper we consider the algorithmic problem of constructing such an embedding from a polycyclic presentation of a  $\mathcal{T}$ -Group.

Lo and Ostheimer ([8]) published a first algorithm for this purpose. For a  $\mathcal{T}$ -group  $G$ , they define a right ideal  $I$  of the group ring  $\mathbb{Z}G$  such that  $\mathbb{Z}G/I$  is a finite-dimensional free  $\mathbb{Z}$ -module, and a faithful  $G$ -module. In order to construct a basis of this quotient the algorithm relies on the calculation of a Gröbner basis for  $I$ .

The main idea of our algorithm is borrowed from a proof of Ado's theorem for Lie algebras, as can for instance be found in [1], see also [4]. We work our way up a polycyclic series of  $G$ . The basic step consists of an algorithm for extending a unipotent matrix representation of a  $\mathcal{T}$ -group  $N$  to a unipotent matrix representation of a  $\mathcal{T}$ -group  $H$ , where  $N$  is a normal subgroup of  $H$ , and  $H/N \cong \mathbb{Z}$ . We let  $H$  act on the dual space  $(\mathbb{Z}N)^*$  and construct a finite-dimensional faithful submodule.

As a byproduct this also leads to an essentially new proof of the theoretical result that every  $\mathcal{T}$ -group can be embedded into  $U_r(\mathbb{Z})$ . Our algorithm uses only simple concepts from linear algebra. As a consequence, its implementation is less involved than the Lo-Ostheimer approach. Moreover, experiments suggest that the algorithm presented here produces matrix representations of smaller dimension.

This paper is organized as follows. Section 2 recalls properties of  $\mathcal{T}$ -groups needed later. In Section 3 we give the theoretical foundation of our algorithm. In Section 4 we formulate the algorithm and illustrate it with an example. Finally, in Section 5 we discuss the implementation of the algorithm in the computer algebra system GAP 4 ([3]), and the running times of the program on some sample inputs are given.

## 2. Preliminaries

A  $\mathcal{T}$ -group is polycyclic. In the following we will recall basic facts about polycyclic groups, specialised for the context of  $\mathcal{T}$ -groups. Sims [10, Chapter 9] gives a general introduction into the concepts mentioned here.

Let  $G$  be a  $\mathcal{T}$ -group. Then  $G$  is a poly- $C_\infty$  group and there is a series of normal subgroups

$$G = G_1 \supset G_2 \supset \cdots \supset G_{m+1} = \{1\}$$

of  $G$  such that  $G_i/G_{i+1}$  is infinite cyclic and central for  $1 \leq i \leq m$  (see [5]). Now let  $u_i \in G_i$  be such that  $u_i G_{i+1}$  generates  $G_i/G_{i+1}$ . By induction, it follows that each  $g \in G$  can be written as a unique *normal word*  $g = u_1^{e_1} \cdots u_m^{e_m}$  with  $e_i \in \mathbb{Z}$ . We call the sequence  $(u_1, \dots, u_m)$  a *poly- $C_\infty$  generating sequence* for  $G$ . Since the normal series is central,  $[u_j, u_i] \in G_{j+1}$  and  $[u_j, u_i]$  is a normal word  $w_{ij}$  in  $u_{j+1}, \dots, u_m$ . Relative to a poly- $C_\infty$  generating sequence  $(u_1, \dots, u_m)$  the group  $G$  has a presentation of the form

$$G = \langle u_1, \dots, u_m \mid u_j u_i = u_i u_j w_{ij} \text{ for } 1 \leq i < j \leq m \rangle \quad (1)$$

A presentation of the form (1) is called a polycyclic presentation of  $G$ . Using the polycyclic presentation any word in  $(u_1, \dots, u_m)$  can be rewritten as a normal word. There are algorithms for rewriting a group element to a word in normal form, called collection algorithms (see [10], [6], [7], [11]). If every element  $g \in G$  has a *unique* normal form, then the polycyclic presentation (1) is called consistent. In the sequel we assume that every  $\mathcal{T}$ -group is given by a consistent polycyclic presentation of the form (1).

If a  $\mathcal{T}$ -group is given by a consistent polycyclic presentation, then the multiplication of two elements in normal form can be performed by collecting the concatenation of the two words to normal forms. Clearly, the exponents in the normal form of the product are a function of the exponents in the factors.

**Theorem 1 (P. Hall)** *Let  $G$  be a  $\mathcal{T}$ -group and  $u_1, \dots, u_m$  a poly- $C_\infty$  generating sequence for  $G$ . Let  $g = u_1^{\alpha_1} \dots u_m^{\alpha_m}$  and  $h = u_1^{\beta_1} \dots u_m^{\beta_m}$  be elements of  $G$  and denote the sequence of exponents of  $g$  and  $h$  by  $\alpha$  and  $\beta$ , respectively.*

*Then there are polynomials  $f_i \in \mathbb{Q}[x_1, \dots, x_m, y_1, \dots, y_m]$  such that*

$$gh = u_1^{f_1(\alpha, \beta)} \dots u_m^{f_m(\alpha, \beta)}.$$

*Furthermore, each  $f_i$  has the form*

$$f_i(x, y) = x_i + y_i + p_i(x_1, \dots, x_{i-1}, y_1, \dots, y_{i-1}).$$

*Proof:* The first part is Theorem 6.5 of Hall [5]. The second part follows from the fact that  $G_i/G_{i+1}$  is central and from the observation, that the exponent of  $u_i$  in the product can be computed in the factor group  $G/G_{i+1}$ . Therefore, only the exponents of  $u_1, \dots, u_i$  contribute to the exponent of  $u_i$  in the product.  $\square$

**Lemma 2** *Let  $G$  be a  $\mathcal{T}$ -group with poly- $C_\infty$  generating sequence  $u_1, \dots, u_m$  and  $\rho : G \rightarrow U_r(\mathbb{Z})$  a representation of  $G$ .*

*Then there are polynomials  $p_{ij} \in \mathbb{Q}[x_1, \dots, x_m]$  such that the  $(i, j)$ -th entry of the matrix*

$$\rho(u_1^{\alpha_1} \dots u_m^{\alpha_m})$$

*is equal to  $p_{ij}(\alpha_1, \dots, \alpha_m)$ .*

*Proof:* We have that  $\rho(u_i) = 1 + M_i$  where  $M_i$  is strictly upper triangular. Therefore,  $M_i^r = 0$  and

$$\rho(u_i^{\alpha_i}) = (1 + M_i)^{\alpha_i} = \sum_{k=0}^{\infty} \binom{\alpha_i}{k} M_i^k = \sum_{k=0}^{\alpha_i-1} \binom{\alpha_i}{k} M_i^k.$$

Now the binomial coefficient  $\binom{\alpha_i}{k}$  is the polynomial

$$\frac{\alpha_i(\alpha_i - 1) \dots (\alpha_i - k + 1)}{k!}$$

in  $\alpha_i$ . Since the number of terms in the sum is independent of  $\alpha_i$ , each entry in the matrix  $\rho(u_i^{\alpha_i})$  is a polynomial in  $\alpha_i$ . It follows that the entries of the matrix  $\rho(u_1^{\alpha_1} \dots u_m^{\alpha_m})$  are polynomials in  $\alpha_1, \dots, \alpha_m$ .  $\square$

### 3. Constructing extensions

Now let  $G$  be a  $\mathcal{T}$ -group with polycyclic generating sequence  $u_1, \dots, u_m$ . We want to construct an injective homomorphism  $\rho : G \rightarrow U_r(\mathbb{Z})$  for some  $r > 0$ . The method for constructing such a homomorphism described in this paper starts with a representation  $\rho_m$  of the subgroup generated by  $u_m$ . For example we can set

$$\rho_m(u_m) = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}.$$

The main part of the algorithm consists of a method for constructing a representation  $\rho_k$  for the group generated by  $u_k, \dots, u_m$  from a representation  $\rho_{k+1}$  of the group generated by  $u_{k+1}, \dots, u_m$ . Starting with  $\rho_m$  as given above and iterating this process  $m - 1$  times gives a representation for  $G$  of the required type.

Suppose that  $H$  is a  $\mathcal{T}$ -group,  $N$  a normal subgroup of  $H$  such that  $H/N$  is an infinite cyclic group and  $\rho : N \rightarrow U_r(\mathbb{Z})$  a representation of  $N$ . Let  $h_0$  be an element of  $H$  such that  $\langle h_0N \rangle = H/N$  and let  $n_1, \dots, n_t$  be a poly- $C_\infty$  generating sequence of  $N$ .

The group algebra of  $N$  over the integers is denoted by  $\mathbb{Z}N$  and

$$(\mathbb{Z}N)^* = \{f : \mathbb{Z}N \rightarrow \mathbb{Z} \mid f \text{ is linear}\}$$

is its dual space. Let  $N$  act on  $\mathbb{Z}N$  via multiplication from the right and define  $a^{h_0} = h_0^{-1}ah_0$  for  $a \in \mathbb{Z}N$ . It is elementary to check that this definition can be extended to a right action of  $H$  on  $\mathbb{Z}N$  (compare Segal [9, Proposition 5.1]). This induces an action from the left of  $H$  on  $(\mathbb{Z}N)^*$  as follows

$$(h \cdot f)(a) = f(ah) \text{ for } f \in (\mathbb{Z}N)^*, h \in H, a \in \mathbb{Z}N.$$

For  $1 \leq i, j \leq r$  define  $c_{ij} \in (\mathbb{Z}N)^*$  to be the function that maps each  $a \in \mathbb{Z}N$  to the  $(i, j)$ -th entry of  $\rho(a)$ . The linear function  $c_{ij}$  is called a *coefficient* of the representation. By Lemma 2, there is a polynomial  $p_{ij}(x_1, \dots, x_t)$  such that  $c_{ij}(n_1^{\alpha_1} \dots n_t^{\alpha_t}) = p_{ij}(\alpha_1, \dots, \alpha_t)$ . Let  $C_\rho$  be the  $\mathbb{Z}$ -module generated by  $\{c_{ij} \mid 1 \leq i, j \leq r\}$ . We call  $C_\rho$  the *coefficient space* of  $\rho$ .

**Lemma 3** *The coefficient space  $C_\rho$  is an  $\mathbb{Z}N$ -module. If  $\rho$  is faithful, then  $C_\rho$  is faithful.*

*Proof:* For  $n \in N$  the equation

$$(n \cdot c_{ij})(a) = c_{ij}(an) = \sum_{k=1}^r c_{ik}(a)c_{kj}(n) \text{ for all } a \in \mathbb{Z}N$$

shows that  $n \cdot c_{ij} = \sum_{k=1}^r c_{kj}(n)c_{ik} \in C_\rho$  for  $1 \leq i, j \leq r$  and that  $C_\rho$  is an  $\mathbb{Z}N$ -module.

For  $n \in N$  suppose that  $n \cdot c_{ij} = c_{ij}$  for all  $1 \leq i, j \leq r$ . Evaluating  $c_{ij}$  at the identity element of  $N$  gives

$$c_{ij}(1) = (n \cdot c_{ij})(1) = c_{ij}(n) \text{ for } 1 \leq i, j \leq r$$

and  $\rho(n) = \rho(1)$ . If  $\rho$  is faithful, this implies  $n = 1$  and shows that  $C_\rho$  is a faithful  $N$ -module.  $\square$

Now let  $S_\rho$  be the  $\mathbb{Z}H$ -submodule of  $(\mathbb{Z}N)^*$  generated by  $C_\rho$ . As a  $\mathbb{Z}$ -module,  $S_\rho$  is generated by the set  $\{h_0^k c_{ij} \mid 1 \leq i, j \leq r; k \in \mathbb{Z}\}$ . This follows directly from the fact that  $N$  is normal in  $H$  and that  $C_\rho$  is generated by  $\{c_{ij} \mid 1 \leq i, j \leq r\}$  as a  $\mathbb{Z}$ -module.

**Lemma 4** *Let  $f \in S_\rho$ . Then there is  $p_f \in \mathbb{Q}[x_1, \dots, x_t]$  such that*

$$f(n_1^{\alpha_1} \cdots n_t^{\alpha_t}) = p_f(\alpha_1, \dots, \alpha_t).$$

*Proof:* For  $f \in C_\rho$  this is clear by the remarks made earlier. By Theorem 1 there are polynomials  $q_1, \dots, q_t \in \mathbb{Q}[x_1, \dots, x_t]$  such that for  $k \in \mathbb{Z}$

$$n_1^{\alpha_1} \cdots n_t^{\alpha_t} h_0^k = h_0^k n_1^{\alpha_1} n_2^{\alpha_2 + q_2(\alpha_1)} \cdots n_t^{\alpha_t + q_t(\alpha_1, \dots, \alpha_{t-1})}$$

and each  $q_i$  is a polynomial in  $x_1, \dots, x_{i-1}$  only. Setting  $\beta_i = q_i(\alpha_1, \dots, \alpha_{i-1})$  we get

$$\begin{aligned} (h_0^k \cdot c_{ij})(n_1^{\alpha_1} \cdots n_t^{\alpha_t}) &= c_{ij}(h_0^{-k} n_1^{\alpha_1} \cdots n_t^{\alpha_t} h_0^k) = c_{ij}(n_1^{\alpha_1 + \beta_1} \cdots n_t^{\alpha_t + \beta_t}) \\ &= p_{ij}(\alpha_1 + \beta_1, \dots, \alpha_t + \beta_t). \end{aligned}$$

Hence the polynomial corresponding to  $h_0^k c_{ij}$  is  $p_{ij}(x_1 + q_1, \dots, x_t + q_t)$ .  $\square$

**Theorem 5** *As a  $\mathbb{Z}$ -module,  $S_\rho$  is finite-dimensional. Furthermore, there exists a basis of  $S_\rho$  such that the corresponding matrix representation maps each  $h \in H$  to an upper triangular matrix with all diagonal entries 1.*

*Proof:* Let  $R$  be the polynomial ring  $\mathbb{Q}[x_1, \dots, x_t]$  and let  $<$  be the reverse lexicographic order on the monomials of  $R$  defined as follows:  $x_1^{k_1} \cdots x_t^{k_t} < x_1^{l_1} \cdots x_t^{l_t}$  if there is an index  $1 \leq i \leq t$  such that  $k_t = l_t, \dots, k_{i+1} = l_{i+1}$  and  $k_i < l_i$ . Define the *leading monomial*  $\text{lm}(p)$  of an element  $p \in R$  to be the largest monomial with respect to  $<$  that occurs in  $p$  with non-zero coefficient. Then  $<$  induces a partial order on  $R$  defined by  $p < q$  if  $\text{lm}(p) < \text{lm}(q)$ . It is routine to show that  $<$  satisfies the descending chain condition and is translation invariant, i.e. that  $p < q$  implies  $sp < sq$  for any  $s \in R$ .

By the previous lemma, the partial order on  $R$  induces a partial order on  $S_\rho$  by  $f < g$  if  $p_f < p_g$ . We will prove that for all  $h \in H$  and  $f \in S_\rho$  there is a  $g \in S_\rho$  with  $g < f$  such that  $h \cdot f = f + g$ .

By Theorem 1 there are polynomials  $q_i \in R$  in  $x_1, \dots, x_{i-1}$  such that for  $h \in H$

$$h^{-1} n_1^{\alpha_1} \cdots n_t^{\alpha_t} h = n_1^{\alpha_1 + \beta_1} \cdots n_t^{\alpha_t + \beta_t}$$

where  $\beta_i = q_i(\alpha_1, \dots, \alpha_{i-1})$ . In particular  $q_i < x_i$ .

Let  $f \in S_\rho$  and set  $f' = h \cdot f$ , then  $p_{f'} = p_f(x_1 + q_1, x_2 + q_2, \dots, x_t + q_t)$ . By expanding and reordering the monomials in  $p_{f'}$  we get  $p_{f'} = p_f(x_1, \dots, x_t) + q(x_1, \dots, x_t)$ . The polynomial  $q$  is a sum of monomials from  $p_f$  in which some of the  $x_i$  have been replaced by  $q_i$ . The translation invariance of  $<$  and the fact that  $q_i < x_i$  imply that  $q < p_f$ . Since  $p_{f'}$  and  $p_f$  are polynomials corresponding to elements of  $S_\rho$ , there is an element  $g \in S_\rho$  such that  $p_g = q$ . This shows that  $h \cdot f = f + g$  and  $g < f$ .

Now let  $f_0 \in S_\rho$  and set  $f_{k+1} = h_0 \cdot f_k - f_k$  for  $k \geq 0$ . Since the order  $<$  satisfies the descending chain condition, there exists a  $K \geq 0$  such that  $h_0 \cdot f_K = f_K$ .

Since  $S_\rho$  is generated as a  $\mathbb{Z}$ -module by the set  $\{h_0^k c_{ij} \mid 1 \leq i, j \leq r; k \in \mathbb{Z}\}$ , we have that  $S_\rho$  is finite-dimensional as a  $\mathbb{Z}$ -module.

Let  $\{f_1, \dots, f_s\}$  be a  $\mathbb{Z}$ -basis of  $S_\rho$ . By subtracting basis elements from other basis elements if necessary, we may assume that  $\text{lm}(p_{f_i}) \neq \text{lm}(p_{f_j})$  for  $i \neq j$ . Therefore, all elements of this basis are comparable with respect to the order  $<$ . Suppose they have been ordered such that  $f_i < f_j$  if  $i < j$ . Then for  $h \in H$  the matrix of  $h$  with respect to this basis is the identity matrix plus a strictly upper triangular matrix.  $\square$

**Proposition 6** *Suppose that  $\rho$  is a faithful representation of  $N$ . Then the representation  $\sigma : H \rightarrow GL(S_\rho)$  afforded by  $S_\rho$  is also faithful on  $N$ . Furthermore,  $\sigma$  is a faithful representation of  $H$  or  $h_0 n = n h_0$  for all  $n \in N$ .*

*Proof:* The  $\mathbb{Z}N$ -module  $S_\rho$  contains the  $\mathbb{Z}N$ -submodule  $C_\rho$ . By Lemma 3,  $C_\rho$  is faithful if  $\rho$  is faithful.

Let  $h$  be an arbitrary element of  $H$ . Then  $h$  has the form  $h_0^k n$  where  $n \in N$  and  $k \in \mathbb{Z}$ . Now suppose that  $\sigma(h) = 1$ . Then  $(h_0^k n \cdot c_{ij})(a) = c_{ij}(a)$  for all  $1 \leq i, j \leq r$  or, equivalently,  $c_{ij}(h_0^{-k} a h_0^k n) = c_{ij}(a)$  for all  $1 \leq i, j \leq r$  and  $a \in N$ . Since  $C_\rho$  is a faithful  $\mathbb{Z}N$ -module, this is equivalent to  $h_0^{-k} a h_0^k n = a$  for all  $a \in N$ . Taking  $a = 1$ , implies  $n = 1$  and  $h_0^{-k} a h_0^k = a$  for all  $a \in N$ , hence  $\sigma(h_0^k) = 1$ . Since  $\sigma(h_0)$  is unipotent by the previous theorem, we have that  $\sigma(h_0) = 1$ , whence the second statement.  $\square$

## 4. The algorithm

We formulate the algorithm based on the results of the previous section. For that we set

$$E_2 = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}.$$

### Algorithm Representation

**Input:** a  $\mathcal{T}$ -group  $G$  and a polycyclic generating sequence  $u_1, \dots, u_m$  of  $G$ .

**Output:** a faithful representation  $\sigma : G \rightarrow U_r(\mathbb{Z})$ .

1. Let  $H_k$  be the subgroup of  $G$  generated by  $u_k, \dots, u_m$  for  $k = 1, \dots, m$ .
2. Let  $\rho_m : H_m \rightarrow U_2(\mathbb{Z})$  be the representation of  $H_m$  given by  $\rho_m(u_m) = E_2$ .
3. For  $i = m - 1, m - 2, \dots, 1$  do the following.

Let  $\rho_{i+1} : H_{i+1} \rightarrow U_s(\mathbb{Z})$  be a faithful unipotent matrix representation of  $H_{i+1}$ .

If  $u_i$  commutes with  $u_{i+1}, \dots, u_m$  then let  $\rho_i$  be the representation

$$\rho_i(u_j) = \left( \begin{array}{c|cc} \rho_{i+1}(u_j) & & \\ \hline & 1 & 1 \\ & 0 & 1 \end{array} \right)$$

for  $j = i + 1, \dots, m$  and

$$\rho_i(u_i) = \left( \begin{array}{ccc|cc} 1 & & & & \\ & \ddots & & & \\ & & 1 & & \\ \hline & & & 1 & 1 \\ & & & 0 & 1 \end{array} \right).$$

If  $u_i$  does not commute with  $u_{i+1}, \dots, u_m$ , then calculate the spaces  $C_{\rho_{i+1}}$  and  $S_{\rho_{i+1}}$ . Let  $\rho_i$  be the representation of  $H_i$  acting on  $S_{\rho_{i+1}}$ .

4. Return  $\rho_1$ .

**Proposition 7** *The algorithm **Representation** returns a faithful representation of a  $\mathcal{T}$ -group  $G$  by unipotent matrices over  $\mathbb{Z}$ .*

*Proof:* We prove that the representations  $\rho_i$  are faithful and by unipotent matrices. For  $\rho_m$  this is clear. Furthermore, suppose that this holds for  $\rho_{i+1}$ . If  $u_i$  commutes with  $u_{i+1}, \dots, u_m$  then it is straightforward to see that the map  $\rho_i$  is a group homomorphism, that it is faithful and by unipotent matrices. If on the other hand  $u_i$  does not commute with  $u_{i+1}, \dots, u_m$ , then  $\rho_i$  is by unipotent matrices by Theorem 5. Furthermore, it is faithful by Proposition 6.  $\square$

**Corollary 8** *Let  $G$  be a  $\mathcal{T}$ -group, then  $G$  has a faithful finite dimensional representation by unipotent matrices over the integers.*

**Example 9** We consider the following group:

$$G = \langle a, b, c, d, e, f \mid [b, a] = c, [c, a] = d, [d, a] = e, [d, b] = f, [e, b] = f, [d, c] = f^{-1} \rangle,$$

(trivial commutators of the generators have been omitted). The generators  $d, e, f$  generate an Abelian subgroup. It follows that for first two steps of the algorithm, the first half of Step 3 applies. Hence we get a representation  $\rho_4 : \langle d, e, f \rangle \rightarrow \text{GL}(6, \mathbb{Z})$  given by

$$\rho_4(d^m e^n f^p) = \begin{pmatrix} 1 & p & & & & \\ 0 & 1 & & & & \\ & & 1 & n & & \\ & & 0 & 1 & & \\ & & & & 1 & m \\ & & & & 0 & 1 \end{pmatrix}.$$

Now we extend  $\rho_4$  to a representation of the group generated by  $c, d, e, f$ . First

we calculate  $C_{\rho_4}$ . Since this is the space of all coefficients,  $C_{\rho_4}$  is spanned by four functions  $f_1, f_2, f_3, f_4$ , given by

$$\begin{aligned} f_1(d^m e^n f^p) &= 1 \\ f_2(d^m e^n f^p) &= m \\ f_3(d^m e^n f^p) &= n \\ f_4(d^m e^n f^p) &= p. \end{aligned}$$

In order to calculate  $S_{\rho_4}$  we let  $c$  act on these functions. For example:

$$c \cdot f_4(d^m e^n f^p) = f_4(c^{-1} d^m e^n f^p c) = f_4(d^m e^n f^{p-m}) = p - m.$$

Hence  $c \cdot f_4 = f_4 - f_2$ . In the same way one sees that  $c \cdot f_1 = f_1$ ,  $c \cdot f_2 = f_2$  and  $c \cdot f_3 = f_3$ . Hence  $S_{\rho_4} = C_{\rho_4}$ . Using column convention, the representation  $\rho_3$  is given by

$$\rho_3(c^l d^m e^n f^p) = \begin{pmatrix} 1 & m & n & p \\ 0 & 1 & 0 & -l \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

Analogously it is seen that  $C_{\rho_3}$  contains five functions and  $S_{\rho_3} = C_{\rho_3}$ . The representation  $\rho_2$  of the group generated by  $b, c, d, e, f$  is given by

$$\rho_2(b^k c^l d^m e^n f^p) = \begin{pmatrix} 1 & l & m & n & p \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & k-l \\ 0 & 0 & 0 & 1 & k \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

Hence  $C_{\rho_2}$  is spanned by six functions:

$$\begin{aligned} f_1(b^k c^l d^m e^n f^p) &= 1 \\ f_2(b^k c^l d^m e^n f^p) &= k \\ f_3(b^k c^l d^m e^n f^p) &= l \\ f_4(b^k c^l d^m e^n f^p) &= m \\ f_5(b^k c^l d^m e^n f^p) &= n \\ f_6(b^k c^l d^m e^n f^p) &= p. \end{aligned}$$

Before letting  $a$  act on  $C_{\rho_2}$  we calculate its conjugation action on the normal subgroup generated by  $b, c, d, e, f$ . By an induction argument it is readily seen that

$$a^{-1} b^k c^l d^m e^n f^p a = b^k c^{l+k} d^{m+l} e^{n+m} f^{p-\binom{l}{2}}.$$

Hence only the action of  $a$  on  $f_6$  promises to yield new functions. Now

$$a \cdot f_6(b^k c^l d^m e^n f^p) = f_6(b^k c^{l+k} d^{m+l} e^{n+m} f^{p-\binom{l}{2}}) = p - \binom{l}{2}.$$



So  $a \cdot f_6$  is not a linear combination of functions that we saw before. We define  $f_7$  by

$$f_7(b^k c^l d^m e^n f^p) = \frac{1}{2}l^2 - \frac{1}{2}l.$$

Going on:

$$\begin{aligned} a \cdot f_7(b^k c^l d^m e^n f^p) &= f_7(b^k c^{l+k} d^{m+l} e^{n+m} f^{p-\binom{l}{2}}) \\ &= \frac{1}{2}(l+k)^2 - \frac{1}{2}(l+k) \\ &= \frac{1}{2}l^2 - \frac{1}{2}l + kl + \frac{1}{2}k^2 - \frac{1}{2}k. \end{aligned}$$

Again we get a new function  $f_8$  defined by  $f_8(b^k c^l d^m e^n f^p) = kl + \frac{1}{2}k^2 - \frac{1}{2}k$ . Continuing:

$$\begin{aligned} a \cdot f_8(b^k c^l d^m e^n f^p) &= f_8(b^k c^{l+k} d^{m+l} e^{n+m} f^{p-\binom{l}{2}}) \\ &= k(l+k) + \frac{1}{2}k^2 - \frac{1}{2}k \\ &= kl + \frac{1}{2}k^2 - \frac{1}{2}k + k^2. \end{aligned}$$

This leads yet again to a new function:  $f_9(b^k c^l d^m e^n f^p) = k^2$ . But now the process stops; we have  $a \cdot f_9 = f_9$ . So we arrive at a 9-dimensional representation; we leave it to the reader to write down the matrices.

## 5. Implementation

We have implemented the algorithm in the computer algebra system **GAP 4**. In this system the basic functionality for dealing with  $\mathcal{T}$ -groups (e.g., representation of words in the group and the collection algorithm) is already present. In this section we outline the implementation of the algorithm **Representation**. For this we revert to the language of Section 3.

First of all we note that if  $h_0$  commutes with the generators  $n_1, \dots, n_t$ , then the representation is extended to  $H$  without problems. So suppose that this is not the case, and we have to compute the module  $S_\rho$ . The main problem that we have to deal with is due to the fact that the dual space  $(\mathbb{Z}N)^*$  is infinite-dimensional. However since we are only interested in the finite-dimensional subspace  $S_\rho$  there is a way around this problem. First we deal with the problem of representing a function in  $S_\rho$  on a computer. If  $c$  is a coefficient of  $\rho$  then this is easy: we only need to store the position  $ij$  such that  $c(n)$  is the  $ij$ -th coefficient of the matrix  $\rho(n)$ . Furthermore, for  $f = h_0^k \cdot c$  we store the integer  $k$  together with the position  $ij$ . Then we can calculate the value of  $f(a)$  for all  $a \in \mathbb{Z}N$ . Also because any function in  $S_\rho$  is a linear combination of functions of the form  $h_0^k \cdot c$  we can represent all elements of  $S_\rho$ .

Now we consider the problem of doing linear algebra inside  $S_\rho$ . For that we

need to represent any element as a vector (list of coefficients). Then using Gaussian elimination we can find linearly independent sets, calculate matrices of endomorphisms of  $S_\rho$  and so on. For this we choose a finite set  $A$  of elements of the group  $N$  and we represent an element  $f$  of  $S_\rho$  as a vector  $(f(a))_{a \in A}$ . The set  $A$  is called a *discriminating set* for  $S_\rho$  if for every  $f \in S_\rho$  such that  $f \neq 0$  there is an  $a \in A$  such that  $f(a) \neq 0$ . Since  $S_\rho$  is finite-dimensional, discriminating sets for  $S_\rho$  exist, and a smallest discriminating set consists of  $\dim S_\rho$  elements. Unfortunately we only have the following rather crude method for selecting a discriminating set. Initially we let  $A$  be the set of all elements of  $N$  of degree bounded by some limit  $d \geq 1$ . Here we choose  $d$  large enough to ensure that  $A$  is a discriminating set for  $C_\rho$ . In particular we have that 1, along with the generators  $n_1, \dots, n_t$  are in  $A$ . Using this discriminating set we calculate the closure of  $C_\rho$  under the action of  $h_0$ . If  $A$  happens to be a discriminating set for  $S_\rho$ , then this will give us a basis of  $S_\rho$ . If  $A$  is not a discriminating set for  $S_\rho$ , then two things could go wrong: the resulting representation might not be a group homomorphism, or it might not be faithful. In the first case we increase the bound  $d$ , and start again. On the other hand, if the resulting representation is a group homomorphism, then it is also faithful. To see this we use the notations from the proof of Proposition 6. In this case we get that  $f(h_0^{-k} a h_0^k n) = f(a)$  for  $f \in \overline{S}_\rho$  and  $a \in A$ , where  $\overline{S}_\rho$  is the space that we computed. Since  $\overline{S}_\rho$  contains  $C_\rho$  we see that  $h_0^{-k} a h_0^k n = a$  for all  $a \in A$ . As  $A$  contains 1 we have that  $n = 1$ . Since  $\overline{S}_\rho$  is a quotient module of  $S_\rho$  we have that  $h_0$  acts by a unipotent matrix on  $\overline{S}_\rho$  as well. So again we get that  $h_0$  commutes with all elements of  $A$ , and in particular with the generators  $n_i$ . But this is excluded, and therefore the representation is faithful.

## 6. Examples

Table 1 lists a number of experimental results obtained with the implementation of the algorithm in **GAP 4**. All computations were done on a Linux system with a 600MHz Pentium III processor and 40MB of working memory for **GAP**. We use the following naming conventions. First,  $U_k(\mathbb{Z})$  denotes as before the full unitriangular group over  $\mathbb{Z}$ . Furthermore,  $F(k, n)$  is the free nilpotent group with  $k$  generators and class  $n$ , and  $E(k, n)$  is the largest, nilpotent  $k$ -generator group satisfying the  $n$ -th Engel identity. An asterisk  $*$  at the name of a group indicates that we have taken the largest torsion free quotient of the named group. This can be done by factoring out the torsion subgroup using the **GAP 4** package ‘Polycyclic’ [2].

From Table 1 we see that the algorithm is efficient enough to be able to deal with groups with rather large Hirsch length. However, the running times and dimensions obtained appear to increase exponentially. Also the dimensions of the modules found by the algorithm are generally much smaller than the ones constructed in [8].

Description	Class	Hirsch length	Dimension	Time
$U_2(\mathbb{Z})$	1	1	2	0
$U_3(\mathbb{Z})$	2	3	3	0
$U_4(\mathbb{Z})$	3	6	7	0.4
$U_5(\mathbb{Z})$	4	10	16	12.3
$U_6(\mathbb{Z})$	5	15	35	181.3
$F(2, 2)$	2	3	3	0
$F(2, 3)$	3	5	6	0.3
$F(2, 4)$	4	8	10	2.3
$F(2, 5)$	5	14	20	119.7
$F(3, 2)$	2	6	6	0.2
$F(3, 3)$	3	14	17	14.9
$\langle x, y \mid [y, x, x], [y, x, y, y] \rangle^*$	5	6	11	1.7
$E(2, 3)^*$	3	6	6	0.3
$E(3, 3)^*$	4	17	26	151.4
$E(4, 2)$	6	11	19	19.0

**Table 1:** Experimental results for the algorithm. The fourth column displays the dimension of the module obtained by the algorithm. The fifth column contains the running time in seconds.

## References

- [1] N. Bourbaki. *Groupes et Algèbres de Lie, Chapitre I*. Hermann, Paris, 1971.
- [2] B. Eick and W. Nickel. *Polycyclic*, 2000. A GAP package, see [3].
- [3] The GAP Group, Aachen, St Andrews. *GAP – Groups, Algorithms, and Programming, Version 4.2*, 2000. (<http://www-gap.dcs.st-and.ac.uk/~gap>).
- [4] W. A. de Graaf. *Lie Algebras: Theory and Algorithms*, volume 56 of *North-Holland Mathematical Library*. Elsevier Science, 2000.
- [5] P. Hall. Nilpotent groups. Notes of Lectures given at the Canadian Mathematical Congress, University of Alberta, 1957.
- [6] M. Hall, Jr. *The Theory of Groups*. Macmillan, New York, 1959.
- [7] C. R. Leedham-Green and L. H. Soicher. Collection from the left and other strategies. *J. Symbolic Comput.*, 9:665–675, 1990.
- [8] E. H. Lo and G. Ostheimer. A practical algorithm for finding matrix representations for polycyclic groups. *J. Symbolic Comput.*, 28(3):339–360, 1999.
- [9] D. Segal. *Polycyclic Groups*. Cambridge University Press, 1994.
- [10] C. C. Sims. *Computation with Finitely Presented Groups*. Cambridge University Press, Cambridge, 1994.

- [11] M. R. Vaughan-Lee. Collection from the left. *J. Symbolic Comput.*, 9:725–733, 1990.