Nonlinear Mathematical Physics 1995, V.2, N 3–4, 367–373.

Lie Algebras and Superalgebras Defined by a Finite Number of Relations: Computer Analysis

V.P. GERDT and V.V. KORNYAK

Laboratory of Computing Techniques and Automation,
Joint Institute for Nuclear Research, 141980 Dubna, Russia

Abstract

The presentation of Lie (super)algebras by a finite set of generators and defining relations is one of the most general mathematical and algorithmic schemes of their analysis. It is very important, for instance, for investigation of the particular Lie (super)algebras arising in different (super)symmetric physical models. Generally, one can put the following question: what is the most general Lie algebra or superalgebra satisfying to the given set of Lie polynomial equations? To solve this problem, one has to perform a large volume of algebraic transformations which sharply increases with growth of the number of generators and relations. By this reason, in practice, one needs to use a computer algebra tool. We describe here an algorithm and its implementation in C for constructing the bases of finitely presented Lie (super)algebras and their commutator tables.

1 Introduction

Finitely presented algebras are determined by a finite number of *generators* subject a finite number of *relations* having a form of polynomials in the algebra. The investigation of finitely presented Lie (super)algebras is one of the most important problems of combinatorial algebra [1]. This problem is of great practical importance covering applications ranging from mathematical physics, e.g., in the theory of integrable nonlinear partial differential equations [2], to mathematics and theoretical computer science, e.g., in studying the word problem, generally algorithmically unsolvable, in noncommutative and nonassociative combinatorial algebra.
Some examples of finitely presented algebras:

1. Any finite-dimensional algebra.

2. Kac and Kac-Moody (super)algebras with their generalization known as Borcherds algebras [3].

3. Lie (super)algebras of the string theories: Virasoro, Neveu-Schwarz and Ramond algebras.

4. Any simple finite-dimensional Lie algebra can be generated by two elements with the number and structure of relations independent on the rank of the algebra. This allows one to define such objects as Lie algebras of matrices of a complex size $\mathbf{sl}(\lambda)$, $\mathbf{o}(\lambda)$ and $\mathbf{sp}(\lambda)$, where λ might be any complex number or even ∞ [4]. In a similar way, one can define some Lie superalgebras of supermatrices of a complex size [5].

Below we describe briefly an algorithm and its C implementation for determining the explicit structure of an finitely presented Lie (super)algebra from defining relations, i.e., for constructing its basis and commutator table. In fact, our algorithm produces the Gröbner basis [1] for noncommutative and nonassociative cases. The algorithm and its actual implementation are illustrated by a rather simple example arising in mathematical physics.

2 Algorithm and Its Implementation

Input: The set of generators $X = \{x_1, x_2, \ldots\}$ with prescribed $\mathbf{Z}_2$ parities $d_i = 0, 1$ and positive integer weights w_i (= 1 by default); set of scalar parameters $P = \{p_1, p_2, \ldots\}$ if they present in the relations; set of defining relations $R = \{r_1, r_2, \ldots\}$, where r_i are Lie polynomials with coefficients from the commutative ring $\mathbf{Z}[p_1, p_2, \ldots]$.

Output: The reduced set of relations (Gröbner basis) $\tilde{R} = \{\tilde{r}_1, \tilde{r}_2, \ldots\}$; the list of basis elements $E = \{e_1, e_2, \ldots\}$; the commutator table $[e_i, e_j] = c_{ij}^k e_k$, where c_{ij}^k are the structure constants; the table of expressions containing p_i and considered as nonzeros during computation. Particular values of p_i may cause a branching of computation and, possibly, of the resulting algebra structure; dimensions of homogeneous components in the obtained Lie (super)algebra.

The algorithm includes the following principal steps:

1. **Reduction of the initial set R to an equivalent canonical form $\tilde{R}$**. In $\tilde{R} = \{\tilde{r}_1, \tilde{r}_2, \ldots\}$ all the relations and their algebraic consequences are mutually reduced, i.e., all possible substitutions in $\tilde{r}_i$ are done.

2. **Construction of the Lie (super)algebra basis**. Some basis elements are obtained at Step 1 as Lie (sub)monomials of $\tilde{r}_i$, but in the infinite-dimensional case the basis must be completed by the *regular* commutators of already existing basis elements. The term *regular* means basis monomials of a *free* algebra, i.e., algebra with the empty set R.

3. **Construction of the commutator table**. Here the commutators of the basis elements obtained at Step 2 are computed by the direct commutating with the further reduction of the resulting expression modulo the relations $\tilde{R}$.

Step 1 of the algorithm is ideologically similar to construction of Gröbner bases for polynomial ideals in a commutative algebra [6]. In our case, however, we deal with noncommutative and, moreover, with nonassociative monomials. One can easily prove that for constructing algebraic consequences it is sufficient to multiply relations not by arbitrary Lie monomials but by generators only, and even by those of them which do not form a regular monomial with the leading monomial of the relation. These facts allow us to increase the efficiency of the algorithm considerably.

All computations, starting with reading the input relations, are executed modulo Jacobi identities and the relations have been treated. This allows us to minimize resimplification of the calculated structures.

The algorithm has been implemented in the C language. The source code has the total length about 7500 lines and contains about 140 C functions realizing: top-level algorithms, Lie (super)algebra operations, manipulation with scalar polynomials, multiprecision integer arithmetic, substitutions, list processing, input and output, etc.

3 Sample Session

The following session file is the result of applying the program to the relations obtained in the investigation [7] of symmetries of $N = 1$ Manin-Radul [2] superization of the KdV equation. The relations contain two even generators x_1 and x_2 and odd generator y. We deform the original system by two parameters a and b to demonstrate the problem of classification.

```
--------------------------------------------------------------------
The program for construction of finitely presented Lie superalgebras
           Hall numeration. Version of April 8, 1995
               V. P. Gerdt and V. V. Kornyak
--------------------------------------------------------------------

Enter name of existing or new input file -> skdvab.in
Input data:
Generators: x_2 -y x_1;
Parameters: a b;
Relations:
[[[y,x_1],x_1],x_1];
[y,x_2];
[y,[[[y,x_1],x_1],y]];
[y,[[y,x_1],[y,x_1]]] - a [[y,[[y,x_1],y]],x_1];
[x_1,x_2] - [[y,[[y,[[y,x_1],y]],y]],y];
[x_1,[[y,x_1],y]] + [[y,x_1],[y,x_1]] + [[[y,x_1],x_1],y];
[x_1,[[y,x_1],[y,x_1]]] + b [x_1,[[[y,x_1],x_1],y]];
[x_1,[[y,[[y,x_1],y]],y]] - 3 [[y,x_1],y] - [[[y,[[y,x_1],y]],x_1],y];

Right-normed output for Lie monomials? (y/n) -> n
Standard grading assumes unit weight for every generator.
Do you want to use a different grading? (y/n) -> n
Enter limiting number for relations -> 20

Initial relations:

(1)  [x ,y] = 0
       2

(2)  [x ,[x ,[y,x ]]] = 0
       1   1     1

(3)  [x ,[y,[y,[y,x ]]]] = 0
       1           1
```

(4) $[[y,x_1],[y,[y,x_1]]] = 0$

(5) $(2\,b - 2)\,[[y,x_1],[x_1,[y,x_1]]] + b\,[x_1,[x_1,[y,[y,x_1]]]] = 0$

(6) $[x_1,[y,[y,[y,[y,x_1]]]]] - 3\,[y,[y,x_1]] = 0$

(7) $[y,[y,[y,[y,[y,[y,x_1]]]]]] + [x_2,x_1] = 0$

Nonzero parametric coefficients:

(1) $a - 2$

(2) $b - 1$

(3) $b^2 + b - 2$

Reduced relations:

(1) $[x_2,y] = 0$

(2) $[x_2,x_1] = 0$

(3) $[y,[y,x_1]] = 0$

(4) $[x_1,[x_1,[y,x_1]]] = 0$

(5) $[[y,x_1],[x_1,[y,x_1]]] = 0$

(6) $[[x_1,[y,x_1]],[x_1,[y,x_1]]] = 0$

Basis elements:

(1) $E_1 = x_2$

(2) $O_2 = y$

(3) $E_3 = x_1$

(4) $E_4 = [y,y]$

(5) $O_5 = [y,x_1]$

```
(6)  O   = [x ,[y,x ]]
      6     1     1

(7)  E   = [[y,x ],[y,x ]]
      7        1      1

Nonzero commutators of basis elements:

(1)  [O ,O ] = E
       2  2     4

(2)  [O ,E ] = O
       2  3     5

(3)  [E ,O ] = O
       3  5     6

(4)  [O ,O ] = E
       5  5     7

(5)  [O ,O ] = E
       2  6     7

Dimensions of homogeneous components:

dim G   = 3
     1

dim G   = 2
     2

dim G   = 1
     3

dim G   = 1
     4

Time: 0.05 sec
Number of relations:    15  Relation space:       120 bytes
Number of ordinals:     40  Ordinal space:        480 bytes
Number of nodes:        50  Node space:           600 bytes
Total space: 1200 bytes
```

Here E_i and O_i are even and odd basis elements, respectively. In the case of an infinite-dimensional algebra, the program prints out only those commutators which can be expressed in terms of the basis elements that have been computed.

In the above example, the chosen ordering among generators $x_2 \prec y \prec x_1$ provides the minimal number of reduced relations in the output. As well as for the commutative Gröbner bases method, the final structure of the reduced relations and even their number essentially depend on the ordering chosen. It can be easily seen that for the generic values of parameters a and b we have a seven-dimensional nilpotent Lie superalgebra. A branching of the algebra structure is possible at the values of parameters $a = 2$, $b = 1$ and $b = -2$. The computations with these particular values show that the choice $b = 1$

or $b = -2$ leads to the same algebra structure, whereas at $a = 2$ the algebra becomes infinite-dimensional. In [7], this algebra at $a = 2$ and $b = 1$ has been identified with the product of some seven-dimensional nilpotent algebra and the positive subalgebra of the twisted Kac-Moody superalgebra $C^{(2)}(2)$.

4 Conclusion

We have tested the program on the *standard relations* $[h_i, h_j] = 0$, $[e_i, f_j] = \delta_{ij} h_j$, $[h_i, e_j] = a_{ji} e_j$, $[h_i, f_j] = -a_{ji} f_j$, for *Chevalley generators* e_i, f_i, h_i with *Serre relations* $(ad\, e_i)^{1-a_{ji}} e_j = 0$, $(ad\, f_i)^{1-a_{ji}} f_j = 0$ added to them for all simple Lie algebras of the rank up to 10. Here a_{ij} is the Cartan matrix, $i, j = 1, ..., rank$. The most cumbersome computations among these algebras are related to the exceptional algebra E_8. Here the number of initial relations is 290. The program generates a Gröbner basis which contains 23074 relations involving Lie monomials up to 58 degree with Lie algebra basis elements going up to 29 degree. The task requires 15 min 36 sec of computing time and 815516 bytes of memory on an 25 MHz MS-DOS based AT/386 PC. One can see that the implementation is rather efficient.

Unlike commutative algebra, where such an universal algorithmic tool for analysis of polynomial ideals as the Gröbner basis method has been developed,[6] its generalizations [1] to noncommutative and, especially, to nonassociative algebras are still far from being of practical interest. Moreover, because of very serious mathematical and algorithmic problems are still to be solved, there are only a few packages implementing the noncommutative Gröbner basis technique, and no one of them so far is able to deal with nonassociative algebras. It justifies the practical use of other algorithmic methods. Among them there is one based on the straightforward verification of Jacobi identities [8, 9].

Our approach reveals some common features with the involutive approach for commutative algebra [10]. The latter can be considered as another algorithmic method to Gröbner basis construction, different from Buchberger's algorithm [6]. By this reason, the further analysis of our method could give a new insight to generalization of the Gröbner basis approach to Lie (super)algebras.

Acknowledgments

We are grateful to P.K.H. Gragert, D.A. Leites and especially to V.A. Ufnarovsky for fruitful discussions. This work was supported in part by the INTAS project No. 93-2058.

References

[1] Ufnarovsky V.A., *Itogi Nauki i Tekhniki. Modern Problems in Mathematics. Fundamental Branches*, 1990, V.57, VINITI, Moscow, p.5 (in Russian).

[2] Manin Yu.I. and Radul A.O., *Commun. Math. Phys.,* 1985, V.98, 65.

[3] Gebert R.W., Beyond Affine Kac-Moody Algebras in String Theory, DESY 94-209, Hamburg, 1994.

[4] Grozman P. and Leites D., Defining Relations Associated with the Principal $\mathbf{sl}(2)-subalgebras$ (to appear).

[5] Grozman P. and Leites D., Lie Superalgebras of Supermatrices of Complex Size (to appear).

[6] Buchberger B., in Recent Trends in Multidimensional System Theory, Editor N.K.Bose, D.Reidel, 1985, p.184.

[7] Roelofs G.H.M., Ph.D. Thesis, Univ. of Twente, Enschede, Netherlands, 1993.

[8] Gragert P.K.H., *Acta Applicandae Mathematicae,* 1989, V.16, 231.

[9] Roelofs G.H.M., The LIESUPER Package for REDUCE, Memorandum 943, Univ. of Twente, Netherlands, 1991.

[10] Gerdt V.P. and Blinkov Yu. A., Involutive Polynomial Bases, Publication IT-95-271, LIFL USTL, Lille, 1993. Submitted in J. Symb. Comp.