Записки научных
семинаров ПОМИ
Том 258, 1999 г.

V. V. Kornyak

A PROGRAM FOR COMPUTING COHOMOLOGIES OF LIE SUPERALGEBRAS OF VECTOR FIELDS

ABSTRACT. The cohomology of Lie (super)algebras has many important applications in mathematics and physics. At present, because of the need for very tedious algebraic computation, the explicitly computed cohomology for different classes of Lie (super)algebras is known only in a few cases. That is why the application of computer algebra methods is important for this problem. We describe here an algorithm and its C implementation for computing the cohomology of Lie algebras and superalgebras. When elaborating the algorithm we paid primary attention to cohomology in trivial, adjoint and coadjoint modules for Lie algebras and superalgebras of the formal vector fields. These algebras have found many applications to modern supersymmetric models of theoretical and mathematical physics. As an example, we present 3- and 5-cocycles from the cohomology in the trivial module for the Poisson algebra $Po(2)$, as found by computer.

1. INTRODUCTION

There are many applications of the Lie (super)algebra cohomology in mathematics: characteristic classes of foliations; invariant differential operators; MacDonald-type combinatorial identities etc. (see [1] for details) Besides, the cohomology is used widely in mathematical and theoretical physics [2]: construction of central extensions and deformations for Lie superalgebras; construction of supergravity equations for N-extended Minkowski superspace and search for possible models for these superspaces; study of stability for nonholonomic systems like ballbearings, gyroscopes, electro-mechanical devices, waves in plasma, etc.; description of an analogue of the curvature tensor for nonlinear nonholonomic constraints; new methods for the study of integrability of dynamical systems.

General definitions and properties of cohomology of Lie algebras and superalgebras are described in [1]. Let's recall briefly some basic definitions.

This work was supported in part by INTAS project No. 96-184 and RFBR project No. 96-01-01860.

A *Lie superalgebra* is a $\mathbf{Z}_2$-graded algebra over a commutative ring K with a unit:

$$L = L_{\bar{0}} \oplus L_{\bar{1}},\ u \in L_\alpha,\ v \in L_\beta,\ \alpha, \beta \in \mathbf{Z}_2 = \{\bar{0}, \bar{1}\} \Longrightarrow [u, v] \in L_{\alpha+\beta}.$$

The elements of $L_{\bar{0}}$ and $L_{\bar{1}}$ are called *even* and *odd*, respectively. We shall assume K is one of the fields $\mathbf{C}$ or $\mathbf{R}$. By definition, the *Lie product* (shortly, *brackets*) $[\ ,\ ]$ satisfies the following axioms

$$\begin{aligned} [u, v] &= -(-1)^{p(u)p(v)}[v, u], && skew - symmetry, \\ [u, [v, w]] &= [[u, v], w] + (-1)^{p(u)p(v)}[v, [u, w]], && Jacobi\ identity, \end{aligned}$$

where $p(a)$ is the parity of element $a \in L_{p(a)}$.

A *module* over a Lie superalgebra A is a vector space M (over the same field K) with a mapping $A \times M \to M$, such that $[a_1, a_2]m = a_1(a_2 m) - (-1)^{p(a_1)p(a_2)} a_2(a_1 m)$, where $a_1, a_2 \in A$, $m \in M$. The most important for our purposes are *trivial* (M is arbitrary vector space, e.g., $M = K$; $am = 0$), *adjoint* ($M = A; am = [a, m]$) and *coadjoint* ($M = A'; am = \{a, m\}$ is coadjoint action) modules.

A *cochain complex* is a sequence of linear spaces C^k with linear mappings d^k

$$0 \to C^0 \xrightarrow{d^0} \cdots \xrightarrow{d^{k-2}} C^{k-1} \xrightarrow{d^{k-1}} C^k \xrightarrow{d^k} C^{k+1} \xrightarrow{d^{k+1}} \cdots,$$

where the linear space $C^k = C^k(A; M)$ is a super skew-symmetric k-linear mapping $A \times \cdots \times A \to M$, $C^0 = M$ by definition. The super skew-symmetry means symmetry w.r.t. transpositions of odd elements of A and antisymmetry for all other transpositions. Elements of C^k are called *cochains*.

The linear mapping d^k (or, briefly, d) is called the *differential* and satisfies the following property: $d^k \circ d^{k-1} = 0$ (or $d^2 = 0$).

The cochains mapped into zero by the differential are called the *cocycles*, i.e., the space of cocycles is

$$Z^k = \text{Ker } d^k = \{C^k \mid dC^k = 0\}.$$

Those cochains which can be represented as differentials of other cochains are called the *coboundaries*, i.e., the space of coboundaries is

$$B^k = \text{Im } d^{k-1} = \{C^k \mid C^k = dC^{k-1}\}.$$

Any coboundary is, obviously, a cocycle.

The non-trivial cocycles, i.e., those of them which are not coboundaries, form the *cohomology.* In other words, the cohomology is the quotient space

$$H^k(A;M) = Z^k/B^k.$$

The explicit form of the differential for a Lie superalgebra is

$$dC(e_0,\dots,e_q;O_{q+1},\dots,O_k) =$$

$$\sum_{i<j}^{q}(-1)^j C(e_0,\dots,e_{i-1},[e_i,e_j],\dots,\widehat{e_j},\dots,e_q;O_{q+1},\dots,O_k) +$$

$$(-1)^{q+1}\sum_{i=0}^{q}\sum_{j=q+1}^{k} C(e_0,\dots,e_{i-1},[e_i,O_j],\dots,e_q;O_{q+1},\dots,\widehat{O_j},\dots,O_k) +$$

$$(-1)^{i+1}\sum_{i=q+1}^{k-1}\sum_{j=q+2}^{k} C(e_0,\dots,e_q;O_{q+1},\dots,O_{i-1},[O_i,O_j],\dots,\widehat{O_j},\dots,O_k) +$$

$$\sum_{i=0}^{q}(-1)^{i+1} e_i C(e_0,\dots,\widehat{e_i},\dots,e_q;O_{q+1},\dots,O_k) +$$

$$(-1)^q \sum_{i=q+1}^{k} O_i C(e_0,\dots,e_q;O_{q+1},\dots,\widehat{O_i},\dots,O_k).$$

Here e_i and O_i are even and odd elements of the algebra, respectively, and the hat "$\widehat{\ }$" marks the omitted elements.

Here are some properties and statements we use in the sequel.

An algebra and a module are called *graded* if they can be presented as sums of homogeneous components in a way compatible with the algebra brackets and the action of the algebra on the module:

$$A = \oplus_{g\in G} A_g,\ M = \oplus_{g\in G} M_g,\ [A_{g_1},A_{g_2}] \subset A_{g_1+g_2},\ A_{g_1}M_{g_2} \subset M_{g_1+g_2},$$

where G is some abelian (semi)group. The grading in the algebra and module induces a grading on cochains and, hence, in the cohomology:

$$C^*(A;M) = \oplus_{g\in G} C^*_g(A;M), \quad H^*(A;M) = \oplus_{g\in G} H^*_g(A;M).$$

This property allows one to compute the cohomology separately for different homogeneous components; this is especially useful when the homogeneous components are finite–dimensional.

If there is an element $a_0 \in A$, such that eigenvectors of the operator $a \mapsto [a_0,a]$ form a (topological) basis of algebra A, then $H^*(A) \simeq H^*_0(A)$. In other words, all the non-trivial cocycles of the cohomology in the trivial module lie in the zero grade component. The element a_0 is called an *internal grading element.* If also eigenvectors of the operator $m \mapsto a_0 m$

form a topological basis of module M, then the same statement holds for the cohomology in the module M: $H^*(A; M) \simeq H^*_0(A; M)$.

Any Lie superalgebra acts on its cohomology trivially, i.e., $A \circ H^k(A; M) = 0$. The explicit formula for the algebra action on cochains is

$$a \circ C^k(a_1, \dots, a_k) = -\sum_{i=1}^{k} C^k(a_1, \dots, [a, a_i], \dots, a_k) + aC^k(a_1, \dots, a_k).$$

The triviality of the action allows one to reduce computation in some cases.

2. Lie superalgebras of vector fields

Below a list of the main Lie superalgebras of formal vector fields is given [3]. We consider some sets of even (x_i, q_i, p_i, t) and odd (called also *Grassmann*) variables (U_i, W). In many cases the vector fields can be expressed in terms of generating functions. The coordinates of vector fields and generating functions are assumed to be formal power series in the even and odd variables. Note that all the algebras depending only on the odd variables are finite-dimensional. All these algebras are graded due to prescribed grading of the variables. There are some *standard* gradings for the variables: all variables x_i, q_i, p_i, U_i have grade 1 and the separate variables t, W have grade 2. *Non-standard* gradings with zero grades for some odd variables are possible too. The divergence-free algebras are called *special*. The symbol **Z** denotes the center of an algebra.

1. *General vectorial superalgebra* $\mathbf{W}(n \mid m)$ or $\mathbf{v}ect(n \mid m)$ Variables: $x_1, \dots, x_n; U_1, \dots, U_m$ The brackets denote the supercommutator of vector fields of the form $\sum\limits_{i}^{n} f_i \frac{\partial}{\partial x_i} + \sum_{k}^{m} g_k \frac{\partial}{\partial U_k}$

2. *Special vectorial superalgebra* $\mathbf{S}(n \mid m)$ or $\mathbf{s}vect(n \mid m)$ consists of the elements from $\mathbf{W}(n \mid m)$ satisfying the condition $\sum\limits_{i}^{n} \frac{\partial f_i}{\partial x_i} + \sum\limits_{k}^{m} (-1)^{p(g_k)} \frac{\partial g_k}{\partial U_k} = 0$

3. *Poisson superalgebra* $\mathbf{P}o(2n \mid m)$
   Variables: $p_1, \ldots, p_n, q_1, \ldots, q_n; U_1, \ldots, U_m$
   Brackets:
$$\{f,g\}_{Pb} =$$
$$= \sum_i^n \left(\frac{\partial f}{\partial p_i} \frac{\partial g}{\partial q_i} - \frac{\partial f}{\partial q_i} \frac{\partial g}{\partial p_i} \right) - (-1)^{p(f)} \sum_k^m \frac{\partial f}{\partial U_k} \frac{\partial g}{\partial U_k}$$
   Hamiltonian superalgebra is $\mathbf{H}(2n \mid m) = Po(2n \mid m)/Z$.

In the case of purely odd superdimension there exists the special form of Hamiltonian superalgebra $\mathbf{S}H(0 \mid m)$.

4. *Contact superalgebra* $\mathbf{K}(2n+1 \mid m)$
   Variables: $t, p_1, \ldots, p_n, q_1, \ldots, q_n; U_1, \ldots, U_m$
   Brackets: $\{f,g\}_{Kb} = \Delta(f)\frac{\partial g}{\partial t} - \frac{\partial f}{\partial t}\Delta(g) - \{f,g\}_{Pb}$
   $\Delta(f) = 2f - E(f),\ E = \sum_i^n \left(p_i \frac{\partial}{\partial p_i} + q_i \frac{\partial}{\partial q_i}\right) + \sum_k^m U_k \frac{\partial}{\partial U_k}$
5. *Buttin superalgebra* $\mathbf{B}(n)$
   Variables: $q_1, \ldots, q_n; U_1, \ldots, U_n$
   Brackets: $\{f,g\}_{Bb} = \sum_i^n \left(\frac{\partial f}{\partial q_i} \frac{\partial g}{\partial U_i} + (-1)^{p(f)} \frac{\partial f}{\partial U_i} \frac{\partial g}{\partial q_i} \right)$
   Leites superalgebra is $\mathbf{L}e(n) = B(n)/Z$
6. *Special Buttin superalgebra* $\mathbf{S}B(n)$ is subalgebra of $\mathbf{B}(n)$ subject to the constraint $\sum_i^n \frac{\partial^2 f}{\partial q_i \partial U_i} = 0$ for generating function.
   Special Leites superalgebra is $\mathbf{S}Le(n) = SB(n)/Z$
7. *Odd contact superalgebra* $\mathbf{M}(n)$
   Variables: $q_1, \ldots, q_n; W, U_1, \ldots, U_n$
   Brackets: $\{f,g\}_{Mb} = \Delta(f)\frac{\partial g}{\partial W} + (-1)^{p(f)} \frac{\partial f}{\partial W}\Delta(g) - \{f,g\}_{Bb}$
   $\Delta(f) = 2f - E(f),\ E = \sum_i^n \left(q_i \frac{\partial}{\partial q_i} + U_i \frac{\partial}{\partial U_i}\right)$
8. *Special odd contact superalgebra* $\mathbf{S}M(n)$ is subalgebra of $\mathbf{M}(n)$ subject to the constraint $(1-E)\frac{\partial f}{\partial W} - \sum_i^n \frac{\partial^2 f}{\partial q_i \partial U_i} = 0$ for generating function.

3. Outline of algorithm and its implementation

To compute the cohomology one needs to solve the equation
$$dC^k = 0, \tag{1}$$

and throw away those solutions of (1) which can be expressed in the form

$$C^k = dC^{k-1}. \tag{2}$$

In some exceptional cases it is possible to solve equation (2) in closed form. To feel the character of computation let's consider one of the such examples. We'll extend the Witt algebra, i.e., algebra of vector fields on a circle $Vect(S^1)$, to the Virasoro algebra, the basic algebra of the string theory. The Virasoro algebra is a central extension of the Witt algebra. The central extension is determined by the second cohomology in the trivial module, thus we need to compute $H^2(Vect(S^1))$. The commutator table of the Witt algebra takes the form $[e_i, e_j] = (j-i)e_{i+j}, \quad i, j \in \mathbf{Z}$. One can see that e_0 is internal grading element: $[e_0, e_j] = je_j$. Hence, it is sufficient to consider only zero grade cochains. Expression (2) for coboundaries takes the form $dC(e_i, e_j) = -C([e_i, e_j]) = (i-j)C(e_{i+j})$, and, after transition to zero grade, we come to

$$dC(e_i, e_{-i}) = 2iC(e_0) \equiv \beta i.$$

As to determining equation (1) for cocycles, the following sequence of transformations including transition to zero grade $(i+j+k=0)$ and subsequent simplification of notation for cochains

$$\begin{aligned} dC(e_i, e_j, e_k) &= -C([e_i, e_j], e_k) + C([e_i, e_k], e_j) + C(e_i, [e_j, e_k]) \\ &\to (i-j)C(e_{i+j}, e_k) + (k-i)C(e_{i+k}, e_j) + (k-j)C(e_i, e_{j+k}) \\ &\to (i-j)C_{i+j,k} + (k-i)C_{i+k,j} + (k-j)C_{i,j+k} \\ &\to (i-j)C_{i+j,-i-j} + (2i+j)C_{j,-j} - (i+2j)C_{i,-i} \end{aligned}$$

leads to

$$(i-j)C_{i+j} + (2i+j)C_j - (i+2j)C_i = 0. \tag{3}$$

The general solution of recurrent equation (3) is $C_i = \alpha i^3 + \beta i$. This formula contains non-trivial term allowing to write the commutator table for the Virasoso algebra in the form $[e_i, e_j] = (j-i)e_{i+j} + \delta_{i,-j} i^3 c$, where c is a central element.

Generally, in the case of Lie superalgebras of vector fields, determining equation (1) is a system of linear homogeneous functional equations with integer arguments resembling (3) (usually with more terms and indices). Unfortunately there is no general method for solving such systems in closed form. Hence, we need to carry out the corresponding computation "numerically". There are several packages for computing cohomology of

Lie algebras and superalgebras written in *Reduce* [4], [5] and *Mathematica* [2]. Some new results were obtained completely or partly with the help of these packages. However, these packages, being based on general purpose computer algebra systems, appeared to be too inefficient for large real problems. In view of this, we wrote the program in C language.

The C code, of total length near 14000 lines, contains about 300 functions realizing top level algorithms, simplification of indexed objects, working with Grassmannian objects, exterior calculus, linear algebra, substitutions, list processing, input and output etc. As internal structures we use 8 types of lists for different objects. We represent Grassmann monomials by integer numbers using one-to-one correspondence between (binary codes of) non-negative integers and Grassmann monomials. This representation allows one efficiently to implement the operations with Grassmann monomials by means of the basic computer commands.

The program performs sequentially the following steps:

1. *Reading input information.*
2. *Constructing a basis* for the algebra. The basis can be read from the input file; otherwise the program constructs it from the definition of the algebra. Non-trivial computations at this step arise only in the case of divergence-free algebras. The basis elements of such algebras should satisfy some conditions. In fact, we should construct the basis elements of a subspace given by a system of linear equations. The task is thereby reduced to some problem of linear algebra combined with shifts of indices. For example, among the divergence-free conditions for the special Buttin algebra $\mathbf{S}B(3)$ there are the following two equations

$$ia_{ijk;UV} - (k+1)a_{i-1,j,k+1;VW} = 0,$$

$$ia_{ijk;UW} + (j+1)a_{i-1,j+1,k;VW} = 0.$$

Here $a_{ijk;UV}, \ldots$ are coefficients at the monomials $p^i q^j r^k UV, \ldots$ in the generating function; p, q, r and U, V, W are even and odd variables, respectively. First of all, we have to shift indices j and k in the second equation to reduce the last terms of both equations to the same multiindices. Then, using some simple tricks of linear algebra, we can easily construct the corresponding basis element

$$E_{ijk} =$$

$$= (k+1)p^i q^j r^k UV - jp^i q^{j-1} r^{k+1} UW + ip^{i+1} q^j r^{k+1} VW.$$

3. *Constructing the commutator table* for the algebra (if this table has not been read from the input file).
4. *Creating the general form* of expressions for coboundaries and determining equations for cocycles.
5. *Transition to a particular grade* in general expressions. At this step expressions for coboundaries take the form $\mathbf{x} = bt$, equations for cocycles take the form $\mathbf{Z}x = 0$, where vector $\mathbf{x}$ corresponds to C^k, parameter vector $\mathbf{t}$ corresponds to C^{k-1}, matrices $\mathbf{Z}, b$ correspond to the differential d. All these vector spaces are finite-dimensional for any particular grade.
6. *Computing the quotient space* $H^k(A; M) = Z^k/B^k$. Here cocycle subspace Z^k is given by relations $\mathbf{Z}x = 0$, and coboundary subspace B^k is given parametrically by $\mathbf{x} = bt$.

Substeps:

(a) Eliminate $\mathbf{t}$ from $\mathbf{x} = bt$ to get equations $\mathbf{B}x = 0$

(b) Reduce both relations $\mathbf{B}x = 0$ and $\mathbf{Z}x = 0$ to the canonical form by Gauss elimination. If rank$\mathbf{B}$ = rank$\mathbf{Z}$, then there is no non-trivial cocycle; otherwise go to Substep (c).

(c) Set $\mathbf{B}x = y$ and substitute these relations into $\mathbf{Z}x = 0$ to get relations $\mathbf{A}y = 0$. The *parametric* (non-leading) y's of the last relations are non-trivial cocycles; that is, they form a basis of the cohomology.

In fact, the above procedure is based on the relation for quotient spaces

$$Z/B = \frac{Y/B}{Y/Z},$$

where Y is an artificially introduced space, combining the above x's and y's.

7. *Output the non-trivial cocycles.*

4. Output for $H^5_{(-2)}(Po(2))$

Preliminary notes:

- $H^3_{(-4)}(Po(2))$ also contains non-trivial cocycle

$$C(E_{-1,-1}, E_{-1,0}, E_{0,-1}) = C(1, q, p),$$

other cocycles up to number 5 and grade 10 are trivial.

- We omit some elements in the below output for brevity.

Input file: IN Po(2).in

Input data:

<= *Algebra of 2D Poisson polynomial vector fields Po(2)* =>
Module type: Trivial.
Poisson superalgebra: 2.
Cohomology number: 5.

Even variables of vector field: p q

Basis elements of vector field superalgebra:

$$(1) \quad E_{ij} = p^{i+1} \quad q^{j+1}$$

Non-zero commutators of Lie superalgebra:

$$(1) \quad [E_{ij}, E_{kl}] = (i+il-j-jk-k+l)E_{i+k,j+l}$$

Enter grade -> -2

Expression for coboundaries:

$$
\begin{aligned}
dC^4 = {} & (o+or-p-pq-q+r)C(E_{ij},E_{kl},E_{mn},E_{o+q,p+r}) \\
& + (-m-mp+n+no+o-p)C(E_{ij},E_{kl},E_{m+o,n+p},E_{qr}) \\
& + (m+mr-n-nq-q+r)C(E_{ij},E_{kl},E_{m+q,n+r},E_{op}) \\
& + (k+kn-l-lm-m+n)C(E_{ij},E_{k+m,l+n},E_{op},E_{qr}) \\
& + (-k-kp+l+lo+o-p)C(E_{ij},E_{k+o,l+p},E_{mn},E_{qr}) \\
& + (k+kr-l-lq-q+r)C(E_{ij},E_{k+q,l+r},E_{mn},E_{op}) \\
& + (-i-il+j+jk+k-l)C(E_{i+k,j+l}, E_{mn},E_{op},E_{qr}) \\
& + (i+in-j-jm-m+n)C(E_{i+m,j+n},E_{kl},E_{op},E_{qr}) \\
& + (-i-ip+j+jo+o-p)C(E_{i+o,j+p},E_{kl},E_{mn},E_{qr}) \\
& + (i+ir-j-jq-q+r)C(E_{i+q,j+r},E_{kl},E_{mn},E_{op}) \\
& \text{‘}E_{ij} \wedge \text{‘}E_{kl} \wedge \text{‘}E_{mn} \wedge \text{‘}E_{op} \wedge \text{‘}E_{qr}
\end{aligned}
$$

Equation for cocycles:

$$
\begin{aligned}
dC^5 = {} & (-q-qt+r+rs+s-t)C(E_{ij},E_{kl},E_{mn},E_{op},E_{q+s,r+t}) \\
= {} & (o+or-p-pq-q+r)C(E_{ij}, E_{kl},E_{mn},E_{o+q,p+r},E_{st}) \\
& + (-o-ot+p+ps+s-t)C(E_{ij},E_{kl},E_{mn},E_{o+s,p+t},E_{qr})
\end{aligned}
$$

$$+ (-m-mp+n+no+o-p)C(E_{ij},E_{kl},E_{m+o,n+p},E_{qr},E_{st})$$
$$(m+mr-n-nq-q+r)C(E_{ij}, E_{kl},E_{m+q,n+r},E_{op},E_{st})$$
$$+ (-m-mt+n+ns+s-t)C(E_{ij},E_{kl},E_{m+s,n+t},E_{op},E_{qr})$$
$$+ (k+kn-l-lm-m+n)C(E_{ij},E_{k+m,l+n},E_{op},E_{qr},E_{st})$$
$$+ (-k-kp+l+lo+o-p)C(E_{ij}, E_{k+o,l+p},E_{mn},E_{qr},E_{st})$$
$$+ (k+kr-l-lq-q+r)C(E_{ij},E_{k+q,l+r},E_{mn},E_{op},E_{st})$$
$$+ (-k-kt+l+ls+s-t)C(E_{ij},E_{k+s,l+t},E_{mn},E_{op},E_{qr})$$
$$+ (-i-il+j+jk+k-l)C(E_{i+k,j+l},E_{mn},E_{op},E_{qr},E_{st})$$
$$+ (i+in-j-jm-m+n)C(E_{i+m,j+n},E_{kl},E_{op},E_{qr},E_{st})$$
$$+ (-i-ip+j+jo+o-p)C(E_{i+o,j+p},E_{kl},E_{mn},E_{qr},E_{st})$$
$$+ (i+ir-j-jq-q+r)C(E_{i+q,j+r},E_{kl},E_{mn},E_{op},E_{st})$$
$$+ (-i-it+j+js+s-t)C(E_{i+s,j+t},E_{kl},E_{mn},E_{op},$$
$$E_{qr})`E_{ij} \wedge `E_{kl} \wedge `E_{mn} \wedge `E_{op} \wedge `E_{qr} \wedge `E_{st} = 0$$

Coboundary component expressions in grade -2:

$$C(E_{-1,-1},E_{-1,0},E_{-1,1},E_{-1,2},E_{00}) = x_1 = - 6 t_1$$
$$C(E_{-1,-1},E_{-1,0},E_{-1,1},E_{-1,2},E_{1,-1}) = x_2$$
$$= - 6 t_2 + 4 t_5 - 2 t_{18}$$

.

.

.

$$C(E_{-1,0},E_{-1,1},E_{0,-1},E_{00},E_{1,-1}) = x_{46} = - t_{22}$$

where

$$t_1 = C(E_{-1,-1},E_{-1,0},E_{-1,1},E_{-1,2})$$
$$t_2 = C(E_{-1,-1},E_{-1,0},E_{-1,1},E_{01})$$

.

.

.

$$t_{31} = C(E_{-1,0},E_{-1,1},E_{0,-1},E_{1,-1})$$

Coboundary subspace equations (rank = 24):

$$2x_1 - 3x_3 = y_1 = 0$$

$$2x_2 + 4x_{14} - 3x_{18} - 2x_{34} = y_2 = 0$$

.
.
.

$$12x_6 - 4x_{16} + 6x_{26} - 9x_{39} - 6x_{42} + 36x_{46} = y_5 = 0$$

.
.
.

$$6x_{12} - 4x_{16} - 6x_{26} + 12x_{37} - 3x_{39} - 10x_{42} + 36x_{46} = y_{11} = 0$$

.
.
.

$$12x_{24} - 6x_{26} - 12x_{37} + 3x_{39} + 14x_{42} - 36x_{46} = y_{18} = 0$$

.
.
.

$$x_{33} + 2x_{41} + 3x_{44} = y_{24} = 0$$

Cocycle subspace equations (rank = 23):

$$2x_1 - 3x_3 = 0$$

$$2x_2 + 4x_{14} - 3x_{18} - 2x_{34} = 0$$

.
.
.

$$6x_6 - 2x_{16} + 6x_{24} - 6x_{37} - 3x_{39} + 4x_{42} = 0$$

.
.
.

$$3x_{12} - 2x_{16} + 6x_{24} - 6x_{26} + 2x_{42} = 0$$

.
.

$$x_{33} + 2x_{41} + 3x_{44} = 0$$

```
Nontrivial cocycles:
 (1)  12 C(E      ,E     ,E     ,E  ,E  )
            -1,-1  -1,0   0,-1   01  10

        - 6 C(E      ,E     ,E     ,E  ,E    )
                -1,-1  -1,0   0,-1   02  1,-1

        - 12 C(E      ,E     ,E     ,E  ,E  )
                 -1,-1  -1,1   0,-1   00  10

        + 3 C(E      ,E     ,E     ,E  ,E    )
                -1,-1  -1,1   0,-1   01  1,-1

        + 14 C(E      ,E     ,E     ,E  ,E    )
                 -1,-1  -1,2   0,-1   00  1,-1

       - 36 C(E     ,E     ,E     ,E  ,     E    )
                -1,0   -1,1   0,-1   00       1,-1
                        2 2
      = 12 C(1, q, p, p q , p  q)
                          3   2                2
      - 6 C(1, q, p, p q , p ) - 12 C(1, q , p,
            2          2       2   2
      p q, p  q) + 3 C(1, q , p, p q , p )
                     3         2
      + 14 C(1, q , p, p q, p )
                 2          2
      - 36 C(q, q , p, p q, p  )
```

Time: 7.17 sec

5. Conclusions

We have not found reports of an explicit computation of cohomology for the algebra $Po(2)$. In [6] some results of computation of cohomology for the algebra of Hamiltonian vector fields $H(2) = Po(2)/Z$ are presented. That computation was also based on the use of a computer. Though our program is oriented to work with algebras of vector fields, it can be easily adapted to work with Lie superalgebras of other types. It is necessary only to write some additional input functions. The current preliminary version of the program uses standard 32-bit words for integers. As our experience shows, the computation of more than 5th cohomologies and in more than 10th grades for $Po(2)$ leads to overflow of such integers. Thus, it is necessary to add to the program the arithmetic of arbitrary precision. It would be useful to add also the arithmetic of finite fields, because there is a lot of unsolved problems even for cohomology of finite-dimensional Lie algebras over such fields.

Acknowledgments. I am grateful to V. Gerdt and D. Leites for fruitful discussions and to O. Khudaverdian for useful advices.

References

1. D. B. Fuks, *Cohomology of Infinite Dimensional Lie Algebras.* Consultants Bureau, New York, 1987.
2. P. Grozman, D. Leites, *Mathematica-aided study of Lie algebras and their cohomology. From supergravity to ballbearings and magnetic hydrodynamics.* **IMS**'97 (to appear).
3. D. Leites, *Lie superalgebras.* In: *Modern Problems of Mathematics. Recent developments*, v. 25, VINITI, Moscow, 1984, 3–49; (in Russian; English translation in: JOSMAR v. 30(6), 1985, 2481–2512).
4. D. Leites, G. Post, *Cohomology to compute.* In: Kaltofen E, Watt S.M. (eds.), *Computers and Mathematics*, Springer, NY ea, 1989, 73–81.
5. G. Post, N. von Hijligenberg, *Calculation of Lie algebra cohomology by computer.* Memo# 833, Faculty of Appl. Math. Univ. Twente, 1989; id. ibid. #928, 1991.
6. I. M. Gel'fand, D. I. Kalinin, and D. B. Fuks, *On Cohomology of Lie Algebra of Hamiltonian Formal Vector Fields.* Funkts. Anal. Prilozhen., **6**, 1972, 25–29 (in Russian).

JINR, Dubna, Russia

Поступило 12 мая 1999 г.