90-508

сообщения объединенного института ядерных исследований
дубна

E5-90-508

I.R.Akselrod[1], V.P.Gerdt, V.E.Kovtun[2],
V.N.Robuk[1]

CONSTRUCTION OF A LIE ALGEBRA BY A SUBSET OF GENERATORS AND COMMUTATION RELATIONS

[1]Phys.-Techn. Inst of Low Temperature, Kharkov, USSR
[2]Kharkov State University, Kharkov, USSR

1990

§1 Introduction

Many applied problems lead to systems of nonlinear algebraic equations for some (generally non-commutative) set of variables. In the commutative case the most developed way of analyzing and solving algebraic systems is based on the Groebner basis construction for polynomial ideal generated by a set of polynomials of a given system [1]. In the framework of this approach the universal effective algorithms have been developed and implemented in modern computer algebra systems [2]. In the general (non-commutative) case finite Groebner basis for a given finite set of polynomials could not exist. This fact essentially restricts an applicability of the basis Groebner technique in non-commutative case [3].

In the present paper we propose another approach to the problem of solvability of a given system of nonlinear algebraic equations in the class of Lie algebras over the field $\mathbb{C}$ of complex numbers. We use a concept of Hall basis for a free Lie algebra [4,5]. It should be noted that in the finite-dimensional case the effective algorithms are developed for the problem of construction of matrix representations for an algebra generated by a finite set of polynomials [6].

The method described below for restoring Lie algebras can be applied also to the infinite-dimensional case and has an algorithmic form as well. It is realized in the computer algebra system REDUCE [7].

The indicated problem is very actual, in particular, for Walhquist-Estabrook (W.-E.) method in the theory of nonlinear partial differential equations [8-10]. In fact, numerous successful attempts, including computer-aided [11], to realize W.-E. scheme in Lie algebras are based on *a priori* additional assumptions on a class of Lie algebra for the solution. It could lead to complete or partial loss of solutions unlike our approach which is directed to obtaining all the solutions of an initial algebraic system.

In ref. [12] by verification of Jacobi identities for basis elements taking into account initial (determining) algebraic relations one of particular problems of such a type has been solved. However, more general cases cause very tedious algebraic manipulations and therefore necessity of using a computer.

In the present paper we propose the algorithm for the whole computational process and describe its realization in REDUCE.

§2 Main notations and definitions.

Let L be Lie algebra over field $\mathbb{C}$. Then Lie product $[.,.]$ satisfies to the following axioms for any $u,v,w \in L$:

1) $[u,[v,w]] + [v,[w,u]] + [w,[u,v]] = 0$
2) $[u,u] = 0$, отсюда следует $[u,v] = - [v,u]$
3) $[(c_1u + c_2v),w] = c_1[u,w] + c_2[v.w]$, when $c_1,c_2 \in C$.

Generators $X(x_1,x_2,...,x_k)$ ("alphabetical letters") are a set of Lie algebra elements from which any other element is constructed by Lie product, addition and multiplication by complex numbers from $\mathbb{C}$.

The *basis* $R(X)$ of Lie algebra is a minimal set of elements such that any other element is their linear combination with the coefficients from $\mathbb{C}$.

Lie monomial ("word") $m(X)$ is any element from L constructed of the generators x_i by Lie products.

Lie polynomial $P(X,C)$ is a sum of Lie monomials with a set of coefficients $C(c_1,...c_k) \in \mathbb{C}$.

Determining relations ("phrases") are a set of Lie polynomial equalities of the form $P_i(X,C)=0$[1]. In the general case a set of determining relations leads to a system of algebraic equations in unknowns $c_i \in C, .x_k \in L$. When the values of c_i are strictly defined one can say about the construction of the quotient Lie algebra for a free Lie algebra over an ideal given by a set of generators and determining relations.

The algorithm proposed below for solving this problem is based on the concept of "Hall structure" [4] of a Lie monomial. By *base Hall family* of a free Lie algebra with alphabet X we shall mean any set R of Lie monomials with linear ordering $<$ which satisfies the following conditions:

[1] In discrete group theory the term "genetic code" is used [13] for the aggregate of generators and algebraically independent relations.

1. $X \in R$.
2. An element m of the form $[m_1,m_2] \in R$ iff
   (a) $m_1,m_2 \in R$,
   (b) $m_1 < m_2$,
   (c) if $m_2 = [m_3,m_4]$ then $m_1 > m_3$.
3. $[m_1,m_2] > m_1$.

For basis elements as Lie monomials of the above structure a computer can be effectively used to simplify algebraic expressions and to verify Jacobi identities.

Under quotation of free Lie algebra some of Lie monomials from R may remain free, i.e. they will play a role of basis for a quotient Lie algebra. In addition to it every such a monomial according to the above definition either is Lie product of a pair of other basis elements or a separate generator which can be treated as a pair as well. Therefore we shall call basis monomials free pairs. R^f will denote a set of such pairs. Correspondingly R^b will denote a set of pairs $[m_1,m_2]$, such that $m_1,m_2 \in R^f$, $m_1<m_2$ and $[m_1,m_2] \notin R^f$. These will be called bound ones. In the process of computation the bound pairs are expressed in terms of linear combinations of basis elements with coefficients from $\mathbb{C}$. Expressions of such a kind will be denoted phrases. Thus, $R^f \cup R^b$ is commutant of the desired Lie algebra together with generators.

For subsequent description it is useful a concept of weight $V(x_i)$ for a generator. It means that the certain natural number is assigned to each generator. The weight of Lie monomial is defined as sum of weights of its constituents (letters), for example:

if $V(x_1) = 1$, $V(x_3) = 2$,

then $V([x_1,[x_1,[[x_1,x_3]x_3]]]) = 3V(x_1) + 2V(x_3) = 3 + 4 = 7$.

§3 Statement of problem. Example

Let R_k be subset $R^f \cup R^b$, such that $[u,v] \in R_k$, if $u,v \in R^f$ and $V(u) + V(v) = k$ where k is a natural number. Then $R^f = \bigcup_k R^f_k$; $R^b = \bigcup_k R^b_k$; $R^f \cup R^b = \bigcup_k R_k$. Let us call R_k k-row. The question now arises how to construct R_{n+1} row by the given n rows, i.e. $\bigcup_{k=1}^{n} R_k$.

To make this question clear and for a better insight into the algorithm of §4 let us analyze one of the problems [12] in more detail.

The determining relations for that problem have the following form

$$[x_1,[x_2,x_1]] - c_1\,[x_2,x_1] + c_3x_2 = 0 \qquad (1)$$
$$[x_2,[x_2,x_1]] - [x_1,x_3] - c_4x_2 = 0 \qquad (2)$$
$$[x_2,x_3] - c_2[x_2,x_1] - c_5x_2 = 0. \qquad (3)$$

Here x_1,x_2,x_3 are generators, $c_i \in C$, $[\,,\,]$ - Lie product. There are five elements of basis $x_1,x_2,x_3,[x_2,x_1]$ and $[x_1,x_3]$ or $[x_2,[x_2,x_1]]$. It means that one of the last two Lie monomials can be chosen as an element of basis and another is expressed according to (2). Let $[x_2,[x_2,x_1]]$ be an element of our basis. Then introduce the ordering

$$x_2 < x_1 < [x_2,x_1] < x_3 < [x_2,[x_2,x_1]] < \dots$$

with weights: $V(x_1) = V(x_2) = 1$, $V(x_3) = 2$. Then $V([x_2,x_1]) = 2$, $V([x_2,[x_2,x_1]]) = 3$.

Now we can present the problem in the table form:

R_1	R_1^f	x_2 x_1
R_2	R_2^f	$[x_2,x_1]$ x_3
R_3	R_3^f	$[x_2,[x_2,x_1]]$
	R_3^b	$[x_1,[x_2,x_1]] = c_1[x_2,x_1] - c_3x_1$ (1)' $[x_2,x_3] = c_2[x_2,x_1] + c_5x_2$ (2)' $[x_1,x_3] = [x_2,[x_2,x_1]] - c_4x_1$ (3)'

In the given specific case arbitrariness in a choice of the ordering for initial elements of Hall basis (§2) is relatively small. Therefore the problem (1)-(3) is easily reduced to the form (1)'-(3)' to which our algorithm (§4) can be effectively applied. In the general case the optimal choice of ordering as well as in Groebner basis construction [2] is a difficult problem that calls for further investigation.

In our example we shall verifying a certain sequence of Jacobi identities for the basis element already available.

Calculation of R_4:

Jacobi identity for $x_2,x_1,[x_2,x_1]$ is:

$$[x_2,[x_1,[x_2,x_1]]] + [[x_2,x_1],[x_2,x_1]] + [[[x_2,x_1],x_2],x_1] = 0$$

Using skew-symmetry and bilinearity of Lie brackets we obtain from (1')

$$c_1[x_2,[x_2,x_1]] + [[[x_2,x_1],x_2],x_1] = 0. \qquad (4)$$

Transform the last monomial in eq. (4) to the Hall structure according to the chosen ordering and rewrite (4) to the form

$$[x_1,[x_2,[x_2,x_1]]] = -\,c_1[x_2,[x_2,x_1]]. \qquad (4)'$$

Thus, we receive the expression for the commutator of the basis elements x_1 and $[x_2,[x_2,x_1]]$ in terms of the basis element $[x_2,[x_2,x_1]]$.

Computation of the commutator of the elements x_3 and $[x_2,x_1]$ by verification of Jacobi identity for x_1,x_2,x_3 gives

$$[x_3,[x_2,x_1]] = -\,[x_2,[x_2,[x_2,x_1]]] + (c_1c_2 - c_5)[x_2,x_1] - c_2c_3x_2 .$$

We have looked over all possible Jacobi identities for basis elements of weight 4 and the result is as follows

R_4	R_4^f	$[x_2,[x_2,[x_2,x_1]]]$
	R_4^b	$[x_1,[x_2,[x_2,x_1]]] = -c_1[x_2,[x_2,x_1]]$
		$[x_3,[x_2,x_1]]=-[x_2,[x_2,[x_2,x_1]]]+(c_1c_2-c_5)[x_2,x_1]-c_2c_3x_2$

Computation R_6 with verification of Jacobi identities for the basis elements $x_1, [x_2,x_1]$ and $[x_2,[x_2,x_1]]$ relations gives:

$$c_3[x_2,[x_2,[x_2,x_1]]] = c_1c_4[x_2,[x_2,x_1]]\ /2\ . \qquad (5)$$

We call the relation of type (5) "key phrase" as far as at this moment our computational process gets a branch. Indeed, now we have three possibilities:

1) $c_3 \neq 0$.

In this case the monomial $[x_2,[x_2,[x_2,x_1]]]$ belonging earlier to R_4^f now should be considered as element of R_4^b and hence computation has to start at R_4 in view of eq. (5).

2) $c_3 = c_1 = 0$.

3) $c_3 = c_4 = 0$.

In these cases one has to repeat all computation from the very beginning with the assigned values for c_i.

In the general case similar calculations can lead to three different situations:

1. Finite-dimensional Lie algebra is obtained.

2. After calculating a sufficiently large volume, it is possible to deduce the structure of further rows R_k by induction and to write an infinite-dimensional Lie algebra in the recurrent form.

3. The number of basis elements is growing too fast that makes further analysis by induction very difficult. In this case one can introduce new determining relations, i.e. express separate basis elements in terms of linear combinations of low weight elements with arbitrary coefficients from $\mathbb{C}$. In our example we receive four distinguished solutions:

First solution.

All c_i are arbitrary constants.

Lie algebra basis: $x_1, x_2, x_3, [x_2,x_1]$. Let $[x_2,x_1]=x_4$.

Four-dimensional Lie algebra:

$$\begin{aligned}
&[x_2,x_1] = x_4,\\
&[x_2,x_3] = c_2x_4 + c_5x_2,\\
&[x_2,x_4] = 0,\\
&[x_1,x_3] = -c_4x_2,\\
&[x_1,x_4] = -c_3x_2,\\
&[x_3,x_4] = (c_1c_2-c_5)x_4 - c_2c_3x_2.
\end{aligned}$$

Second solution.

$c_4 = 0$ (the other c_i are arbitrary).

Lie algebra basis: $x_1, x_2, x_3, [x_2,x_1], [x_2,[x_2,x_1]]$,

Let $x_4 ; [x_2,x_1]$, $x_5 ; [x_2,[x_2,x_1]]$

Five-dimensional Lie algebra:

$$\begin{aligned}
&[x_2,x_1] = x_4,\\
&[x_1,x_3] = x_5,\\
&[x_2,x_4] = x_5,\\
&[x_2,x_3] = c_2x_4 + c_5x_2,\\
&[x_1,x_4] = c_1x_4 - c_3x_2,\\
&[x_1,x_5] = c_1x_5,\\
&[x_4,x_5] = 0,\\
&[x_3,x_4] = (\ c_1c_2 - c_5)x_4 - c_2c_3x_2,\\
&[x_3,x_5] = (\ c_1c_2 - 2c_5\ )x_5,\\
&[x_2,x_5] = 0.
\end{aligned}$$

Third solution.

$c_3 = c_4 = 0$ (the other c_i are arbitrary).

Lie algebra basis: $x_2, x_3,\ ad^k x_2(x_1),\ \{k\}_0^\infty$. Let $y_k = ad^k x_2(x_1)$.

Infinitely-dimensional Lie algebra:

$$\begin{aligned}
&[x_2,x_3] = c_1y_1 + c_5x_2,\\
&[x_2,y_k] = y_{k+1},
\end{aligned}$$

$$[y_0,y_n] = c_1 y_n,$$
$$[y_0,x_3] = y_2,$$
$$[y_n,x_3] = y_{n+2} + (nc_5 - c_1 c_2) y_n,$$
$$[y_p,y_n] = 0,$$
$$\{n\}_1^\infty,\ \{p\}_1^\infty.$$

Forth solution.

$c_1 = c_3 = c_5 = 0$ (the other c_i are arbitrary).

Lie algebra basis: $x_2, x_3,\ ad^k x_2(x_1),\ \{k\}_0^\infty$. Let $y_k = ad^k x_2(x_1)$

Infinitely-dimensional Lie algebra:

$$[x_2,y_k] = y_{k+1},$$
$$[y_1,\ x_3] = y_3,$$
$$[y_1,y_0] = 0,$$
$$[x_2,x_3] = c_2 y_1,$$
$$[y_0,x_3] = y_2 + c_4 x_2,$$
$$[y_p,x_3] = y_{p+2} + \frac{1}{2}(p - 2)c_2 c_4 y_{p-1},$$
$$[y_p,y_0] = \frac{1}{2}(p - 2)c_4 y_{p-1},$$
$$[y_p,y_n] = 0,$$
$$[y_1,y_p] = \frac{1}{2} c_4 y_p,$$
$$\{p\}_2^\infty,\ \{n\}_2^\infty .$$

§4 Algorithm description

The algorithm for solving the problem formulated earlier (§2) is based on the advent of the ordering by numeration of each word of Lie algebra L. Searching for a given word in whole list of words is performed by its number. To abbreviate we shall denote a word m by a number in parenthesis (i), $i \in N$, omitting the symbol m. Then any element $[m_i,m_j] \in R$ will have a form of the pair of number (i,j). Call this pair free, so as in set R it can be found by its number. A bounded pair from L has not its own number. Jacobi identities in L are fulfilled for any triple numbers.

To realize the ordering corresponding to the Hall ordering it is necessary to know also the contents of subwords of a number pair. Let

us describe the procedure of construction of the row with weight $n+1$ by given rows with weight $1,2,\dots,n$.

Step 1. Generate a word with weight $n+1$ from free words. It can be done, for example, by adding a word with weight $k+1$ to a word with weight $n-k$. Since the number of the first word is less than the number of the second one the rule (a) in definition of the Hall basis family (§3) is satisfied.

Step 2. Assign to pairs, obtained at the previous step numbers in accordance to general numbering of the basis elements. A number is assigned to a word if the two other rules (b) and (c) in the Hall basis definition are satisfied. If these rules are not satisfied one has to make substitutions of the subwords which follow from Jacobi identities (Lie differentiation):

$$[a,[b,c]] := [[a,b],c] - [[a,c],b]\ ,\ a,b,c \in N\ .$$

The appearing pairs $[a,b]$ and $[a,c]$ are replaced by the word having a number.

Step 3. Construct a word with weight $n+1$ from free pairs with weight $k+1$ and bound pairs with weight $n-k$. Add from the left words with a number from the initial data in such a way that the resulting weight is still equal to $n+1$. Differentiate the word with weight $n+1$ in the same way as it has been done at step 2. On the other hand the same word can be produced by adding a free pair, i.e a word with a number, to the right side of a bound pair. Everywhere substitute words with numbers for number pairs. The equation obtained is solved with respect to the word with a leading number. This equation is called "key phrase". Here the branching of the problem with different solutions may take place. If the leading word has a number less than $n+1$ the process is interrupted. It means that the problem must be solved under considering the new phrase that has been obtained. In other words it is necessary to start with new initial data. Other solutions leading essentially to new tasks will appear from analysis of the letter coefficients of a key phrase.

Step 4. The row with weight $n+1$ has been already constructed. All possible free pairs R_{n+1} have been obtained. There are no key phrases. Initial data are supplementing by the row with weight $n+1$ and then the next rows with weights $n+2,n+3,\dots$ are computed. The process of computation is finished when, beginning with some row, there are no free words or when we obtain data enough for restoring Lie algebra by induction.

Now we can present our algorithm in REDUCE-like form:

Comment: Input of initial data, contents of first i rows R_i and their lengths $l_i = |R_i|$, $i = 0,1,\dots,n$.;

Input : R_i , $i = 0,1,\dots,n$; $l_i = |R_i|$
Output : R_{i+1} , $l_{i+1} = |R_{i+1}|$

for $k := 1 : n$ *do* % start of steps 1 and 2 ;
    for $j1 := 1 : l_{k+1}$ *do* % generation of words, weight= $n+1$;
        for $j2 := 1 : l_{n-k}$ *do*
        if $j1 < j2$ *then* % verification of Hall conditions ;
    if $x_{j1} \in R^f_{k+1}$ *and*
        $x_{j2} \in R_{i-k}$ *then*
        if $x_{j1} > y_1$ *and* $y_2 = 0$ % $y_1 := car\ x_{j2}$;
        then % $y_2 := cadr\ x_{j2}$;
        $[x_{j1}, x_{j2}] := \sum_{s=1}^{i} l_s + N$ % assignment of ordinal numbers to ;
        % pars ;
        else
    $[x_{j1},[y_1,y_2]] := [[x_{j1},y_1],y_2]-[[x_{j1},y_2],y_1]$
end;

Comment: As a result of steps N and - we obtain all possible pairs with weight $n+1$;

for $k := 1 : n$ *do* % generation of words of weight= $n+1$;
    for $j1 := 1 : l_{k+1}$ *do*
        for $j2 := 1 : l_{n-k}$ *do*
        if $j1 < j2$ *then* % selection of pairs, satisfying ;
        if $x_{j1} \in R^f$ *and* % condition: first subword is basis ;
        $x_{j1} < y_1$ *and* % element, second subword does not ;
        $x_{j2} \in R^b$ % satisfy the Hall conditions ;
        then % $y_1 := car\ x_{j2}$;
        { $0 := [x_{j1},x_{j2}]-[x_{j1},[y_1,y_2]]$ }
end;

Comment: At step 3 we obtain equations in coefficients c_i and basis elements of L. Their analysis and solving are carried out in the block {...} . It is realized in the following way: the basic element with the leading word is selected and then it is expressed in terms of the remaining element of the phrase. It could not be done if all monomials in the equation contain coefficients c_i or they do not belong to the row with weight $n+1$.;

Output1: R_{n+1} , $l_{n+1} = |R_{n+1}|$ % contents of row with weight $n+1$;
0 and its length ;
Output2: "key phrase"

Comment: Further analysis of a key phrase and rows are done "by hand";

end of algorithm.

We have implemented the above algorithm in the REDUCE computer algebra system. The interactive regime is assumed. In the process of collection of the row contents and key phrases analysis a user applies that information to restore algebra L. For example, to make it for solution 3 (§3), the knowledge of twelve rows is necessary. The following table gives computing time on an IBM PC AT (12 Mhz):

row number	time in secs	comment
4 5 6	10 17 25	before the key phrase is produced
4 5 6 7 8 9 10 11 12	10 13 18 25 32 40 51 62 77	after substituting the conditions $c_3 = c_4 = 0$ from solution 3 in initial data

Our program includes two main procedures *timkk1* and *timkk2*, which form the ways of by-passing throughout the list of data: along the rows and inside them.

The procedure *timkk1* produces also adding an ordinal number and transposition of subwords, differentiating subwords by means of the procedure *dim2*.

The procedure *timkk2* calls for the procedure *lied* of subword transposition at the second by-passing throughout the list of data as well as the procedure *liep* for differentiating of the right-hand side of number pairs. The procedure *dfsu* serves to extract the equations and analyze the solutions to expose a key phrase.

To use the procedures *dfsu2*, *liep*, *lied* it is necessary to transform number pairs to some canonical form. That transformation is produced by the procedure *nsm*.

The program starts with loading of initial data and indicating of row weight to be obtained. For instance, after calling the procedure *tstart(n1,jobname)* the rows with weight less than *n1* will be loaded correspondingly to the problem with *jobname*. The present version of the program is oriented to the class of Lie algebras. In particular, it allows verification of Jacobi identities for a concrete Lie algebra and also construction of the Hall basis for any finite number of generators.

§5 Conclusion

The above algorithm does not pretend to be complete and effective for any problem of Lie algebra restoring by a given subset of generators and determining relations. The reasons for such a conclusion are following:

1) The key phrases obtained in the computation process in contrast to eq. (5) considered in §3 may lead to complicated systems of nonlinear algebraic equations in big number of scalar variables. To solve them the Groebner basis technique [1,2] has to be used. It is remarkable that one can apply to these problems the same computer algebra methods as to investigation of integrability of polynomial-nonlinear evolution equations with arbitrary scalar coefficients [14].

2) Our algorithm (§4) possesses only a few facilities given by the Hall basis definition (§2). The third point of that definition has been enhanced by the condition $[m_1,m_2] > m$, for $V(m_1)+V(m_2) > V(m)$. That concrete choice may restrict possibility of optimizing computational process.

3) Possible fast growing of the basis elements number mentioned above (§3) makes serious difficulties in analysis of the results of computation by induction.

In future we suppose to weaken the restrictions 1)-3) by means of extension of the algorithm with a special tool for analyzing and solving algebraic equations similar to those presented in [12].

Acknowledgments

We are grateful to V.A.Marchenko for his comments on an early draft of this paper and for many discussions. We thank also V.A.Andreev, A.E.Borovik, A.M.Vershik, V.G.Makhankov, V.K.Melnikov and M.V.Saveliev for useful remarks.

References

[1] Buchberger B. Groebner Bases: an Algorithmic Method in Polynomial Ideal Theory. In:(Bose N.K.,ed.) Recent Trends in Multidimensional System Theory, Reidel,1985.

[2] Proc. of the 1th Intern. Meeting "Computer and Commutative Algebra" (Genoa,Italy,May,1986). *J.Symb.Comp.* 6,2&3,1988.

[3] Kandri-Rody A.,Weisphenning V. Non-commutative Groebner Bases in Algebras of Solvable Type. *J.Symb.Comp.*9,1,1-26,1990.

[4] Hall M. A Basis for Free Lie Algebras and Higher Commutators in Free Groups. *Proc.Amer.Math.Soc.*,1,575-581,1950.
Hall M. The Theory of Groups. The Macmillan Co.,New York,1959.

[5] Bakturin Yu.A. Identities in Lie Algebras.Nauka.M.1985(in Russian).

[6] Labonte G. An Algorithm for the Construction of Matrix Representations for Finitely Presented Non-commutative Algebras. *J.Symb.Comp.* 9,1,27-38,1990.

[7] Hearn A.C. REDUCE User's Manual.Version 3.3. The Rand Corporation, Santa Monica,1987.

[8] Walhquist H.D.,Estabrook F.B. Prolongation Structures of Nonlinear Evolution Equations. *J.Math.Phys.* 16,1,1-7,1975.

[9] Andreev V.A. Connection Between "Pseudopotentials" and Inverse Scattering Problem. *Theor.Math.Phys.*,36,3,335-344,1978 (in Russian).

[10]Borovik A.E.,Robuk V.N. Linear Pseudopotentials and Conservation Laws for the Landau-Lifshits Equations. *Theor.Math.Phys.* 46,3, 371-381,1981(in Russian).

[11]Reiman A. Computer-Aided Closure of the Lie Algebra Associated with a Nonlinear PDE. Proc.of the 1989 MACSYMA Users Conference. MIT,Cambridge,Mass.,1979,385-396.

[12]Robuk V.N. On Classification of Second Order Evolution Equations. In: Operators in Functional Spaces and Problems in Theory of Functions. (V.A.Marchenko,ed.). Kiev,Naukova Dumka,1987,58-66 (in Russian).

[13]Coxeter H.S.M.,Moser W.O.J. Generators and Relations for Discrete Groups.Third Edition. Springer-Verlag,Berlin,1972.

[14]Gerdt V.P.,Khutornoy N.V.,Zharkov A.Yu. Solving Algebraic Systems which Arise as Necessary Integrability Conditions for Polynomial-Nonlinear Evolution Equations. JINR E5-90-48,Dubna,1990.

Received by Publishing Department
on November 6, 1990.