GEFAD / GUJGEF 34(1): 91-103 (2014)

A MULTI-DIMENSIONAL APPROACH IN MATHEMATICS TEACHER EDUCATION PROGRAMS: "COMPUTATIONS IN FREE AND FINITELY GENERATED LIE ALGEBRAS" EXAMPLE

MATEMATİK EĞİTİMİ PROGRAMLARINA ÇOK BOYUTLU BİR YAKLAŞIM: "LIE CEBİRİ" ÖRNEĞİ

Ebubekir TOPAK[1], Ela AYDIN[1], Orhan SÖNMEZ[2], Ahmet TEMİZYÜREK[3]

[1]*Department of Mathematics, Çukurova University.*

[2]*Department of Mathematics, Melikşah University.*

[3]*Department of Primary Education, Bozok University.*

e-posta: ahmet.temizyurek@bozok.edu.tr

Let L be a finitely generated Lie algebra and B be an arbitrary subalgebra of L. The maximal linearly independent set of the algebra L modulo the subalgebra B is called the modulo B basis of L. In this article we apply computer techniques to compute the modulo B basis of L using an algorithm given by Aydın in her PhD thesis.

***Keywords**: Mathematics Education, Lie Algebras, Content Knowledge.*

INTRODUCTION

Content knowledge is the knowledge that a teacher has about his or her specific domain area. According to Shulman (1986), "the content knowledge is the amount and organization of knowledge in the mind of teacher" (p.7).This knowledge is beyond what the teachers are going to teach. The reason for necessity of extra knowledge may be the teaching standards of National Council of Teachers of Mathematics (NCTM), which claims that a teacher should reason the reform ideas, construct challenges in a classroom, use new curriculum materials. Hence, such a teacher NCTM describes should know much more than s/he is going to teach. Furthermore, the research conducted by Goldhaber and Anthony (2003) asserts that the teacher with a higher degree of achievement in a specific subject has a positive effect on students' learning of this subject in certain settings. Hill, Rowen and Bell (2005) support this notion with the result of their research: "Students' mathematics achievement could be increased by improving teachers' mathematical content knowledge" (p. 398). As a result, it can be concluded that, besides the effects on teaching itself, the level of content knowledge of teachers affects the students' level of success.

The effects of content knowledge are usually measured by researchers as the level of success of students. According to Goldhaber and Anthony (2003), teachers with advanced degrees in specific subjects can have an impact on student learning in those subjects in certain settings. The reason for this effect can be the extra knowledge in content domain that enables teachers to challenge the ideas of students in the classroom or that enables teachers to get students reason mathematical thoughts raised during the lessons. Therefore, it can be concluded that the content knowledge makes a difference, for teachers, in the effectiveness of their teaching. As a result, it can be derived that the efforts to increase the quality of teacher education programs in terms of content knowledge is a grounded and meaningful aim in improving the education at all levels.

Narrowing down the issue, in this paper, it is aimed to present an example of deriving some computations, through algorithms and computer programs, in free and finitely generated Lie algebras. The main logic behind such an interdisciplinary perspective is that an effective collaborative learning outcome among computer science and discrete mathematics can be obtained through the use of technology, and in such an atmosphere students can monitor themselves (Berry, 1997). Therefore, this aim is consistent with the flow of the paper in which, through a computer program, the computations in free and finitely generated Lie algebras will be presented further.

There are many algorithms and computer programs that perform certain calculations associated with a Lie algebra structure. Andary(1997) has given an algorithm computing maximal Lyndon word (with respect to lexicographic order) of $Ly_{\alpha}(A)$ for every given α in N^k. Similarly, Cohen, and Graaf (1996) have described some basic algorithms for the structures of Lie algebras .

Let L be finitely generated Lie algebra. An algorithm for finding a base of L was given by Gerdt and Kornyak (1996). Gerdt also gives a program written in *C*-language calculating the base of L. Let L be a finitely generated Lie algebra and B be an arbitrary subalgebra of L. The maximal linearly independent set of the algebra L modulo the subalgebra B is called the modulo B base of L. Kryazhovskikh (1983) gives an algorithm that counting the base of L modulo the subalgebra B.

In literature, there are many computer programs computing the Hall basis of a free Lie algebra. For a finitely generated Lie algebra L and its ideal B, we introduce a new efficient algorithm and its computer program finding the modulo B base of L.

BASIC RESULTS AND NOTATIONS

Let F be a free Lie algebra generated by the set $X = \{x_1, x_2, \dots, x_n\}$ and H^X be the Hall set constructed on X. A Hall set constructed on X forms an additive base for the free Lie algebra F considered as a vector space.

For a given set X one can construct different Hall sets, each one determined by the order given to it. We now write in a more precise form:

$$H_1 = X,\ X \textit{ is well ordered},$$
$$H_2 = \{[x, y] : x, y \in X, x > y\},$$
$$\vdots$$
$$H_n = \{[[x, y]z] : x, y, z, [x, y] \in H_1 \cup H_2 \cup \cdots \cup H_{n-1}, x > y, y \le z, z < [x, y]\}.$$

We will use the notation $H^X = \bigcup_{n=1}^{\infty} H_n$ for the Hall base of L over X. The set H_n consists of those elements in H of length n. We will denote the length of an element x of H by $\ell(x)$.

Write an arbitrary element $f \in F$ as $f = \sum \gamma_i h_i$, $\gamma_i \in F$, $h_i \in H$. Assume that the monomials occuring with nonzero cofficients in this expression for f have exactly the next degree: $d_1 < d_2 < \cdots < d_k = d$. Then the subsume $\sum \gamma_i h_i = f_t$ of all the summands h_i 's of degree t is said to be the homogenous component of degree t of the element f. f_d is called leading part of f. The greatest monomial in the leading part of f is called leading monomial of f and it is denoted by $\ell d(f)$. For unmentioned terms and results, see (Bourbaki, 1975 & Reutenauer, 1993).

AN ALGORITHM THAT FINDING MODULO B BASE OF L

Let L be a Lie algebra over a field K with generators $x_1, x_2, \ldots, x_n$, and B its subalgebra. We use a method given by Aydın (1997) to find a base $\{a_i\}$ of the algebra L modulo the subalgebra B. For an element $l \in L$, we denote by $\bar{l}$ a linear combination of elements of $\{a_i\}$ such that $l - \bar{l} \in B$.

Theorem 1 *[7]The subalgebra B is generated by the set*

$$Y = \left\{ x_\alpha a_{j_1}^{n_1} \ldots a_{j_k}^{n_k} - \overline{x_\alpha a_{j_1}^{n_1} \ldots a_{j_k}^{n_k}} : \alpha \in I,\ j_1 < j_2 < \ldots < j_k, n_i \in \mathbb{Z}^+ \cup \{0\} \right\}$$

If L is a free Lie algebra, then the non zero elements of Y are free generators of the algebra B.

Henceforth, the arrangament of parentheses on a monomial $ab\ldots cd$ is assumed to be right normalize, i.e., $ab\ldots cd = ((\ldots(ab)\ldots)c)d$.

Theorem 2 *In the hypotheses of Theorem 1, Let* Υ *be an ideal of the algebra* L *with generators* $\{q_j\}$ *and let* $\{d_i\}$ *be an arbitrary base of the Lie algebra* L *modulo the subalgebra* B*. Then, the ideal* $\Upsilon \cap B$ *of the algebra* B *is generated by the elements*

$$q_j d_{i_1}^{n_1} \ldots d_{i_k}^{n_k} b_{s_1} \ldots b_{s_l} - \overline{q_j d_{i_1}^{n_1} \ldots d_{i_k}^{n_k} b_{s_1} \ldots b_{s_l}}$$

where $i_1 < i_2 < \ldots < i_k;\ n_i = 0,1,\ldots$, b_{s_i} are generators of the subalgebra B.

Definition 3 *Let* η *be a set of elements of a free Lie algebra* L^* *such that* $\ell d(f)$ *for each element* $f \in \eta$ *does not belong to the subalgebra generated by* $\ell d(g)$ *where* $g \in \eta$ *and* $g \neq f$ *. Then, the set* η *is said to be reduced.*

It has been shown by Shirshov (1953) that each subalgebra of a free Lie algebra possesses a reduced set of free generators. The base of a free Lie algebra L with free generators $\{x_i\}$ consists of regular words in the alphabet $\{x_i\}$(Shirshov, 1958).

Fix a base of the free Lie algebra L^* modulo its subalgebra B^*. We denote by B_n the subspace of elements of the subalgebra B^* whose degrees does not exceed n $(n = 1,2,\ldots)$. Let $S_0 = \varnothing$ and suppose that the sets S_i $(i < k)$ have been constructed as follows:

We denote by S_k the set of regular words (the base of the free Lie algebra L^*) of degree k which is maximal linearly independent modulo the linear hull of the set $B_k \cup \left(\bigcup_{i<k} S_i \right)$. Put $S = \bigcup_{i=0}^{\infty} S_i$. Obviously the set S is a base of the algebra L^* modulo its subalgebra B^*.

We represent an arbitrary Lie algebra L as a qutient algebra of a free Lie algebra L^*. Let B^* be the full preimage of a subalgebra B of the algebra L in the algebra L^*. The choice of a base in L^* modulo B^* induces the choice of a base in L modulo B.

Let us give the algorithm given by Aydın (1997, p. 2) that calculate a base in L modulo B as follows:

Let L be a Lie algebra over a field F and B is a subalgebra of L. We need the following inputs for the algorithm.

- $X = \{x_1, x_2, \ldots\}$ the generating set of the Lie algebra L.

- $\eta = \{y_1, y_2, \ldots\}$ the generating set of the subalgebra B.
- $R = \{r_1, r_2, \ldots\}$ the set of defining relations of L.

- $S = \{s_1, s_2, \ldots\}$ (if there exists) the set of defining relations of B.

- $P = \{p_1, p_2, \ldots\}$ (if they appear in the set of relations) the set of scaler parameters.

From all these sets we describe an algorithm determining the maximal linearly independent set $\{b_1, b_2, \ldots\}$ modulo the subalgebra B in L where L is a finitely presented Lie algebra as following:

Let η be a generating set of B. Let us partition the set η into the disjoint subsets η_d of elements of degree d. In this step, we enumerate each elements of η and classify these elements by its lengths. We consider the set of the elements of length k with respect to η by

$$\eta_k = \{y_i^k : \ell(y_i^k) = k, i \in I_k, k = 1,2,3,\ldots\}$$

where I_k is an index set. We denote the elements whose degree is 1 by $y_1^1, y_2^1, \ldots, y_k^1$ and the elements whose degree is 2 by $y_1^2, y_2^2, \ldots, y_s^2$, and so on. In this way, we construct the sets $\eta_1, \eta_2, \ldots$.

If the generating set η is not reduced then we must reduce it. If η is reduced there will be no change on η at the end of this step. Firstly, we take $\eta_0 = \varnothing$ and assume the sets $\eta_1, \eta_2 \ldots \eta_i$ are constructed. Let E_i be the subalgebra generated by the set $\bigcup_{j=0}^{\infty} \eta_j$. Let T_i be the vector space consists of the elements of the subalgebra B with degree $d \leq i+1$ and T_i' be the subspace of T_i with the elements whose degree is less than or equal to $i+1$. We can select the set η_{i+1} in T_i which is maximal linearly independent modulo the subspace $T_i^{'}$. If $T_i = T_i^{'}$, then $\eta_{i+1} = \varnothing.$ Thus, the set $\eta = \bigcup_{j=0}^{\infty} \eta_j$ is a reduced set.

Let take the vector space $B_N = span(\{\eta_0 \cup \eta_1 \cup \ldots \cup \eta_N\})$

In this last step, let us take $S_0 = \varnothing$ and suppose that the sets S_i have been already constructed for $i < k$. We denote by S_k the set of regular words of L of degree k which is maximal linearly independent modulo the set $span\left(\left\{B_k \cup \left(\bigcup_{i<k} S_i\right)\right\}\right)$. Let take $S = \bigcup_{i=0}^{\infty} S_i$ and determine the modulo B base of L . In this step, for each integer k , we choose the distinct elements of L of degree k which is not contained in the space spanned by $B_k \cup \left(\bigcup_{i<k} S_i\right)$ and we put these word(s) into the set S_k . We continue this process by increasing the integer $k.$ If, for any k , $S_k = \varnothing$ then we have to stop the algorithm. In this case, the base of L modulo the subalgebra B is finite. If, for all $i,$ $S_i = \varnothing$ then the base of L modulo the subalgebra B is infinite. As a result, all elements $b_i \in S_i$ constitute the base of L modulo B .

DESCRIPTION OF THE COMPUTER PROGRAM

Our program has some variables. We will list these variables below to make the description of the program clearer.

HallType: A record type list. Its fields are *index*, *I1*, *I2*, *L1* and *L2* that used to record the index of this element among the elements with the same length, the index of first component of this element, the index of second component of this element, the length of first component of this element and the length of second component of this element, respectively.

AllHall: A two-dimensional array of elements of HallType. *AllHall[i,j]* records the Hall element where i represents the length of this Hall element and j represents the index of this Hall element among the elements with the same length. Particularly, we record the number of Hall elements of length *n* in index field of *AllHall[n,0]*.

IsHallElement: A boolean function that determines whether an element with its given first and second components is a Hall element or not.

HallWord: A procedure that produces a Hall element with Lie brackets from a given element of *HallType*.

IsElementOfL: A boolean function that determines whether a given element is in L according to R or not.

IsInB: A boolean function that determines whether a given element is in B or not.

MaxLenB: For given parameter k, this function gives the length of the longest element of X -length among the elements with the length k in B.

Following is the main subroutine of our program:

```
procedure ComputeModBase();
var
 BaseElems: array [1..10000] of string;
 NumbOfBaseElems: longword;
 i,j,k,m,n: longword;
 AreWeAlreadyHave: boolean;
 ElemG: shortstring;
 S: boolean;
begin
```

$modBofL = \varnothing$

```
for k:=1 to 10 do
begin
 S:=false;
 for i:=1 to MaxLenB(k) do
 begin
  for j:=1 to AllHall[i,0].index do
  begin
   ElemG:='' ;
   ElemG:=HallWord(AllHall[i,j]);
   if IsInB(ElemG)=false then
   begin
    AreWeAlreadyHave:=false;
    for m:=1 to NumbOfBaseElems do
    begin
     if BaseElems[m]=ElemG then
     AreWeAlreadyHave:=true;
    end;
    if AreWeAlreadyHave=false then
    begin
     S:=true;
     NumbOfBaseElems:=NumbOfBaseElems+1;
     BaseElems[NumbOfBaseElems]:=ElemG;
    end;
   end;
  end;
  if S=false then
  begin
   for n:=1 to NumbOfBaseElems do
```

```
      begin
       modBofL = modBofL ∪ {BaseElems[n]}
      end;
      exit( modBofL );
     end;
    end;
   end;
  end;
```

Note that we are looking around the elements of length at most 10 of B since B is infinite. Also we limit the number of elements of the base we will construct with 10000.

Let us consider the following examples to demonstrate the results of the computer program.

Example 4 *Let*

$$L = \langle x, y, z \mid [y,[y,z]] = [x,[x,[x,y]]] = [x,[x,z]] = 0 \rangle$$

be a Lie algebra and

$$B = \langle y, z, [x,y], [x,[x,y]], [y,[x,z]], [z,[x,z]] \rangle$$

be a subalgebra of L.

When we enter these informations to our computer program as inputs we get a set $\{x,[x,z]\}$ as an output. So, we can say that the set $\{x,[x,z]\}$ is a base of the algebra L modulo the subalgebra B.

Example 5 *Let*

$$L = \langle x, y \mid [x,[x,y]] = [y,[y,[x,y]]] = 0 \rangle$$

be a Lie algebra and

$$B = \langle y, [x,y] \rangle$$

be a subalgebra of L.

When we enter these informations to our computer program as inputs we get a set $\{x\}$ as an output. So, we can say that the set $\{x\}$ is a base of the algebra L modulo the subalgebra B.

Through the computer program designed for computations for Lie algebras, the algorithms can be derived more efficiently. Such a multi-diciplinary content can be presented as an elective course in mathematics teacher education programs to add multi-dimensional perspective by letting prospective teachers to deal with the content beyond the scope they are going to teach. As a result, parallel with the points in the literature, it is believed that this can be one of the ways to increase the quality of teacher education programs.

REFERENCES

Andary, P. (1997). Finely homogeneous computations in free Lie algebras. *Discrete Math. Theor. Comput. Sci*, *1*(1), 101–114.

Aydın, E. (1997). *Subalgbras of Lie Algebras of Finite Codimension*, PhD Thesis, Çukurova University.

Berry, J. (1997). Improving discrete mathematics and algorithms curricula with LINK. ITICSE’ 97 2nd Conference on Integrating Technology into Computer Science Education, 14-20. ACM: New York.

Bourbaki N. (1975). *Lie groups and lie algebras, Part II.*, Addison-Wesley.

Cohen, A. M., & de Graaf, W. A. (1996). Lie algebraic computation. *Comput. Phys. Comm. 97*(1-2), 53–62.

Gerdt, V. P., & Kornyak, V. V. (1996). Construction of finitely presented Lie algebras and superalgebras. *J. Symbolic Comput*, *21*(3), 337–349.

de Graaf, W. A. (2000). *Lie Algebras: Theory and Algorithms*, North Holland.

Goldhaber, D., & Anthony, E. (2003). *Indicators of Teacher Quality*. Retrieved from ERIC database. (ED478408).

Hill C., H., Rowon, B., & Ball D., L. (2005). The effects of teachers mathematical knowledge for teaching on student achievement. *American Educational Research Journal*, *42*(2), 371-406.

Kryazhovskikh, G. V. (1983). Generating and defining relations of subalgebras of Lie algebras. *Sibirsk. Mat. Zh. 24*(6), 80–86.

Reutenauer, C. (1983). *Free lie algebras*, Oxford University Press.

Shirshov, A. I. (1953). Subalgebras of free Lie algebras. *Mat. Sbornik N. S., 33*(75), 441–452.

Shirshov, A. I. (1958). On free lie rings. *Mat. Sbornik N. S.*, *45*(87), 113–122.

Shulman, L., S. (1986). Those who understand knowledge growth in teaching. *Educational Researcher*, *15*(2), 4-14.