

# Minimal Faithful Upper-Triangular Matrix Representations for Solvable Lie Algebras

Manuel Ceballos<sup>†</sup>, Juan Núñez<sup>†</sup> and Ángel F. Tenorio<sup>‡1</sup>

<sup>†</sup>Departamento de Geometría y Topología.  
Facultad de Matemáticas. Universidad de Sevilla.  
Aptdo. 1160. 41080-Seville (Spain).  
Email: {mceballos,jnvaldes}@us.es

<sup>‡</sup>Dpto. de Economía, Métodos Cuantitativos e H.<sup>a</sup> Económica.  
Escuela Politécnica Superior. Universidad Pablo de Olavide.  
Ctra. Utrera km. 1, 41013-Seville (Spain).  
Email: aftenorio@upo.es

**Abstract.** A well-known result on Lie Theory states that every finite-dimensional complex solvable Lie algebra can be represented as a matrix Lie algebra, with upper-triangular square matrices as elements. However, this result does not specify which is the minimal order of the matrices involved in such representations. Hence, the main goal of this paper is to revisit and implement a method to compute both that minimal order and a matrix representative for a given solvable Lie algebra. As application of this procedure, we compute representatives for each solvable Lie algebra with dimension less than 6.

**Key words and phrases:** solvable Lie algebra, faithful upper-triangular matrix representation, algorithm.

**2000 Mathematics Subject Classification:** 17B30, 17B05, 17-08, 68W30, 68W05.

## 1 Introduction

Representation Theory of Lie algebras can be allowed for the classification of Lie algebras and groups, which has broad applications to the analysis of continuous symmetries in Mathematics and Physics. More concretely, in Mathematics, the classification of Lie groups reveals symmetries in differential equations. With respect to Physics, representation theory yields natural connections between representation of Lie algebras and the properties of elementary particles.

Ado's Theorem states that given a finite-dimensional complex Lie algebra  $\mathfrak{g}$ , there exists a matrix algebra isomorphic to  $\mathfrak{g}$  (see [1] for the classical proof and [2] for a very short alternative). In this way, every finite-dimensional complex Lie algebra can be represented

---

<sup>1</sup>Corresponding author. Phone: +34-954349354. Fax: +34-954349339

as a Lie subalgebra of the complex general linear algebra  $\mathfrak{gl}(n; \mathbb{C})$ , of complex  $n \times n$  matrices, for some  $n \in \mathbb{N}$ .

This paper focuses on Lie algebra  $\mathfrak{h}_n$ , of  $n \times n$  upper-triangular matrices. It is well known that every finite-dimensional solvable Lie algebra is isomorphic to a subalgebra of  $\mathfrak{h}_n$ , for some  $n \in \mathbb{N}$  (see [3, Proposition 3.7.3]). Therefore, the following interesting question arises for a given finite-dimensional solvable Lie algebra  $\mathfrak{g}$ : determining the minimal  $n \in \mathbb{N}$  such that  $\mathfrak{h}_n$  contains  $\mathfrak{g}$  as a Lie subalgebra; i.e. obtaining the minimal faithful representation of  $\mathfrak{g}$  by using  $n \times n$  upper-triangular matrices.

Several authors have studied the minimal dimension  $\mu(\mathfrak{g})$  to represent a given Lie algebra  $\mathfrak{g}$  (see Burde [4], for instance). However, most of them have considered faithful  $\mathfrak{g}$ -modules instead of the particular subclass consisting of Lie algebras  $\mathfrak{h}_n$ . Therefore, the value of  $\mu(\mathfrak{g})$  is less than or equal to the dimension to be computed in this paper. Regarding this matter, matrix representations were computed by Ghanam et al. [5] for low-dimensional nilpotent Lie algebras, but not studying the minimality and giving some non-minimal representations.

The interest in these faithful representations is motivated, among other issues, by problems from Geometry and Topology. For example, Milnor [6] and Auslander [7, 8] studied generalizations of crystallographic groups in relation with this minimal value for matrix representations. Another motivation is based on the following result: Lie algebra  $\mathfrak{g}$  of a given Lie group  $G$  admitting a left-invariant affine structure satisfies that its minimal dimension of faithful representations is  $\mu(\mathfrak{g}) \leq n + 1$ , where  $n$  is the dimension of  $\mathfrak{g}$ .

Several papers throughout the literature deal with matrix representation of these solvable Lie algebras. For example, Benjumea et al. [9] introduced an algorithmic procedure which explicitly computed a representative of minimal faithful unitriangular matrix representations for a given nilpotent Lie algebra and its associated Lie group. Subsequently, the complete list of minimal faithful unitriangular matrix representations was given by Benjumea et al. [10] for nilpotent Lie algebras of dimension less than 6. Finally, the matrix representation of filiform Lie algebras of dimension less than 9 was computed in [11]. Additionally, Núñez and Tenorio [12] continued with this research and gave the outlines of an algorithmic procedure to compute explicitly representatives of the minimal faithful matrix representation for solvable Lie algebras by using Lie algebras  $\mathfrak{h}_n$ , giving some examples of application by hand. This procedure adapted that given in Benjumea et al. [9], but neither the algorithm was completely debugged nor implementations were carried out and run.

The main goal of the current paper is to advance in the above-mentioned research by debugging and implementing the algorithm sketched in [12] in order to automate the computation of minimal faithful matrix representations for a given solvable Lie algebra starting from its law. As application, we have also computed representations for each solvable Lie algebra of dimension less than 6 as well as for others of higher dimension. To do so, we have used the classifications given by Mubarakzyanov and Turkowski (see [13, 14, 15, 16]).

This paper is structured as follows: Section 2 reviews some well-known results on Lie Theory to be applied later. Thereupon, Section 3 revisits the algorithmic procedure sketched in [12] to compute minimal faithful representations for solvable Lie algebras by using upper-triangular matrices, incorporating a formulation of the algorithm which can be dealt computationally with implementation in MAPLE 12. To shorten the paper length, the computational method is only explicitly applied to two algebras in Section 4. Just afterwards, Section 5 gives an explicit list with representatives of minimal faithful matrix representations for solvable Lie algebras of dimension less than 6.

## 2 Preliminaries

For an overall review on Lie algebras, the reader can consult [3]. In the present section, we only recall some definitions and results about Invariant Theory and Lie algebras to be applied later. Throughout this article, we only consider finite-dimensional Lie algebras over the complex number field  $\mathbb{C}$ .

Given a Lie algebra  $\mathfrak{g}$ , its *derived series* is defined as follows

$$\mathcal{C}_1(\mathfrak{g}) = \mathfrak{g}, \mathcal{C}_2(\mathfrak{g}) = [\mathfrak{g}, \mathfrak{g}], \mathcal{C}_3(\mathfrak{g}) = [\mathcal{C}_2(\mathfrak{g}), \mathcal{C}_2(\mathfrak{g})], \dots, \mathcal{C}_k(\mathfrak{g}) = [\mathcal{C}_{k-1}(\mathfrak{g}), \mathcal{C}_{k-1}(\mathfrak{g})], \dots \quad (1)$$

Additionally, the Lie algebra  $\mathfrak{g}$  is said to be *solvable* if there exists a natural integer  $m$  such that  $\mathcal{C}_m(\mathfrak{g}) \equiv 0$ . The solvability index of  $\mathfrak{g}$  is precisely the value of  $m \in \mathbb{N}$  such that  $\mathcal{C}_m(\mathfrak{g}) = 0$  and  $\mathcal{C}_{m-1}(\mathfrak{g}) \neq 0$ .

The relation between the derived series of a given Lie algebra  $\mathfrak{g}$  and that of a Lie subalgebras is given as follows

**Proposition 1.** *If  $\mathfrak{h}$  is a Lie subalgebra of a given Lie algebra  $\mathfrak{g}$ , then  $\mathcal{C}_k(\mathfrak{h}) \subseteq \mathcal{C}_k(\mathfrak{g})$ , for all  $k \in \mathbb{N}$ .*

Given  $n \in \mathbb{N}$ , the complex solvable Lie algebra  $\mathfrak{h}_n$  consists of  $n \times n$  upper-triangular matrices; i.e. its vectors are expressed as

$$h_n(x_{r,s}) = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ 0 & x_{22} & \cdots & x_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & x_{nn} \end{pmatrix}, \quad \text{with } x_{r,s} \in \mathbb{C}, \text{ for } 1 \leq r \leq s \leq n. \quad (2)$$

Lie algebra  $\mathfrak{h}_n$  has a basis  $\mathcal{B}_n$  consisting of vectors  $X_{i,j} = h_n(x_{r,s})$  with  $1 \leq i \leq j \leq n$  and such that

$$x_{r,s} = \begin{cases} 1, & \text{if } (r,s) = (i,j), \\ 0, & \text{if } (r,s) \neq (i,j). \end{cases} \quad (3)$$

The dimension of  $\mathfrak{h}_n$  is  $\frac{n(n+1)}{2}$  and the nonzero brackets with respect to basis  $\mathcal{B}_n$  are

$$[X_{i,j}, X_{j,k}] = X_{i,k}, \quad \forall 1 \leq i < j < k \leq n; \quad (4)$$

$$[X_{i,i}, X_{i,j}] = X_{i,j}, \quad 1 \leq i < j \leq n; \quad (5)$$

$$[X_{k,i}, X_{i,i}] = X_{k,i}, \quad \forall k \leq i \leq n. \quad (6)$$

### 3 Computing Minimal Matrix Representations

This section continues the work started in [12] and introduces an algorithmic method to compute minimal matrix representations of solvable Lie algebras in such a way this can be dealt with computer algebra. After explaining step by step the algorithm, this is implemented in Maple 12 and applied to several examples.

Given a Lie algebra  $\mathfrak{g}$ , a *representation* of  $\mathfrak{g}$  in  $\mathbb{C}^n$  is a homomorphism of Lie algebras  $\phi : \mathfrak{g} \rightarrow \mathfrak{gl}(\mathbb{C}, n)$ . Then  $n \in \mathbb{N}$  is called the *dimension* of this representation. Ado's theorem states that every finite-dimensional Lie algebra over a field of characteristic zero (as in the case of  $\mathbb{C}$ ) has a linear injective representation on a finite-dimensional vector space; that is, a *faithful representation*.

Usually, representations are defined as  $\mathfrak{g}$ -*modules*, consisting of homomorphisms of Lie algebras from  $\mathfrak{g}$  to Lie algebra  $\mathfrak{gl}(V)$  of endomorphisms over an arbitrary  $n$ -dimensional vector space  $V$  (like in [18]).

Regarding minimal representations of Lie algebras, Burde [4] introduced the invariant  $\mu(\mathfrak{g})$  for an arbitrary Lie algebra  $\mathfrak{g}$

$$\mu(\mathfrak{g}) = \min\{\dim(M) \mid M \text{ is a faithful } \mathfrak{g}\text{-module}\}.$$

In this section, matrix faithful representations of solvable Lie algebras are studied. Moreover, we are interested in minimal faithful matrix representations with a particular restriction: the representation is contained in  $\mathfrak{h}_n$  for some  $n \in \mathbb{N}$ . In this way, given a solvable Lie algebra  $\mathfrak{g}$ , we want to compute the minimal value  $n$  such that  $\mathfrak{h}_n$  contains a Lie subalgebra isomorphic to  $\mathfrak{g}$ . This value is also an invariant of  $\mathfrak{g}$  and its expression is given by

$$\bar{\mu}(\mathfrak{g}) = \min\{n \in \mathbb{N} \mid \exists \text{ subalgebra of } \mathfrak{h}_n \text{ isomorphic to } \mathfrak{g}\}.$$

In general, invariants  $\mu(\mathfrak{g})$  and  $\bar{\mu}(\mathfrak{g})$  can be different from each other.

Next, we show the algorithmic method to compute minimal faithful matrix representations for those algebras by using Lie algebras  $\mathfrak{h}_n$ . The minimality must be understood in the following sense: There exists a faithful matrix representation of  $\mathfrak{g}$  in  $\mathfrak{h}_n$ , but not in  $\mathfrak{h}_{n-1}$ .

To do so, we give a step-by-step explanation of the algorithm used to determine these minimal representations for a given solvable Lie algebra  $\mathfrak{g}$  of dimension  $n$ .

1. According to Proposition 1, we compute the first natural integer  $k$  such that the derived series of  $\mathfrak{h}_k$  is compatible with that associated with  $\mathfrak{g}$ ; i. e., the first  $k$  verifying  $\mathcal{C}_i(\mathfrak{g}) \subseteq \mathcal{C}_i(\mathfrak{h}_k)$ , for all  $i \in \mathbb{N}$ .
2. We search a Lie subalgebra of  $\mathfrak{h}_k$  isomorphic to  $\mathfrak{g}$ , with  $k$  as low as possible. To do so, the vectors in the basis  $\{e_i\}_{i=1}^n$  of  $\mathfrak{g}$  are expressed as the following linear combinations of basis  $\mathcal{B}_k$

$$e_h = \sum_{1 \leq i \leq j \leq k} \lambda_{i,j}^h X_{i,j}, \quad \text{for } 1 \leq h \leq n. \quad (7)$$

3. Bracket  $[e_i, e_j]$  is computed for  $1 \leq i \leq j \leq n$ . When imposing the law of  $\mathfrak{g}$ , a system of non-linear equations is obtained by comparing coordinate to coordinate with respect to basis  $\mathcal{B}_k$ .
4. We solve the system of equations and a solution of the system provides us one of the representations searched for Lie algebra  $\mathfrak{g}$  if the solution corresponds to a set of vectors being linearly independent. When no solution is obtained, Lie algebra  $\mathfrak{g}$  cannot be represented as a Lie subalgebra of  $\mathfrak{h}_k$ . In this case, we go back to Step 2 and repeat each step with Lie algebra  $\mathfrak{h}_{k+1}$ .

The representation obtained for Lie algebra  $\mathfrak{g}$  is minimal because we start with  $k = 1$  and  $k$  increases one unit when no representation can be obtained from  $\mathfrak{h}_k$ .

Obviously, the set of solutions of the system in the last step depends on the Lie algebra which the algorithm is applied to. As an example of application, we have minimally represented solvable Lie algebras with dimension less than 6 in Section 4. The set of solutions in Step 4 has been computed with the command `solve` in the symbolic computation package Maple 12. This command works efficiently with polynomial equations, receives as inputs the list of equations and the list of variables, and returns as output the algebraic expression of the set of solutions.

Furthermore, in order to compute a particular solution of the previous system, we have searched one having as many coefficients  $\lambda_{i,j}^h$  being equal to 0 as possible. In this way, a coefficient is assumed to be equal zero when it does not appear in the relations obtained by the equations. This will be a natural representative of the Lie algebra  $\mathfrak{g}$ .

### 3.1 Implementation

Next, we show the implementation of the different routines in order to apply the previous method. They have been written using the symbolic computation package MAPLE 12, loading the libraries `DifferentialGeometry`, `LieAlgebras` to activate commands related to Lie algebras.

First, the routine `law_h` is implemented to compute the law of the solvable Lie algebra  $\mathfrak{h}_n$ . This routine receives as input the value of  $n$  and returns the list of brackets

expressing the law of  $\mathfrak{h}_n$  with respect to the basis  $\{e_1, e_2, \dots, e_{\frac{n(n+1)}{2}}\}$ , which corresponds to  $\{X_{1,1}, \dots, X_{1,n}, X_{2,2}, \dots, X_{2,n}, \dots, X_{n,n}\}$ . For the implementation, a list B saves the basis of  $\mathfrak{h}_n$  and S keeps all the (non-zero) brackets involved in the law. To carry out the computations, three different loops are programmed to find and save the three different types of non-zero brackets in Eqs. (4)–(6). Finally, the law of the algebra is saved in the variable Ext1 to be loaded in a later routine.

```

> law_h:=proc(n)
> local B, S;
> B:=[]; S:=[];
> for i from 1 to n do                                     (*Constructing the basis*)
>   for j from i to n do
>     B:=[op(B),X[i,j]];
>   end do;
> end do;
> for i from 1 to n-1 do                                  (*Finding brackets in Eq. (5)*)
>   for j from i+1 to n do
>     S:=[op(S),[X[i,i],X[i,j]]=X[i,j]];
>   end do;
> end do;
> for i from 1 to n-1 do                                  (*Finding brackets in Eq. (6)*)
>   for j from i+1 to n do
>     S:=[op(S),[X[i,j],X[j,j]]=X[i,j]];
>   end do;
> end do;
> for i from 1 to n-2 do                                  (*Finding brackets in Eq. (4)*)
>   for j from i+1 to n-1 do
>     for k from j+1 to n do
>       S:=[op(S),[X[i,j],X[j,k]]=X[i,k]];
>     end do;
>   end do;
> end do;
> return LieAlgebraData(S,B,Ext1, "LieAlgebraData");     (*Defining the algebra*)
> end proc;

```

Next, the routine `DerivedSeries_h` receives as input the value of  $n$  and computes a list with the dimension of each term in the derived series of  $\mathfrak{h}_n$ . Let us note that we have to distinguish two different cases; being the first when  $n$  is very low. Otherwise, we only have to consider powers of 2 as it can be proved by a straightforward inductive reasoning.

```

> DerivedSeries_h:=proc(n)
> local L;
> L:=[n*(n+1)/2];
> if n<4 then
>   for i from 1 to n do L:=[op(L),(n-i)*(n-i+1)/2]; end do;
> else
>   for i from 0 by 1 while 2^i<n do L:=[op(L),(n-2^i)*(n-2^i +1)/2]; end do;
> end if;
> if member(0,L)=false then L:=[op(L),0]; end if;
> return L;
> end proc;

```

Now, we define solvable Lie algebra  $\mathfrak{g}$  according to the following notation

```
> L:= _DG(["LieAlgebra", g, [n]], [A]);
> DGsetup(L);
```

where  $n$  is the dimension of  $\mathfrak{g}$  (a value inserted by the user) and  $A$  is a list containing information about the structure constants of the law of  $\mathfrak{g}$ . Elements in  $A$  must be of the form  $[[i, j, k], cij_k]$  where  $cij_k$  is the structure constant  $\lambda_{i,j}^k$  corresponding to the coefficient of  $e_k$  in the bracket  $[e_i, e_j]$ . Once these data are loaded, we can operate over Lie algebra  $\mathfrak{g}$ .

The next routine is called `DerivedSeries` and computes a list with the dimension of each ideal in the derived series of  $\mathfrak{g}$ .

```
g > DerivedSeries:=proc()
g > local k;
g > C[1]:=[seq(e||i, i=1..n)]; C[2]:=DerivedAlgebra();          (*Initiating derived series*)
g > if C[2]=[] then return "Abelian Lie algebra"; end if;      (*Testing abelian Lie algebras*)
g > for i from 3 by 1 while C[i]<> [] do                          (*Constructing derived series*)
g >   C[i]:=DerivedAlgebra(C[i-1]);
g >   if C[i]=[] then return [seq(nops(C[j]),j=1..i)]; end if;
g > end do;
g > end proc;
```

Next, the solvability index of  $\mathfrak{g}$  is saved in the variable called `index_g` and all the ideals of the derived series are also defined. To do so, we execute the following sentences

```
g > assign(index_g,nops(DerivedSeries()));
g > C[1](g):=[seq(e||i, i=1..n)];
g > C[2](g):=DerivedAlgebra();
g > for i from 3 to index_g do C[i](g):=DerivedAlgebra(C[i-1](g)); end do;
```

The routine `DimRepresentation` computes the minimal dimension for a matrix representation of  $\mathfrak{g}$  by using Lie algebras  $\mathfrak{h}_n$ . To implement this routine, we compare the dimension sequence of the derived series of both  $\mathfrak{g}$  and  $\mathfrak{h}_n$  by using the routines `DerivedSeries_h` and `DerivedSeries`. The output is the minimal  $k \in \mathbb{N}$  such that Step 1 is verified.

```
g > DimRepresentation:=proc()
g > L:=DerivedSeries();
g > k:=0; d:=nops(L);
g > if d <= 4 then          (*Initiating dimension of representation*)
g >   n:=d-1; else n:=d;
g > end if;
g > while k=0 do
g >   M:=DerivedSeries_h(n);
g >   for i from 1 to nops(L) do
g >     if L[i]<=M[i] then k:=k; (*Comparing dimension sequences of derived algebras*)
g >     else k:=k+1;
g >     end if;
g >   end do;
g >   if k>0 then k:=0;      (*Checking compatibility between dimension sequences*)
g >   else return n;        (*Returning dimension of representation*)
g >   end if;
g >   n:=n+1;
g > end do;
g > end proc;
```

After completing Step 1 in the algorithm, we need to express all the vectors in the basis of  $\mathfrak{g}$  as a linear combination of basis  $\mathcal{B}_k$  of  $\mathfrak{h}_k$ , where  $k$  is the output of the routine `DimRepresentation`. Therefore, we start loading Lie algebra  $\mathfrak{h}_k$  with the sentence

```
g > DGsetup(law_h(k), [x], [a])
```

In this sentence, `[x]` is used to denote the basis vectors in  $\mathcal{B}_k$  as  $\{x_i\}_{i=1}^{\frac{k(k+1)}{2}}$  instead of  $\{e_i\}_{i=1}^{\frac{k(k+1)}{2}}$ , since we need different notations for the bases of both Lie algebras  $\mathfrak{g}$  and  $\mathfrak{h}_k$ . The notation `[a]` corresponds to the list of structure constants defining the law of Lie algebra  $\mathfrak{h}_n$  and expressed as in (4), (5) and (6). From this point, we can also work over the Lie algebra  $\mathfrak{h}_k$ , named `L1` by the package. Next, as we did with Lie algebra  $\mathfrak{g}$ , we define all the ideals of the derived series of  $\mathfrak{h}_k$  as follows

```
L1 > C[1](h):=[seq(x||i, i=1..k*(k+1)/2)];
L1 > C[2](h):=DerivedAlgebra();
L1 > for i from 3 to nops(DerivedSeries_h(k)) do C[i](h):=DerivedAlgebra(C[i-1](h)); end do;
```

Next, we express all the vectors  $\{e_i\}_{i=1}^n$  from the basis of  $\mathfrak{g}$  as a linear combination of basis  $\mathcal{B}_k = \{x_i\}_{i=1}^{\frac{k(k+1)}{2}}$  of  $\mathfrak{h}_k$ . To do so, we first implement a subroutine called `listposi`, which computes the position of an element within a list. Then, we also implement the routine `expr`. The latter uses the derived series of both  $\mathfrak{g}$  and  $\mathfrak{h}_k$  and, by applying Proposition 1, returns as output two lists: the first contains all the expressions according to the second step of the method, that is, Equation (7); and in the second, the conditions over the coefficients so that non-zero vectors are considered.

```
L1 > listposi:=proc(a,L)
L1 > for i from 1 to nops(L) do
L1 >   if a=L[i] then return i; end if;
L1 > end do;
L1 > end proc;
```

Let us note that in order to implement routine `expr` we have defined several clusters comparing the derived series of  $\mathfrak{g}$  and  $\mathfrak{h}_k$ . With this routine, we express the basis vectors of  $\mathfrak{g}$  as a linear combination of the basis vectors from  $\mathfrak{h}_k$ .

```
L1 > expr:=proc()
L1 > L:=[];M:=[];
L1 > for i from 1 to index_g - 1 do
L1 >   for j from 1 to nops(C[i](g)) do
L1 >     if member(C[i](g)[j],C[i+1](g))=false then
L1 >       if C[i+1](g)<>[] then
L1 >         L:=[op(L),C[i](g)[j]=sum(a[j,k]*C[i](h)[k],k=1..nops(C[i](h)))];
L1 >         M:=[op(M),sum(a[j,k]^2,k=1..nops(C[i](h)))<>0];
L1 >       end if;
L1 >     else member(C[i](g)[j],C[1](g),'p');
L1 >       N:=[seq(listposi(C[i+1](h)[k],C[1](h)),k=1..nops(C[i+1](h)))];
L1 >       L:=[op(L),C[i](g)[j]=sum(a[p,N[k]]*C[i+1](h)[k],k=1..nops(C[i+1](h)))];
L1 >       M:=[op(M),sum(a[p,N[k]]^2,k=1..nops(C[i+1](h)))<>0];
```



```

L1 > end if;
L1 > end do;
L1 > end do;
L1 > return L,M;
L1 > end proc:

```

After expressing the basis of  $\mathfrak{g}$  with respect to basis  $\mathcal{B}_k$  of  $\mathfrak{h}_k$ , we impose the law of both Lie algebras to the previous expressions. In this way, we implement the routine `Listeq`, which returns two lists: the first one containing the elements to be equal to zero; and the second one with the conditions to assure the linear independence of the basis.

```

L1 > Listeq:=proc()
L1 > R:=[];
L1 > for i from 1 to nops(expr()[1])-1 do
L1 > for j from i+1 to nops(expr()[1]) do
L1 > if BracketOfSubspaces([lhs(expr()[1][i])],[lhs(expr()[1][j])])=[] then
L1 > R:=[op(R),op(BracketOfSubspaces([rhs(expr()[1][i])],[rhs(expr()[1][j])]))];
L1 > else R:=[op(R),op(BracketOfSubspaces([lhs(expr()[1][i])],[lhs(expr()[1][j])])))-
op(BracketOfSubspaces([rhs(expr()[1][i])],[rhs(expr()[1][j])]))];
L1 > end if;
L1 > end do;
L1 > end do;
L1 > return [op(eval(R,expr()[1])),[op(expr()[2])]];
L1 > end proc:

```

Next, we define two variables, `Listexp` and `Listcond`, to save the two outputs of the routine `Listeq` respectively. Finally, we implement the routine `sys` to solve the system of equations resulting from the previous expressions. This routine must receive as input the lists `Listexp` and `Listcond`, returning as output the set of solutions which determine the coefficients of the representation of  $\mathfrak{g}$  by using Lie algebra  $\mathfrak{h}_k$ .

```

L1 > sys:=proc(L,M)
L1 > local Q;
L1 > Q:=[];
L1 > for i from 1 to nops(L) do
L1 > Q:=[op(Q),seq(coeff(L[i],x||j),j=1..nops(C[1](h)))]];
L1 > end do;
L1 > Q:=[op(Q),op(M)];
L1 > return solve(Q);
L1 > end proc:

```

If no solution is obtained, then it is not possible to represent  $\mathfrak{g}$  as a Lie subalgebra of Lie algebra  $\mathfrak{h}_k$  and we must try with the next Lie algebra:  $\mathfrak{h}_{k+1}$ . Therefore, we would have to repeat the process from the execution of  $\mathfrak{h}_{k+1}$  with the sentence `DGsetup(law_h(k), [x], [a])`, but replacing `k` with `k+1`.

### 3.2 Examples of application

Next, we show an example with the 3-dimensional solvable Lie algebra with law  $[e_1, e_3] = e_2$ . We must run all the routines implemented in the previous section. Here, we only

reproduce the most important outputs and those sentences to be modified for this specific example. To define the solvable Lie algebra  $\mathfrak{g}$ , the follow sentence is run

```
> L:= _DG(["LieAlgebra", g, [3], [[1, 3, 2], 1]]);
> DGsetup(L);
```

We must fill in `DerivedSeries` the value  $n = 3$  and then, execute the following sentences

```
> DerivedSeries();
      [3, 1, 0]
> assign(index_g,nops(DerivedSeries()));
> index_g;
      3
> C[1](g):=[seq(e||i, i=1..3)];
      C[1](g):=[e1,e2,e3]
> C[2](g):=DerivedAlgebra();
      C[2](g):=[e2]
> for i from 3 to index_g do C[i](g):=DerivedAlgebra(C[i-1](g));
      C[3](g):=[]
```

At this point, loading the routine `DimRepresentation`, we obtain

```
> DimRepresentation();
      2
```

Therefore, we must use  $k = 2$  in Step 1 of the algorithm. We look for a representation of  $\mathfrak{g}$  as a Lie subalgebra of  $\mathfrak{h}_2$ . Now we start loading Lie algebra  $\mathfrak{h}_2$  and we define all the terms of its derived series as follows

```
g > DGsetup(law_h(2), [x], [a])
      Lie algebra: L1
L1 > MultiplicationTable("LieBracket");
      [[x1, x2] = x2, [x2, x3] = x2]
L1 > C[1](h):=[seq(x||i, i=1..2*3/2)];
      C[1](h):=[x1,x2,x3]
L1 > C[2](h):=DerivedAlgebra();
      C[2](h):=[x2]
L1 > for i from 3 to nops(DerivedSeries_h(2)) do C[i](h):=DerivedAlgebra(C[i-1](h));end do;
      C[3](h):=[]
```

By executing the routine `expr`, we obtain the following output

```
L1 > expr();
      [e1=a[1,1]*x1+a[1,2]*x2+a[1,3]*x3,e2=a[2,2]*x2,e3=a[3,1]*x1+a[3,2]*x2+a[3,3]*x3],
      [a[1,1]^2+a[1,2]^2+a[1,3]^2<>0,a[2,2]^2<>0,a[3,1]^2+a[3,2]^2+a[3,3]^2<>0]
```

The output of `Listeq` is

```
L1 > Listeq();
      [(-a[1,3]*a[2,2]+a[1,1]*a[2,2])*x2,a[2,2]*x2-(-a[1,3]*a[3,2]+a[1,2]*a[3,3]-
      a[1,2]*a[3,1]+a[1,1]*a[3,2])*x2,(a[2,2]*a[3,3]-a[2,2]*a[3,1])*x2],
      [a[1,1]^2+a[1,2]^2+a[1,3]^2<>0,a[2,2]^2<>0,a[3,1]^2+a[3,2]^2+a[3,3]^2<>0]
```

After defining the variables `Listexp` and `Listcond` from the previous output, the routine `sys` is executed as follows

```
L1 > sys(Listexp,Listcond);
```

Since no answer is returned, there is no solution for the underlying system. Hence, Lie algebra  $\mathfrak{g}$  cannot be represented as a Lie subalgebra of  $\mathfrak{h}_2$ . Thus, the process must now be repeated from the execution of the sentence `DGsetup(law_h(2), [x], [a])`, where  $\mathfrak{h}_2$  is replaced with  $\mathfrak{h}_3$ .

```
g > DGsetup(law_h(3), [x], [a])
      Lie algebra: L1
L1 > MultiplicationTable("LieBracket");
      [[x1,x2]=x2, [x1,x3]=x3, [x2,x4]=x2, [x2,x5]=x3, [x3,x6]=x3, [x4,x5]=x5, [x5,x6]=x5]
L1 > C[1](h):=[seq(x|i, i=1..3*4/2)];
      C[1](h):=[x1,x2,x3,x4,x5,x6]
L1 > C[2](h):=DerivedAlgebra();
      C[2](h):=[x2,x3,x5]
L1 > for i from 3 to nops(DerivedSeries_h(3)) do
L1 >   C[i](h):=DerivedAlgebra(C[i-1](h));
L1 > end do;
      C[3](h):=[x3]
      C[4](h):=[]
```

This time, the variables `Listexp` and `Listcond` are defined from the output of `Listeq` as follows

```
L1 > Listexp:=((-a[1,4]*a[2,2]+a[1,1]*a[2,2])*x2+(-a[1,6]*a[2,3]-a[1,5]*a[2,2]+
a[1,2]*a[2,5]+a[1,1]*a[2,3])*x3+(-a[1,6]*a[2,5]+a[1,4]*a[2,5])*x5,
a[2,2]*x2+a[2,3]*x3+a[2,5]*x5-(-a[1,4]*a[3,2]+a[1,2]*a[3,4]-a[1,2]*a[3,1]+
a[1,1]*a[3,2])*x2-(-a[1,6]*a[3,3]-a[1,5]*a[3,2]+a[1,3]*a[3,6]-a[1,3]*a[3,1]+
a[1,2]*a[3,5]+a[1,1]*a[3,3])*x3-(-a[1,6]*a[3,5]+a[1,5]*a[3,6]-a[1,5]*a[3,4]+
a[1,4]*a[3,5])*x5, (a[2,2]*a[3,4]-a[2,2]*a[3,1])*x2+(-a[2,5]*a[3,2]+
a[2,3]*a[3,6]-a[2,3]*a[3,1]+a[2,2]*a[3,5])*x3+(a[2,5]*a[3,6]-a[2,5]*a[3,4])*x5);
L1 > Listcond:=[a[1,1]^2+a[1,2]^2+a[1,3]^2+a[1,4]^2+a[1,5]^2+a[1,6]^2<>0, a[2,2]^2+
a[2,3]^2+a[2,5]^2<>0, a[3,1]^2+a[3,2]^2+a[3,3]^2+a[3,4]^2+a[3,5]^2+a[3,6]^2<>0];
```

Finally, the routine `sys` is executed with `Listexp` and `Listcond` as parameters

```
L1 > sys(Listexp,Listcond);
{a[1,1]=a[1,6], a[1,2]=a[1,2], a[1,3]=a[1,3], a[1,4]=a[1,6], a[1,5]=a[1,5], a[1,6]=a[1,6],
a[2,2]=0, a[2,3]=-a[1,5]*a[3,2]+a[1,2]*a[3,5], a[2,5]=0, a[3,1]=a[3,1], a[3,2]=a[3,2],
a[3,3]=a[3,3], a[3,4]=a[3,1], a[3,5]=a[3,5], a[3,6]=a[3,1]},
{a[1,1]=a[1,1], a[1,2]=a[1,2], a[1,3]=a[1,3], a[1,4]=a[1,1], a[1,5]=a[1,5], a[1,6]=a[1,1],
a[2,2]=0, a[2,3]=-a[1,5]*a[3,2]+a[1,2]*a[3,5], a[2,5]=0, a[3,1]=a[3,1], a[3,2]=a[3,2],
a[3,3]=a[3,3], a[3,4]=a[3,1], a[3,5]=a[3,5], a[3,6]=a[3,1]}
```

As a particular solution of this system, we obtain the representative

$$e_1 = -x_5, \quad e_2 = x_3, \quad e_3 = x_2$$

or, by considering the original notation for basis  $\mathcal{B}_3$ ,

$$e_1 = -X_{2,3}, \quad e_2 = X_{1,3}, \quad e_3 = X_{1,2}$$

To conclude this section, we would like to point out that our algorithmic method is not only valid for low-dimensional solvable Lie algebras; but it provides minimal faithful representations for solvable Lie algebras of higher dimension. In this sense, we have computed some other examples corresponding to solvable Lie algebras such that their dimension is greater than the last being classified, namely: dimension  $n \geq 7$ . More concretely, Kobel [19] gave the list of 7-dimensional solvable Lie algebras with codimension 1. From that list, we have considered that with law

$$[e_2, e_7] = e_3, [e_3, e_7] = e_4, [e_4, e_7] = e_5, [e_5, e_7] = e_6, [e_6, e_7] = e_6$$

and our algorithm returns the following representation

$$\begin{aligned} e_1 &= X_{1,1}, \quad e_2 = X_{2,3}, \quad e_3 = X_{2,4}, \quad e_4 = X_{2,5}, \quad e_5 = X_{2,6}, \\ e_6 &= X_{2,6}, \quad e_7 = X_{3,4} + X_{4,5} + X_{5,6} + X_{6,6} \end{aligned}$$

We have also considered a second example consisting of the 8-dimensional solvable Lie algebra with a 4-dimensional abelian ideal and law

$$[e_1, e_2] = e_3, [e_2, e_5] = e_6, [e_4, e_5] = e_8, [e_1, e_6] = e_7, [e_2, e_6] = e_8, [e_3, e_5] = e_7.$$

In this case, our algorithmic method gives this representation

$$\begin{aligned} e_1 &= X_{1,4} + X_{3,5}, \quad e_2 = X_{1,4} + X_{5,6}, \quad e_3 = X_{3,6}, \quad e_4 = -2X_{1,2}, \\ e_5 &= X_{1,3} + X_{2,6} + X_{4,5}, \quad e_6 = X_{1,5} - X_{4,6}, \quad e_7 = -X_{1,6}, \quad e_8 = -2X_{1,6}. \end{aligned}$$

## 4 Solvable Lie algebras of dimension less than 6

This section is devoted to apply the algorithm implemented in Section 3, obtaining a minimal faithful upper-triangular matrix representation for each solvable Lie algebra of dimension less than 6. In addition, we compute such representations for several important families of  $n$ -dimensional solvable Lie algebras. Tables 1 to 3 show the classification of solvable Lie algebras of dimension less than 6 given in [13, 14], taking into account that we have only written the nonzero brackets in the law of each Lie algebra; whereas Tables 4 to 6 contain a representative for each algebra in the previous tables. In virtue of these tables, we can state the following

**Proposition 2.** *A minimal faithful representation by upper-triangular matrices for each solvable Lie algebra of dimension less than 6 with the dimension of such a minimal representation is given in Tables 4 to 6. Moreover, such representations can be obtained with a natural representative.*

**Remark 1.** For a natural representative, we mean a subalgebra in the corresponding  $\mathfrak{h}_k$  such that there are as many null coefficients as possible in its basis vectors and the remaining parameters are normalized.

Next, we show several results to determine a representative for minimal faithful upper-triangular matrix representations of three different families of solvable Lie algebras. Proposition 3 deals with a family of solvable non-nilpotent Lie algebras. Then, Proposition 4 computes a minimal representative for Heisenberg algebras. These Lie algebras constitute a special subclass of nilpotent Lie algebras and are very interesting for their applications to both the theory of nilpotent Lie algebras itself and Theoretical Physics. Finally, Proposition 5 provides a minimal representation for model filiform Lie algebras. These algebras are the most structured Lie algebras in the nilpotent class and were introduced by Vergne [17] in 1966. These propositions can be proved by applying the algorithm considered in Section 3 from a theoretical point of view and not by running the implementation. More concretely, the value of invariant  $\bar{\mu}$  for each algebra corresponds to the first natural integer  $k$  which can be considered in Step 1 of the above-mentioned algorithm.

**Proposition 3.** Let  $\mathfrak{s}_n$  be an  $n$ -dimensional solvable Lie algebra with basis  $\{e_i\}_{i=1}^n$  and law  $[e_i, e_n] = e_i$ , for  $1 \leq i < n$ . Then,  $\bar{\mu}(\mathfrak{s}_n) = n$ . In fact, a natural representative of  $\mathfrak{s}_n$  is given by

$$\{e_j = X_{1,j+1}\}_{j=1}^{j=n-1} \cup \{e_n = -X_{1,1}\}$$

**Proposition 4.** Let  $\mathfrak{H}_{2n+1}$  be the  $(2n + 1)$ -dimensional Heisenberg algebra with basis  $\{e_i\}_{i=1}^{2n+1}$  and law  $[e_{2i}, e_{2i+1}] = e_1$ , for  $1 \leq i \leq n$ . Then,  $\bar{\mu}(\mathfrak{H}_{2n+1}) = n + 2$ . Moreover, a natural representative of  $\mathfrak{H}_{2n+1}$  is given by

$$\{e_{2j+1} = X_{j+1,n+2}\}_{j=0}^{j=n} \cup \{e_{2k} = X_{1,k+1}\}_{k=1}^{n+1}$$

**Proposition 5.** Let  $\mathfrak{f}_n$  be the  $n$ -dimensional filiform Lie algebra with basis  $\{e_i\}_{i=1}^n$  and law  $[e_1, e_h] = e_{h-1}$ , for  $3 \leq h \leq n$ . Then,  $\bar{\mu}(\mathfrak{f}_n) = n$ . Moreover, a natural representative of  $\mathfrak{f}_n$  is given by

$$\left\{ e_1 = \sum_{i=1}^{n-2} X_{i,i+1} \right\} \cup \{e_j = X_{j-1,n}\}_{j=2}^n$$

## References

- [1] [N. Jacobson, A note on automorphisms and derivations of Lie algebras, \*Proc. Amer. Math. Soc.\* \*\*6\*\* \(1955\) 281–283.](#)
- [2] [Y.A. Neretin, A construction of finite-dimensional faithful representation of Lie algebras, \*Rend. Circ. Mat. Palermo Supp.\* \*\*71\*\* \(2003\) 159–161.](#)
- [3] [V.S. Varadaraajan, \*Lie Groups, Lie Algebras and their Representations\*, Selected Monographies \*\*17\*\*, Collège Press, Beijing, 1998.](#)

- [4] [D. Burde, On a refinement of Ado's Theorem. \*Arch. Math. \(Basel\)\* \*\*70\*\* \(1998\) 118–127.](#)
- [5] [R. Ghanam, I. Strugar, G. Thompson, Matrix representations for low dimensional Lie algebras, \*Extracta Math.\* \*\*20\*\* \(2005\) 151–184.](#)
- [6] [J. Milnor, On fundamental groups of complete affinely flat manifolds, \*Adv. Math.\* \*\*25\*\* \(1977\) 178–187.](#)
- [7] [L. Auslander, The structure of complete locally affine manifolds, \*Topology\* \*\*3\*\*: Suppl. 1 \(1964\) 131–139.](#)
- [8] [L. Auslander, Simply transitive groups of affine motions, \*Amer. J. Math.\* \*\*99\*\*:4 \(1977\) 809–826.](#)
- [9] [J.C. Benjumea, F.J. Echarte, J. Núñez, A.F. Tenorio, A method to obtain the Lie group associated with a nilpotent Lie algebra, \*Comput. Math. Appl.\* \*\*51\*\* \(2006\) 1493–1506.](#)
- [10] [J.C. Benjumea, J. Núñez, A.F. Tenorio, Minimal linear representations of the low-dimensional nilpotent Lie algebras, \*Math. Scand.\* \*\*102\*\* \(2008\) 17–26.](#)
- [11] [M. Ceballos, J. Núñez, A.F. Tenorio, Representing Filiform Lie Algebras Minimally and Faithfully by Strictly Upper-Triangular Matrices, \*Journal of Algebra and its Applications\* \*\*12\*\* \(2013\) 1250196.](#)
- [12] [J. Núñez, A.F. Tenorio, Minimal Faithful Upper-triangular Matrix Representations for Low-Dimensional Solvable Lie Algebras, in: \*Electronic Proceedings of MEGA 2007 - Effective Methods in Algebraic Geometry\*, Strobl, 2007, 13 pp.](#)
- [13] [A. Andrada, M.L. Barberis, I.G. Dotti, G.P. Ovando, Product structures on four dimensional solvable Lie algebras, \*Homology Homotopy Appl.\* \*\*7\*\* \(2005\) 9–37.](#)
- [14] [G.M. Mubarakzyanov, On solvable Lie algebras, \*Izv. Vyssh. Uchebn. Zaved. Mat.\* \*\*32\*\* \(1\) \(1963\) 114–123 \(in Russian\).](#)
- [15] [G.M. Mubarakzyanov, The classification of the real structure of five-dimensional Lie algebras, \*Izv. Vyssh. Uchebn. Zaved. Mat.\* \*\*34\*\* \(3\) \(1963\) 99–106 \(in Russian\).](#)
- [16] [P. Turkowski, Solvable Lie algebras of dimension six, \*J. Math. Phys.\* \*\*31\*\* \(1990\) 1344–1350.](#)
- [17] [M. Vergne, Cohomologie des algèbres de Lie nilpotentes, Application à l'étude de la variété des algèbres de Lie nilpotentes, \*Bull. Soc. Math. France\* \*\*98\*\* \(1970\) 81–116.](#)
- [18] [W. Fulton, J. Harris, \*Representation theory: a first course\*, Springer-Verlag, New York, 1991.](#)

- [19] C. Kobel, *On the Classification of Solvable Lie Algebras of Finite Dimension Containing an Abelian Ideal of Codimension One*, Master's Thesis, School of Information Science, Computer and Electrical Engineering, Halmstad University, Halmstad, 2008.

Table 1: Solvable Lie algebras of dimension less than 5.

Dim.	Lie algebra	(Non-zero) Lie brackets
1	$\mathfrak{s}_1^1$	—
2	$\mathfrak{s}_2^1$ $\mathfrak{s}_2^2$	— $[e_1, e_2] = e_1$
3	$\mathfrak{s}_3^1$ $\mathfrak{s}_3^2$ $\mathfrak{s}_3^3$ $\mathfrak{s}_3^4$ $\mathfrak{s}_3^5$ $\mathfrak{s}_3^6$	— $[e_1, e_3] = e_2$ $[e_1, e_3] = e_1, [e_2, e_3] = e_2$ $[e_1, e_3] = e_2, [e_2, e_3] = -e_1$ $[e_1, e_3] = -e_1, [e_2, e_3] = -e_1 - e_2$ $[e_1, e_3] = -e_1$
4	$\mathfrak{s}_4^1$ $\mathfrak{s}_4^2$ $\mathfrak{s}_4^3$ $\mathfrak{s}_4^4$ $\mathfrak{s}_4^5$ $\mathfrak{s}_4^6$ $\mathfrak{s}_4^7$ $\mathfrak{s}_4^8$ $\mathfrak{s}_4^9$ $\mathfrak{s}_4^{10}$ $\mathfrak{s}_4^{11}$ $\mathfrak{s}_4^{12}$	— $[e_1, e_3] = e_2, [e_1, e_4] = e_3$ $[e_1, e_3] = e_3, [e_1, e_4] = e_4, [e_2, e_3] = e_4$ $[e_1, e_3] = e_3, [e_1, e_4] = e_4, [e_2, e_3] = -e_4, [e_2, e_4] = e_3$ $[e_1, e_3] = e_3, [e_1, e_2] = e_4$ $[e_4, e_1] = e_1, [e_4, e_2] = \alpha e_2, [e_4, e_3] = \beta e_3$ $[e_3, e_1] = \alpha e_1, [e_3, e_2] = e_2, [e_3, e_4] = e_2 + e_4$ $[e_1, e_2] = e_2 + e_3, [e_1, e_3] = e_3 + e_4, [e_1, e_4] = e_4$ $[e_1, e_2] = \beta e_2 - e_3, [e_1, e_3] = e_2 + \beta e_3, [e_1, e_4] = \alpha e_4$ $[e_1, e_2] = (\alpha - 1)e_2, [e_1, e_3] = e_3, [e_1, e_4] = \alpha e_4, [e_2, e_3] = e_4$ $[e_1, e_2] = e_2 + e_3, [e_1, e_3] = e_3, [e_1, e_4] = 2e_4, [e_2, e_3] = e_4$ $[e_1, e_2] = \alpha e_2 - e_3, [e_1, e_3] = e_2 + \alpha e_3, [e_1, e_4] = 2\alpha e_4, [e_2, e_3] = e_4$

Table 2: 5-dimensional non-decomposable real solvable Lie algebras

Lie algebra	(Non-zero) Lie brackets	Parameters
$\mathfrak{g}_{5,1}$	$[e_1, e_3] = e_5, [e_2, e_4] = e_5$	
$\mathfrak{g}_{5,2}$	$[e_1, e_2] = e_4, [e_1, e_3] = e_5$	
$\mathfrak{g}_{5,3}$	$[e_1, e_2] = e_4, [e_1, e_4] = e_5, [e_2, e_3] = e_5$	
$\mathfrak{g}_{5,4}$	$[e_1, e_2] = e_3, [e_1, e_3] = e_4, [e_2, e_3] = e_5$	
$\mathfrak{g}_{5,5}$	$[e_1, e_2] = e_3, [e_1, e_3] = e_4, [e_1, e_4] = e_5$	
$\mathfrak{g}_{5,6}$	$[e_1, e_2] = e_3, [e_1, e_3] = e_4, [e_1, e_4] = e_5, [e_2, e_3] = e_5$	
$\mathfrak{g}_{5,7}$	$[e_1, e_5] = e_1, [e_2, e_5] = \alpha e_2,$ $[e_3, e_5] = \beta e_3, [e_4, e_5] = \gamma e_4$	$-1 \leq \gamma \leq \beta \leq \alpha \leq 1,$ $\alpha\beta\gamma \neq 0.$
$\mathfrak{g}_{5,8}$	$[e_2, e_5] = e_1, [e_3, e_5] = e_3, [e_4, e_5] = \gamma e_4,$	$0 <  \gamma  \leq 1$
$\mathfrak{g}_{5,9}$	$[e_1, e_5] = e_1, [e_2, e_5] = e_1 + e_3, [e_3, e_5] = \beta e_3, [e_4, e_5] = \gamma e_4$	$0 \neq \gamma \leq \beta$
$\mathfrak{g}_{5,10}$	$[e_2, e_5] = e_1, [e_3, e_5] = e_2, [e_4, e_5] = e_4$	
$\mathfrak{g}_{5,11}$	$[e_1, e_5] = e_1, [e_2, e_5] = e_1 + e_2, [e_3, e_5] = e_2 + e_3, [e_4, e_5] = \gamma e_4$	$\gamma \neq 0$
$\mathfrak{g}_{5,12}$	$[e_1, e_5] = e_1, [e_2, e_5] = e_1 + e_2,$ $[e_3, e_5] = e_2 + e_3, [e_4, e_5] = e_3 + e_4$	
$\mathfrak{g}_{5,13}$	$[e_1, e_5] = e_1, [e_2, e_5] = \gamma e_2,$ $[e_3, e_5] = p e_3 - s e_4, [e_4, e_5] = s e_3 + p e_4$	$\gamma s \neq 0,  \gamma  \leq 1$
$\mathfrak{g}_{5,14}$	$[e_2, e_5] = e_1, [e_3, e_5] = p e_3 - e_4, [e_4, e_5] = e_3 + p e_4$	
$\mathfrak{g}_{5,15}$	$[e_1, e_5] = e_1, [e_3, e_5] = \gamma e_3,$ $[e_2, e_5] = e_1 + e_2, [e_4, e_5] = e_3 + \gamma e_4$	$-1 \leq \gamma \leq 1$
$\mathfrak{g}_{5,16}$	$[e_1, e_5] = e_1, [e_2, e_5] = e_1 + e_2,$ $[e_3, e_5] = p e_3 - s e_4, [e_4, e_5] = s e_3 + p e_4$	$s \neq 0$
$\mathfrak{g}_{5,17}$	$[e_1, e_5] = p e_1 - e_2, [e_2, e_5] = e_1 + p e_2,$ $[e_3, e_5] = q e_3 - s e_4, [e_4, e_5] = s e_3 + q e_4$	$s \neq 0$
$\mathfrak{g}_{5,18}$	$[e_3, e_5] = e_1 + p e_3 - e_4, [e_2, e_5] = e_1 + p e_2$ $[e_1, e_5] = p e_1 - e_2, [e_4, e_5] = e_2 + e_3 - p e_4$	$p \geq 0$
$\mathfrak{g}_{5,19}$	$[e_2, e_3] = e_1, [e_1, e_5] = (1 + \alpha)e_1,$ $[e_2, e_5] = e_2, [e_3, e_5] = \alpha e_3, [e_4, e_5] = \beta e_4$	$\beta \neq 0$
$\mathfrak{g}_{5,20}$	$[e_2, e_3] = e_1, [e_1, e_5] = (1 + \alpha)e_2, [e_2, e_5] = e_2,$ $[e_3, e_5] = \alpha e_3, [e_4, e_5] = e_1 + (1 + \alpha)e_4$	
$\mathfrak{g}_{5,21}$	$[e_2, e_3] = e_1, [e_1, e_5] = 2e_1, [e_4, e_5] = e_4,$ $[e_2, e_5] = e_2 + e_3, [e_3, e_5] = e_3 + e_4$	
$\mathfrak{g}_{5,22}$	$[e_2, e_3] = e_1, [e_2, e_5] = e_3, [e_4, e_5] = e_4$	
$\mathfrak{g}_{5,23}$	$[e_2, e_3] = e_1, [e_1, e_5] = 2e_1, [e_3, e_5] = e_3,$ $[e_2, e_5] = e_2 + e_3, [e_4, e_5] = \beta e_4$	$\beta \neq 0$
$\mathfrak{g}_{5,24}$	$[e_2, e_3] = e_1, [e_1, e_5] = 2e_1, [e_3, e_5] = e_3,$ $[e_2, e_5] = e_2 + e_3, [e_4, e_5] = \epsilon e_1 + 2e_4$	$\epsilon = \pm 1$
$\mathfrak{g}_{5,25}$	$[e_2, e_3] = e_1, [e_1, e_5] = 2p e_1, [e_4, e_5] = \beta e_4,$ $[e_2, e_5] = p e_2 + e_3, [e_3, e_5] = -e_2 + p e_3$	$\beta \neq 0$
$\mathfrak{g}_{5,26}$	$[e_2, e_5] = p e_2 + e_3, [e_1, e_5] = 2p e_1, [e_2, e_3] = e_1,$ $[e_3, e_5] = -e_2 + p e_3, [e_4, e_5] = \epsilon e_1 + 2p e_4$	$\epsilon = \pm 1$
$\mathfrak{g}_{5,27}$	$[e_2, e_3] = e_1, [e_3, e_5] = e_3 + e_4, [e_4, e_5] = e_1 + e_4, [e_1, e_5] = e_1$	
$\mathfrak{g}_{5,28}$	$[e_2, e_3] = e_1, [e_2, e_5] = \alpha e_2, [e_4, e_5] = e_4,$ $[e_1, e_5] = (1 + \alpha)e_1, [e_3, e_5] = e_3 + e_4$	
$\mathfrak{g}_{5,29}$	$[e_2, e_3] = e_1, [e_1, e_5] = e_1, [e_2, e_5] = e_2, [e_3, e_5] = e_4$	
$\mathfrak{g}_{5,30}$	$[e_2, e_4] = e_1, [e_3, e_4] = e_2,$ $[e_1, e_5] = (2 + h)e_1, [e_4, e_5] = e_4,$ $[e_2, e_5] = (1 + h)e_2, [e_3, e_5] = h e_3$	
$\mathfrak{g}_{5,31}$	$[e_2, e_4] = e_1, [e_3, e_4] = e_2, [e_1, e_5] = 3e_1, [e_3, e_5] = e_3,$ $[e_2, e_5] = 2e_2, [e_4, e_5] = e_3 + e_4$	
$\mathfrak{g}_{5,32}$	$[e_2, e_4] = e_1, [e_3, e_4] = e_2, [e_1, e_5] = e_1,$ $[e_2, e_5] = e_2, [e_3, e_5] = h e_1 + e_3$	
$\mathfrak{g}_{5,33}$	$[e_1, e_4] = e_1, [e_3, e_4] = \beta e_3, [e_2, e_5] = e_2, [e_3, e_5] = \gamma e_3$	$\beta^2 + \gamma^2 \neq 0$



Table 3: 5-dimensional non-decomposable real solvable Lie algebras (II)

Lie algebra	(Non-zero) Lie brackets	Parameters
$\mathfrak{g}_{5,34}$	$[e_1, e_4] = e_1, [e_2, e_4] = e_2,$ $[e_3, e_4] = e_3, [e_1, e_5] = e_1, [e_3, e_5] = e_2$	
$\mathfrak{g}_{5,35}$	$[e_1, e_4] = he_1, [e_2, e_4] = e_2, [e_3, e_4] = e_3,$ $[e_2, e_5] = -e_3, [e_1, e_5] = \alpha e_1, [e_3, e_5] = e_2$	$h^2 + \alpha^2 \neq 0$
$\mathfrak{g}_{5,36}$	$[e_2, e_3] = e_1, [e_1, e_4] = e_1,$ $[e_2, e_4] = e_2, [e_3, e_5] = e_3, [e_2, e_5] = -e_2$	
$\mathfrak{g}_{5,37}$	$[e_2, e_3] = e_1, [e_1, e_4] = 2e_1, [e_2, e_4] = e_2,$ $[e_3, e_4] = e_3, [e_2, e_5] = -e_3, [e_3, e_5] = e_2$	
$\mathfrak{g}_{5,38}$	$[e_1, e_4] = e_1, [e_2, e_5] = e_2, [e_4, e_5] = e_3$	
$\mathfrak{g}_{5,39}$	$[e_1, e_4] = e_1, [e_2, e_4] = e_2,$ $[e_1, e_5] = -e_2, [e_2, e_5] = e_1, [e_4, e_5] = e_3$	

Table 4: Representation of Solvable Lie algebras of dimension less than 5.

Lie algebra	Representation	$\bar{\mu}$
$\mathfrak{s}_1^1$	$e_1 = X_{11}$	1
$\mathfrak{s}_2^1$	$e_1 = X_{11}, e_2 = X_{22}$	2
$\mathfrak{s}_2^2$	$e_1 = X_{11}, e_2 = X_{12}$	2
$\mathfrak{s}_3^1$	$e_1 = X_{11}, e_2 = X_{22}, e_3 = X_{33}$	3
$\mathfrak{s}_3^2$	$e_1 = X_{12}, e_2 = X_{13}, e_3 = X_{23}$	3
$\mathfrak{s}_3^3$	$e_1 = X_{13}, e_2 = X_{12}, e_3 = -X_{1,1}$	3
$\mathfrak{s}_3^4$	$e_1 = X_{12}, e_2 = iX_{12} + X_{13}, e_3 = iX_{22} + X_{23} - iX_{33}$	3
$\mathfrak{s}_3^5$	$e_1 = X_{13}, e_2 = X_{23}, e_3 = X_{12} - X_{33}$	3
$\mathfrak{s}_3^6$	$e_1 = X_{12}, e_2 = X_{11} + X_{22}, e_3 = -X_{22}$	2
$\mathfrak{s}_4^1$	$e_1 = X_{11}, e_2 = X_{22}, e_3 = X_{33}, e_4 = X_{44}$	4
$\mathfrak{s}_4^2$	$e_1 = -(X_{23} + X_{34}), e_2 = X_{14}, e_3 = X_{13}, e_4 = X_{12}$	4
$\mathfrak{s}_4^3$	$e_1 = X_{11}, e_2 = -X_{23}, e_3 = X_{12}, e_4 = X_{13}$	3
$\mathfrak{s}_4^4$	$e_1 = X_{11}, e_2 = -iX_{22} + X_{23} + iX_{33}, e_3 = X_{12}, e_4 = -iX_{12} + X_{13}$	3
$\mathfrak{s}_4^5$	$e_1 = X_{12} + X_{33}, e_2 = X_{24}, e_3 = X_{34}, e_4 = X_{14}$	4
$\mathfrak{s}_4^6$	$e_1 = X_{14}, e_2 = X_{24}, e_3 = X_{34}, e_4 = (\alpha - 1)X_{22} + (\beta - 1)X_{33} - X_{44}$	4
$\mathfrak{s}_4^7$	$e_1 = X_{14}, e_2 = X_{13}, e_3 = X_{11} + X_{23} + (1 - \alpha)X_{44}, e_4 = -X_{12}$	4
$\mathfrak{s}_4^8$	$e_1 = X_{12} - X_{23} - X_{44}, e_2 = X_{34}, e_3 = -X_{24}, e_4 = -X_{14}$	4
$\mathfrak{s}_4^9$	$e_1 = \alpha X_{11} + (\alpha - \beta - i)X_{22} + (\beta - i)X_{33},$ $e_2 = i(X_{12} - X_{34}), e_3 = X_{12} + X_{34}, e_4 = X_{14}$	4
$\mathfrak{s}_4^{10}$	$e_1 = \alpha X_{11} + X_{22}, e_2 = X_{12}, e_3 = X_{23}, e_4 = X_{13}$	3
$\mathfrak{s}_4^{11}$	$e_1 = X_{12} + X_{11} + X_{22} - X_{44}, e_2 = X_{23} - X_{34}, e_3 = X_{13}, e_4 = X_{14}$	4
$\mathfrak{s}_4^{12}$	$e_1 = 2\alpha X_{11} + (\alpha - i)X_{22}, e_2 = X_{23} + X_{12},$ $e_3 = i(X_{23} - X_{12}), e_4 = 2iX_{13}$	3

Table 5: Representation of 5-dimensional non-decomposable real solvable Lie algebras

Lie algebra	Representation	$\bar{\mu}$
$\mathfrak{g}_{5,1}$	$e_1 = X_{1,2}, e_2 = X_{1,3}, e_3 = X_{1,4}, e_4 = X_{1,5}, e_6 = X_{2,3} + X_{3,4} + X_{4,5}$	4
$\mathfrak{g}_{5,2}$	$e_1 = X_{1,2} + X_{2,4} + X_{3,5}, e_2 = X_{1,3}, e_3 = X_{1,4}, e_4 = X_{1,5}, e_5 = X_{2,3} + X_{3,4} + X_{4,5}$	5
$\mathfrak{g}_{5,3}$	$e_1 = X_{1,2} - X_{3,5}, e_2 = X_{1,3} + X_{2,5}, e_3 = X_{1,4}, e_4 = X_{1,5}, e_5 = X_{2,3} + X_{3,4}$	5
$\mathfrak{g}_{5,4}$	$e_1 = X_{1,2} + X_{2,3}, e_2 = X_{1,3}, e_3 = X_{1,4}, e_4 = X_{2,4}, e_5 = X_{3,4}$	4
$\mathfrak{g}_{5,5}$	$e_1 = X_{1,2}, e_2 = X_{1,3}, e_3 = X_{1,4}, e_4 = X_{2,3}, e_5 = X_{2,4}$	4
$\mathfrak{g}_{5,6}$	$e_1 = X_{1,2}, e_2 = X_{1,3}, e_3 = X_{1,4}, e_4 = X_{2,4}, e_5 = X_{3,4}$	4
$\mathfrak{g}_{5,7}$	$e_1 = X_{1,4}, e_2 = X_{1,2}, e_3 = X_{1,5}, e_4 = X_{3,5},$ $e_5 = \alpha X_{2,2} + (\beta - \gamma)X_{3,3} + X_{4,4} + \beta X_{5,5}$	5
$\mathfrak{g}_{5,8}$	$e_1 = X_{1,4}, e_2 = X_{1,2} + X_{1,4}, e_3 = X_{1,5}, e_4 = X_{1,3},$ $e_5 = X_{2,4} + \gamma X_{3,3} + X_{5,5}$	5
$\mathfrak{g}_{5,9}$	$e_1 = X_{2,4}, e_2 = X_{1,3} + X_{2,4}, e_3 = \beta X_{1,3}, e_4 = X_{2,3},$ $e_5 = (\beta - \gamma)X_{2,2} + \beta X_{3,3} + (1 + \beta - \gamma)X_{4,4}$	4
$\mathfrak{g}_{5,10}$	$e_1 = -X_{1,5}, e_2 = -X_{1,4}, e_3 = X_{1,2} - X_{1,3}, e_4 = X_{1,3} + X_{1,4} + X_{1,5},$ $e_5 = X_{1,1} + X_{2,2} + X_{2,3} + 2X_{3,3} + X_{3,4} + X_{4,4} + X_{4,5} + X_{5,5}$	5
$\mathfrak{g}_{5,11}$	$e_1 = X_{1,5}, e_2 = \frac{1}{2}X_{1,4} + X_{2,5}, e_3 = X_{1,5} + X_{2,4}, e_4 = X_{3,5},$ $e_5 = -\frac{1}{2}X_{1,2} + (1 - \gamma)X_{3,3} + X_{4,4} + X_{4,5} + X_{5,5}$	5
$\mathfrak{g}_{5,12}$	$e_1 = X_{1,5}, e_2 = X_{1,4}, e_3 = X_{1,3}, e_4 = X_{1,2},$ $e_5 = X_{2,2} + X_{2,3} + X_{3,3} + X_{3,4} + X_{4,4} + X_{4,5} + X_{5,5}$	5
$\mathfrak{g}_{5,13}$	$e_1 = X_{2,4}, e_2 = X_{3,4}, e_3 = X_{1,4}, e_4 = iX_{1,4},$ $e_5 = (p - 1 - si)X_{2,2} + (p - \gamma - si)X_{3,3} + (p - si)X_{4,4}$	4
$\mathfrak{g}_{5,14}$	$e_1 = X_{2,4}, e_2 = X_{2,3}, e_3 = X_{1,4}, e_4 = -iX_{1,4},$ $e_5 = iX_{2,2} + iX_{3,3} + X_{3,4} + iX_{4,4}$	4
$\mathfrak{g}_{5,15}$	$e_1 = -X_{1,4}, e_2 = X_{2,4}, e_3 = X_{1,3}, e_4 = -X_{2,3}, e_5 = X_{1,2} + \gamma X_{3,3} + X_{4,4}$	4
$\mathfrak{g}_{5,16}$	$e_1 = X_{2,4}, e_2 = X_{2,3}, e_3 = X_{1,4}, e_4 = -iX_{1,4},$ $e_5 = siX_{2,2} + (1 + si)X_{3,3} + X_{3,4} + (1 + si)X_{4,4}$	4
$\mathfrak{g}_{5,17}$	$e_1 = X_{2,4}, e_2 = iX_{2,4}, e_3 = X_{1,4}, e_4 = iX_{1,4},$ $e_5 = (-p + iq - si)X_{2,2} + (q - si)X_{4,4}$	4
$\mathfrak{g}_{5,18}$	$e_1 = X_{2,4}, e_2 = iX_{2,4}, e_3 = -\frac{i(i-p)}{p}X_{2,4}, e_4 = \frac{i}{p}X_{2,4}, e_5 = (p - i)X_{4,4}$	4
$\mathfrak{g}_{5,19}$	$e_1 = X_{1,4}, e_2 = X_{1,2}, e_3 = X_{2,4}, e_4 = X_{3,4},$ $e_5 = X_{2,2} + (1 + \alpha - \beta)X_{3,3} + \sqrt{\beta(2 - \beta)} + 2\alpha(\beta - \alpha - 2) - 3X_{3,4} + (1 + \alpha)X_{4,4}$	4
$\mathfrak{g}_{5,20}$	$e_1 = -X_{1,4}, e_2 = -X_{1,3}, e_3 = ZX_{2,4} + X_{3,4}, e_4 = X_{2,4},$ $e_5 = -(1 + \alpha)X_{1,1} + X_{2,2} - ZX_{1,3} - (1 + \alpha)X_{2,2} + ZX_{2,3} + X_{2,4} - \alpha X_{3,3}$	4
$\mathfrak{g}_{5,21}$	$e_1 = X_{1,4}, e_2 = X_{2,3} + X_{3,4}, e_3 = -X_{1,3} + X_{2,5}, e_4 = -2X_{1,5},$ $e_5 = X_{1,2} + X_{3,3} + X_{3,5} + 2X_{4,4} + X_{5,5}$	5
$\mathfrak{g}_{5,22}$	$e_1 = X_{1,5}, e_2 = X_{1,2} + X_{2,3}, e_3 = X_{2,5}, e_4 = X_{1,4}, e_5 = X_{3,5} + X_{4,4}$	5
$\mathfrak{g}_{5,23}$	$e_1 = X_{1,4}, e_2 = X_{2,3} + X_{3,4}, e_3 = -X_{1,3}, e_4 = X_{1,5},$ $e_5 = -X_{1,1} + X_{1,2} - X_{2,2} + X_{4,4} + (\beta - 1)X_{5,5}$	5
$\mathfrak{g}_{5,24}$	$e_1 = X_{1,4}, e_2 = X_{2,3} + X_{3,4}, e_3 = -X_{1,3}, e_4 = X_{1,5} - \epsilon X_{2,4},$ $e_5 = -X_{1,1} + X_{1,2} - X_{2,2} + X_{4,4} + X_{5,5}$	5
$\mathfrak{g}_{5,25}$	$e_1 = -2iX_{1,4}, e_2 = -iX_{1,3} + iX_{3,4}, e_3 = X_{1,3} + X_{3,4}, e_4 = (\beta - 2p)X_{1,2} + ZX_{1,4},$ $e_5 = -2pX_{1,1} + (\beta - 2p)X_{2,2} + ZX_{2,4} + (i - p)X_{3,3}, Z = \sqrt{1 + 2pi + 4p\beta - \beta^2 - 9p^2}$	4
$\mathfrak{g}_{5,26}$	$e_1 = 2\omega X_{1,4}, e_2 = \omega X_{1,2} - (1 - p^2 + pi)X_{1,4} - \omega X_{2,4}, e_3 = X_{1,2} + (-p + i)X_{1,4} + X_{2,4},$ $e_4 = -2\omega X_{3,4}, e_5 = X_{1,3} + (p - \omega)X_{2,2} + X_{2,4} + ZX_{3,4} + 2pX_{4,4},$ $Z = \sqrt{2p^3i + 2pi - 2p^4 - 5p^2 - 1}, \omega = -p^3 + i(p^2 + 1)$	4
$\mathfrak{g}_{5,27}$	$e_1 = -X_{1,4}, e_2 = X_{1,3}, e_3 = -X_{3,4}, e_4 = X_{2,4}, e_5 = X_{1,2} + X_{2,3} + X_{4,4}$	4
$\mathfrak{g}_{5,28}$	$e_1 = -X_{1,4}, e_2 = (1 + \alpha)iX_{1,4} + X_{2,4}, e_3 = X_{1,2}, e_4 = X_{1,3},$ $e_5 = (1 + \alpha)iX_{1,2} + X_{2,2} + X_{2,3} + X_{3,3} + (1 + \alpha)X_{4,4}$	4
$\mathfrak{g}_{5,29}$	$e_1 = X_{1,4}, e_2 = X_{1,3}, e_3 = X_{1,1} + X_{2,2} + X_{3,3} + X_{3,4} + X_{4,4},$ $e_4 = -X_{2,4}, e_5 = X_{2,2} + X_{2,3} + X_{3,3} + X_{4,4}$	4
$\mathfrak{g}_{5,30}$	$e_1 = -2X_{1,4}, e_2 = -X_{1,3} + X_{1,4} + X_{2,4}, e_3 = X_{2,3} - X_{4,4},$ $e_4 = X_{1,2} + X_{3,4}, e_5 = X_{2,2} + (1 + h)X_{3,3} + X_{3,4} + (2 + h)X_{4,4}$	5
$\mathfrak{g}_{5,31}$	$e_1 = \frac{3}{2}X_{1,5}, e_2 = X_{1,4} - X_{2,5}, e_3 = \frac{1}{3}X_{1,3} - X_{2,4},$ $e_4 = \frac{1}{2}X_{1,2} + X_{3,4} + X_{4,5}, e_5 = X_{2,2} + X_{2,3} + X_{3,3} + 2X_{4,4} + 3X_{5,5}$	5

Table 6: Representation of 5-dimensional non-decomposable real solvable Lie algebras (II)

Lie algebra	Representation	$\bar{\mu}$
$\mathfrak{g}_{5,32}$	$e_1 = X_{1,4}, e_2 = X_{1,3}, e_3 = X_{1,2}, e_4 = X_{2,2} + X_{3,4},$ $e_5 = X_{2,2} + hX_{2,4} + X_{3,3} + X_{4,4}$	4
$\mathfrak{g}_{5,33}$	$e_1 = X_{2,4}, e_2 = X_{1,4}, e_3 = \gamma X_{2,3} + X_{2,4},$ $e_4 = X_{1,1} + \beta X_{3,3} + \frac{\beta-1}{\gamma} X_{3,4} + X_{4,4}, e_5 = -X_{1,1} + \gamma X_{3,3} + X_{3,4}$	4
$\mathfrak{g}_{5,34}$	$e_1 = X_{1,2}, e_2 = -X_{1,4}, e_3 = X_{3,4}, e_4 = X_{2,2} + X_{4,4}, e_5 = -X_{1,1} + X_{1,3} - X_{3,3} - X_{4,4}$	4
$\mathfrak{g}_{5,35}$	$e_1 = X_{2,4}, e_2 = X_{1,3}, e_3 = -iX_{1,3}, e_4 = X_{3,3} + hX_{4,4}, e_5 = -\alpha X_{2,2} + iX_{3,3}$	4
$\mathfrak{g}_{5,36}$	$e_1 = -X_{1,4}, e_2 = X_{2,4}, e_3 = X_{1,2}, e_4 = X_{4,4}, e_5 = X_{2,2} + iX_{3,3}$	4
$\mathfrak{g}_{5,37}$	$e_1 = 2iX_{1,4}, e_2 = X_{1,2} - iX_{1,4} + X_{2,4}, e_3 = -iX_{1,2} - X_{1,4} + iX_{2,4},$ $e_4 = X_{2,2} + iX_{2,4} + 2X_{4,4}, e_5 = iX_{2,2} + X_{2,4}$	4
$\mathfrak{g}_{5,38}$	$e_1 = X_{1,5}, e_2 = X_{1,4}, e_3 = X_{1,3}, e_4 = X_{1,2} + X_{5,5},$ $e_5 = X_{1,1} + X_{2,2} + X_{2,3} + X_{3,3} + 2X_{4,4} + X_{5,5}$	5
$\mathfrak{g}_{5,39}$	$e_1 = X_{2,4}, e_2 = -iX_{2,4}, e_3 = X_{1,4}, e_4 = -X_{2,2} + X_{3,4}, e_5 = iX_{1,1} - X_{1,3} + iX_{3,3} + iX_{4,4}$	4