LieAlgDB — A database of Lie algebras Sophus — Computing with nilpotent Lie algebras

Csaba Schneider
(joint with Willem de Graaf)

Computer and Automation Research Institute
Budapest
csaba.schneider@sztaki.hu
www.sztaki.hu/∼schneider

GAP Package Developers Workshop
Braunschweig, 14 September 2007

Sophus and LieAlgDB

Determining nilpotent Lie algebras

Problems and solutions

Generic computations

Classification theorems

Examples and Implementation

Sophus V1.22

Sophus: Computing with nilpotent Lie algebras

(i) Computing nice bases (`NilpotentBasis`)

(ii) Computing extensions (`LieCover`, `Descendants`)

(iii) Computing automorphism groups (`AutomorphismGroupOfNilpotentLieAlgebra`)

(iv) Testing for isomorphism (`AreIsomorphicNilpotentLieAlgebras`)

LieAlgDB V2.0.1

LieAlgDB: A database of nilpotent Lie algebras (with Willem de Graaf)

- Solvable of dimension at most 4 (`AllSolvableLieAlgebras`)
- Non-solvable of dimension at most 6 over FF (`AllNonSolvableLieAlgebras`);
- Nilpotent of dimension at most 6 over odd characteristic (`AllNilpotentLieAlgebras`);
- Nilpotent of dimension at most 9 over $\mathbf{F}_2$; at most 7 over $\mathbf{F}_3$ and $\mathbf{F}_5$ (`AllNilpotentLieAlgebras`);
- Simple of dimension at most 9 over $\mathbf{F}_2$ (`AllSimpleLieAlgebras`);

Sophus and LieAlgDB

Determining nilpotent Lie algebras

Problems and solutions

Generic computations

Classification theorems

Examples and Implementation

The classification of nilpotent Lie algebras

Following the p-group generation algorithm (Newman, O'Brien et al.):

If L is a nilpotent Lie algebra, then

$$L > L' = \gamma_2(L) > \gamma_3(L) > \cdots > \gamma_c(L) > \gamma_{c+1}(L) = 0.$$

L is an immediate descendant of $L/\gamma_c(L)$.

Stepsize: $\dim \gamma_c(L)$.

The classification of nilpotent Lie algebras

Following the p-group generation algorithm (Newman, O'Brien et al.):

If L is a nilpotent Lie algebra, then

$$L > L' = \gamma_2(L) > \gamma_3(L) > \cdots > \gamma_c(L) > \gamma_{c+1}(L) = 0.$$

L is an immediate descendant of $L/\gamma_c(L)$.

Stepsize: $\dim \gamma_c(L)$.

The classification of nilpotent Lie algebras

Following the p-group generation algorithm (Newman, O'Brien et al.):

If L is a nilpotent Lie algebra, then

$$L > L' = \gamma_2(L) > \gamma_3(L) > \cdots > \gamma_c(L) > \gamma_{c+1}(L) = 0.$$

L is an immediate descendant of $L/\gamma_c(L)$.

Stepsize: $\dim \gamma_c(L)$.

The classification of nilpotent Lie algebras

Following the p-group generation algorithm (Newman, O'Brien et al.):

If L is a nilpotent Lie algebra, then

$$L > L' = \gamma_2(L) > \gamma_3(L) > \cdots > \gamma_c(L) > \gamma_{c+1}(L) = 0.$$

L is an immediate descendant of $L/\gamma_c(L)$.

Stepsize: $\dim \gamma_c(L)$.

Immediate descendants and the cover

Suppose L is a nilpotent Lie algebra of class c.

The cover: The is a largest central extension $0 \to M \to L^* \to L \to 0$.

M is called the multiplicator and $\gamma_{c+1}(L^*)$ is the nucleus.

If $\overline{L}$ is a central extension of L then $\overline{L} \cong L^*/U$ where $U \leqslant M$.

$\overline{L}$ is an immediate descendant if and only if $U \neq M$ and $U + \gamma_{c+1}(L^*) = M$. Such a U is called allowable.

Immediate descendants and the cover

Suppose L is a nilpotent Lie algebra of class c.

The cover: The is a largest central extension $0 \to M \to L^* \to L \to 0$.

M is called the multiplicator and $\gamma_{c+1}(L^*)$ is the nucleus.

If $\overline{L}$ is a central extension of L then $\overline{L} \cong L^*/U$ where $U \leqslant M$.

$\overline{L}$ is an immediate descendant if and only if $U \neq M$ and $U + \gamma_{c+1}(L^*) = M$. Such a U is called allowable.

Immediate descendants and the cover

Suppose L is a nilpotent Lie algebra of class c.

The cover: The is a largest central extension $0 \to M \to L^* \to L \to 0$.

M is called the multiplicator and $\gamma_{c+1}(L^*)$ is the nucleus.

If $\overline{L}$ is a central extension of L then $\overline{L} \cong L^*/U$ where $U \leqslant M$.

$\overline{L}$ is an immediate descendant if and only if $U \neq M$ and $U + \gamma_{c+1}(L^*) = M$. Such a U is called allowable.

Immediate descendants and the cover

Suppose L is a nilpotent Lie algebra of class c.

The cover: The is a largest central extension $0 \to M \to L^* \to L \to 0$.

M is called the multiplicator and $\gamma_{c+1}(L^*)$ is the nucleus.

If $\overline{L}$ is a central extension of L then $\overline{L} \cong L^*/U$ where $U \leqslant M$.

$\overline{L}$ is an immediate descendant if and only if $U \neq M$ and $U + \gamma_{c+1}(L^*) = M$. Such a U is called allowable.

Determining descendants

Aut(L) acts on M.

Theorem

The isomorphism types of the immediate descendants of L correspond to the Aut(L)-orbits on the set of allowable subspaces.

Further

$$Aut(\overline{L}) = Aut(L^*/U) = Aut(L)_U \cdot p^u$$

where $u = (\dim L/\gamma_2(L)) \cdot (\dim M - \dim U)$.

Determining descendants

Aut(L) acts on M.

Theorem

The isomorphism types of the immediate descendants of L correspond to the Aut(L)-orbits on the set of allowable subspaces.

Further

$$Aut(\overline{L}) = Aut(L^*/U) = Aut(L)_U \cdot p^u$$

where $u = (\dim L/\gamma_2(L)) \cdot (\dim M - \dim U)$.

6-dimensional nilpotent Lie algebras over $\mathbf{F}_2$

Let's compute the 6-dim nilpotent Lie algebras over $\mathbf{F}_2$.

```
gap> l2 := [AbelianLieAlgebra( GF(2), 2 )];;
gap> l3 := [AbelianLieAlgebra( GF(2), 3 )];;
gap> for i in l2 do Append( l3, Descendants( i, 1 )); od; time;
16
gap> Length( l3 );
2
gap> l4 := [AbelianLieAlgebra( GF(2), 4 )];;
gap> for i in l2 do Append( l4, Descendants( i, 2 )); od; time;
0
gap> for i in l3 do Append( l4, Descendants( i, 1 )); od; time;
148
gap> Length( l4 );
3
gap> l5 := [AbelianLieAlgebra( GF(2), 5 )];;
gap> for i in l3 do Append( l5, Descendants( i, 2 )); od; time;
20
gap> for i in l4 do Append( l5, Descendants( i, 1 )); od; time;
648
gap> Length( l5 );
9
gap> l6 := [AbelianLieAlgebra( GF(2), 6 )];;
gap> for i in l3 do Append( l6, Descendants( i, 3 )); od; time;
4
gap> for i in l4 do Append( l6, Descendants( i, 2 )); od; time;
352
gap> for i in l5 do Append( l6, Descendants( i, 1 )); od; time;
1728
gap> Length( l6 );
36
```

Bottleneck: Orbit-stabilizer

Problem: Large number of subspaces for orbit computations.

E.g. compute step-3 immediate descendants of abelian Lie algebra $\langle x_1, \ldots, x_5 \rangle$.

$M = N = \langle [x_i, x_j] \mid i < j \rangle$.

Hence $\dim M = 10$, and every 3-dim subspace is allowable.

#(allowable subspaces): 6,347,715 (over $\mathbf{F}_2$), $1.8 \cdot 10^{11}$ (over $\mathbf{F}_3$), $6.2 \cdot 10^{15}$ (over $\mathbf{F}_5$).

Bottleneck: Orbit-stabilizer

Problem: Large number of subspaces for orbit computations.

E.g. compute step-3 immediate descendants of abelian Lie algebra $\langle x_1, \ldots, x_5 \rangle$.

$M = N = \langle [x_i, x_j] \mid i < j \rangle$.

Hence $\dim M = 10$, and every 3-dim subspace is allowable.

#(allowable subspaces): 6,347,715 (over $\mathbf{F}_2$), $1.8 \cdot 10^{11}$ (over $\mathbf{F}_3$), $6.2 \cdot 10^{15}$ (over $\mathbf{F}_5$).

Bottleneck: Orbit-stabilizer

Problem: Large number of subspaces for orbit computations.

E.g. compute step-3 immediate descendants of abelian Lie algebra $\langle x_1, \ldots, x_5 \rangle$.

$M = N = \langle [x_i, x_j] \mid i < j \rangle$.

Hence $\dim M = 10$, and every 3-dim subspace is allowable.

#(allowable subspaces): 6,347,715 (over $\mathbf{F}_2$), $1.8 \cdot 10^{11}$ (over $\mathbf{F}_3$), $6.2 \cdot 10^{15}$ (over $\mathbf{F}_5$).

Another trick

Task: Find all step-2 descendants of 7-dim abelian.

Need to determine the orbits of $GL(7,2)$ on the set of 2 dimensional subspaces acting on $\mathbf{F}_2^7 \wedge \mathbf{F}_2^7 \cong \mathbf{F}_2^{21}$.

There are $733,006,703,275$ subspaces.

The number of orbits can be found using the Cauchy-Frobenius Lemma:

$$\#\text{orbits} = \frac{1}{|G|}\sum_{g\in G} \text{fix}\, g = 20.$$

Using the list of groups with order 2^9 we can find 20 Lie algebras.

Another trick

Task: Find all step-2 descendants of 7-dim abelian.

Need to determine the orbits of $\mathrm{GL}(7,2)$ on the set of 2 dimensional subspaces acting on $\mathbf{F}_2^7 \wedge \mathbf{F}_2^7 \cong \mathbf{F}_2^{21}$.

There are $733,006,703,275$ subspaces.

The number of orbits can be found using the Cauchy-Frobenius Lemma:

$$\#\text{orbits} = \frac{1}{|G|}\sum_{g\in G} \operatorname{fix} g = 20.$$

Using the list of groups with order 2^9 we can find 20 Lie algebras.

Another trick

Task: Find all step-2 descendants of 7-dim abelian.

Need to determine the orbits of $\mathrm{GL}(7, 2)$ on the set of 2 dimensional subspaces acting on $\mathbf{F}_2^7 \wedge \mathbf{F}_2^7 \cong \mathbf{F}_2^{21}$.

There are $733,006,703,275$ subspaces.

The number of orbits can be found using the Cauchy-Frobenius Lemma:

$$\#\text{orbits} = \frac{1}{|G|}\sum_{g\in G} \operatorname{fix} g = 20.$$

Using the list of groups with order 2^9 we can find 20 Lie algebras.

Another trick

Task: Find all step-2 descendants of 7-dim abelian.

Need to determine the orbits of $\mathrm{GL}(7,2)$ on the set of 2 dimensional subspaces acting on $\mathbf{F}_2^7 \wedge \mathbf{F}_2^7 \cong \mathbf{F}_2^{21}$.

There are $733,006,703,275$ subspaces.

The number of orbits can be found using the Cauchy-Frobenius Lemma:

$$\#\text{orbits} = \frac{1}{|G|}\sum_{g\in G} \text{fix}\, g = 20.$$

Using the list of groups with order 2^9 we can find 20 Lie algebras.

Another trick

Task: Find all step-2 descendants of 7-dim abelian.

Need to determine the orbits of $\mathrm{GL}(7,2)$ on the set of 2 dimensional subspaces acting on $\mathbf{F}_2^7 \wedge \mathbf{F}_2^7 \cong \mathbf{F}_2^{21}$.

There are $733,006,703,275$ subspaces.

The number of orbits can be found using the Cauchy-Frobenius Lemma:

$$\#\text{orbits} = \frac{1}{|G|}\sum_{g\in G} \operatorname{fix} g = 20.$$

Using the list of groups with order 2^9 we can find 20 Lie algebras.

The number of small nilpotent Lie algebras

dimension	1	2	3	4	5	6	7	8	9
# nilp. $\mathbf{F}_2$-Lie algs	1	1	2	3	9	36	202	1831	27073
# nilp. $\mathbf{F}_3$-Lie algs	1	1	2	3	9	34	199		
# nilp. $\mathbf{F}_5$-Lie algs	1	1	2	3	9	34	211		

A generic computation

Suppose that

$$L = \langle 1, 2, 3, 4, 5 \mid [1, 2] = 3,\ [1, 3] = 4,\ [1, 4] = 5\rangle$$

over $\mathbf{F}_q$. Determine the step-1 descendants.

Multiplicator: $\langle [2, 3] = 6, [1, 5] = 7, [2, 5] = 8, [3, 4] = -8, [3, 5] = [4, 5] = 0\rangle$; nucleus: $\langle 7, 8\rangle$.

Number of allowable subspaces: $q^2 + q$. Then

$$\mathrm{Aut}(L) = \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} & a_{15} \\ 0 & a_{22} & a_{23} & a_{24} & a_{25} \\ 0 & 0 & a_{11}a_{22} & a_{11}a_{23} & a_{11}a_{24} \\ 0 & 0 & 0 & a_{11}^2 a_{22} & a_{11}^2 a_{23} \\ 0 & 0 & 0 & 0 & a_{11}^3 a_{22} \end{pmatrix}.$$

$|\mathrm{Aut}(L)| = (q-1)^2 q^7$.

A generic computation

Suppose that

$$L = \langle 1, 2, 3, 4, 5 \mid [1,2] = 3,\ [1,3] = 4,\ [1,4] = 5 \rangle$$

over $\mathbf{F}_q$. Determine the step-1 descendants.

Multiplicator:
$\langle [2,3] = 6, [1,5] = 7, [2,5] = 8, [3,4] = -8, [3,5] = [4,5] = 0 \rangle$;
nucleus: $\langle 7, 8 \rangle$.

Number of allowable subspaces: $q^2 + q$. Then

$$\mathrm{Aut}(L) = \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} & a_{15} \\ 0 & a_{22} & a_{23} & a_{24} & a_{25} \\ 0 & 0 & a_{11}a_{22} & a_{11}a_{23} & a_{11}a_{24} \\ 0 & 0 & 0 & a_{11}^2 a_{22} & a_{11}^2 a_{23} \\ 0 & 0 & 0 & 0 & a_{11}^3 a_{22} \end{pmatrix}.$$

$|\mathrm{Aut}(L)| = (q-1)^2 q^7$.

A generic computation

Suppose that

$$L = \langle 1,2,3,4,5 \mid [1,2]=3,\ [1,3]=4,\ [1,4]=5 \rangle$$

over $\mathbf{F}_q$. Determine the step-1 descendants.

Multiplicator:
$\langle [2,3]=6, [1,5]=7, [2,5]=8, [3,4]=-8, [3,5]=[4,5]=0 \rangle$;
nucleus: $\langle 7,8 \rangle$.

Number of allowable subspaces: q^2+q. Then

$$\mathrm{Aut}(L) = \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} & a_{15} \\ 0 & a_{22} & a_{23} & a_{24} & a_{25} \\ 0 & 0 & a_{11}a_{22} & a_{11}a_{23} & a_{11}a_{24} \\ 0 & 0 & 0 & a_{11}^2 a_{22} & a_{11}^2 a_{23} \\ 0 & 0 & 0 & 0 & a_{11}^3 a_{22} \end{pmatrix}.$$

$|\mathrm{Aut}(L)| = (q-1)^2 q^7$.

A generic computation

Suppose that

$$L = \langle 1,2,3,4,5 \mid [1,2]=3,\ [1,3]=4,\ [1,4]=5\rangle$$

over $\mathbf{F}_q$. Determine the step-1 descendants.

Multiplicator:
$\langle [2,3]=6, [1,5]=7, [2,5]=8, [3,4]=-8, [3,5]=[4,5]=0\rangle$;
nucleus: $\langle 7,8\rangle$.

Number of allowable subspaces: q^2+q. Then

$$\mathrm{Aut}(L) = \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} & a_{15} \\ 0 & a_{22} & a_{23} & a_{24} & a_{25} \\ 0 & 0 & a_{11}a_{22} & a_{11}a_{23} & a_{11}a_{24} \\ 0 & 0 & 0 & a_{11}^2a_{22} & a_{11}^2a_{23} \\ 0 & 0 & 0 & 0 & a_{11}^3a_{22} \end{pmatrix}.$$

$|\mathrm{Aut}(L)| = (q-1)^2q^7$.

Orbits and stabilisers

Orbit 1
$\langle (1,0,0),(0,0,1)\rangle$
Stabiliser: $S_1 = \mathrm{Aut}(L)$
Orbit size: 1
Orbit 2
Representative: $\langle (1,0,0),(0,1,-1)\rangle$
Stabiliser: $S_2 = \left\langle a_{12} = a_{22} - a_{11},\ a_{24} = (-1/2)a_{23}^2/a_{22}\right\rangle$
Orbit size: q^2
Orbit 3
Representative: $\langle (1,-1,0),(0,0,1)\rangle$
Stabiliser: $S_3 = \left\langle a_{22} = a_{11}^3\right\rangle$
Orbit size: $q-1$
The number of points in total is q^2+q.

Another instructive example

Compute step-2 descendants of $L = \mathbf{F}_q^4$ where q is odd.

$\mathrm{Aut}(L) = \mathrm{GL}(4, q)$. Multiplicator=Nucleus= $W = L \wedge L$.

$W = \langle e_1 = [1, 2], e_2 = [1, 3], e_3 = [1, 4],$
$f_3 = [2, 3], f_2 = [4, 2], f_1 = [3, 4]\rangle$.

Orthogonal form on W: $(e_i, e_j) = (f_i, f_j) = 0$; $(e_i, f_j) = \delta_{ij}$.

$\mathrm{Aut}(L)$ preserves form modulo scalars:
$(xg, yg) = (\det g)(x, y)$.

The subspaces of W

There are 4 different 4-dimensional subspaces U of W:

(i) form is non-degenerate on U with type $+$:
$\langle e_1, e_2, f_1, f_2\rangle$.

(ii) form is non-degenerate on U with type $-$:
$\langle (0,0,2,1,-2,0),(0,1,2,0,-a,0),(1,0,-2,0,-a,0),(0,0,0,0,0,1)\rangle$
where $a/2$ is not a square.

(iii) form is degenerate on U with 1-dim kernel:
$\langle e_1+f_1, e_2, e_3, f_2\rangle$.

(iv) form is degenerate on U with 2-dim kernel:
$\langle e_1, e_2, e_3, f_1\rangle$.

6-dim nilpotent Lie algebras

Theorem

There are 34 isomorphism classes of nilpotent Lie algebras over finite fields with odd characteristic. There are 36 such classes over $\mathbf{F}_2$.

Theorem (Willem)

Let char $\mathbb{F} \neq 2$. *Then there are* $26 + 4|F^*/(F^*)^2|$ *isomorphism types of nilpotent Lie algebras with dimension* 6.

6-dim nilpotent Lie algebras

Theorem

There are 34 isomorphism classes of nilpotent Lie algebras over finite fields with odd characteristic. There are 36 such classes over $\mathbf{F}_2$.

Theorem (Willem)

Let char $\mathbb{F} \neq 2$. *Then there are* $26 + 4|F^*/(F^*)^2|$ *isomorphism types of nilpotent Lie algebras with dimension* 6.

The classification of soluble Lie algebras of dimension at most 4

Theorem (Willem '05)

The number of soluble Lie algebras of dimension 3 over $\mathbf{F}_q$ is $q + 5$ if char $\mathbf{F}_q \neq 2$ and $q + 4$ otherwise.
The number of soluble Lie algebras of dimension 4 over $\mathbf{F}_q$ is

$$q^2 + 3q + 9 + \begin{cases} 5 & \text{if } q \equiv 1 \pmod 6 \\ 2 & \text{if } q \equiv 2 \pmod 6 \\ 3 & \text{if } q \equiv 3 \pmod 6 \\ 4 & \text{if } q \equiv 4 \pmod 6 \\ 3 & \text{if } q \equiv 5 \pmod 6. \end{cases}$$

"Which is slightly more than the number found in Patera & Zassenhaus."

Non-solvable Lie algebras

Theorem (Strade)

Over a finite field $\mathbf{F}_q$, the number of nonsolvable Lie algebras

(iii) *of dimension 3 is 1;*

(iv) *of dimension 4 over char 2 is 2; over odd char it is 1;*

(v) *of dimension 5 is 5, 4, 3 over char 2, [3 and 5], and $\geqslant 7$, respectively;*

(vi) *of dimension 6 is $14 + 2q$, $13 + (5/3)q + \varepsilon$, $13 + q$, $11 + q$ over fields of characteristic 2, 3, 5, and $\geqslant 7$.*

Simple Lie algebras

Theorem (Vaughan-Lee)

The number of isomorphism types of 7, 8, *and* 9-*dimensional simple Lie algebras over* $\mathbf{F}_2$ *is* 2, 2, 1, *respectively.*

The LieAlgDB package

```
gap> SolvableLieAlgebra( GF(27), [4,3,1] );
<Lie algebra of dimension 4 over GF(3^3)>
gap> NonSolvableLieAlgebra( GF(27), [5,3] );
sl(2,27).V(1)
```

```
gap> L := AllNonSolvableLieAlgebras( GF(5^20), 6 );
Nonsolvable Lie algebras with dimension 6 over GF(5^20)
gap> Size( L );
95367431640638
gap> e := Enumerator( L );
<enumerator>
gap> e[1233223];
sl(2,95367431640625)+solv([ 3, 4*Z(5,20)^11+4*Z(5,20)^13+3*Z(5,20)^15+4*Z(5,20)^17+4*Z(5,20)^18+4*Z(5,20)^19 ])
```

Nonsolvable algebras over characteristic 3

```
gap> L := AllNonSolvableLieAlgebras( GF(81), 6 );
Nonsolvable Lie algebras with dimension 6 over GF(3^4)
gap> e := Enumerator( L );
fail
gap> e := Iterator( L );
<iterator>
gap> z := [];; for i in e do
gap> Add( z, Dimension( LieCenter( i )));
gap> od;
gap> z;
[ 0, 0, 3, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
  ...
  0, 0 ]
```

Lie algebra identification

```
gap> SolvableLieAlgebra( GF(25), [4,3,1] );
<Lie algebra of dimension 4 over GF(5^2)>
gap> LieAlgebraIdentification( last ); time;
rec( name := "L4_3( GF(5^2), Z(5)^0 )",
     parameters := [ Z(5)^0 ],
     isomorphism := CanonicalBasis(
         <Lie algebra of dimension 4 over GF(5^2)> ) -
         [ v.1, v.2, v.3, v.4 ] )
12
gap>
```


Sophus and LieAlgDB
Determining nilpotent Lie algebras
Problems and solutions
Generic computations
Classification theorems
Examples and Implementation

Storing nilpotent Lie algebras

The package contains about 30000 nilpotent Lie algebras.

Every such algebra is encoded:
Let $L = \langle x_1, \ldots, x_d \rangle$ be such an algebra over $\mathbb{F}_p$. Then

$$[x_i, x_j] = \sum_{k=j+1}^{d} \alpha_{i,j}^{d} x_d \quad \text{for} \quad i < j.$$

Write down the $\alpha_{i,j}^{d}$ in a certain order and consider it as a number in base p. Convert this number to base 62 using the digits, $0, \ldots, 9, a, \ldots, z, A, \ldots, Z$.

These strings are stored in the global variables `_liealgdb_nilpotent_d*f*`.

Storing nilpotent Lie algebras

The package contains about 30000 nilpotent Lie algebras.

Every such algebra is encoded:
Let $L = \langle x_1, \ldots, x_d \rangle$ be such an algebra over $\mathbb{F}_p$. Then

$$[x_i, x_j] = \sum_{k=j+1}^{d} \alpha_{i,j}^{d} x_d \quad \text{for} \quad i < j.$$

Write down the $\alpha_{i,j}^{d}$ in a certain order and consider it as a number in base p. Convert this number to base 62 using the digits, $0, \ldots, 9, a, \ldots, z, A, \ldots, Z$.

These strings are stored in the global variables `_liealgdb_nilpotent_d*f*`.

Storing nilpotent Lie algebras

The package contains about 30000 nilpotent Lie algebras.

Every such algebra is encoded:
Let $L = \langle x_1, \ldots, x_d \rangle$ be such an algebra over $\mathbb{F}_p$. Then

$$[x_i, x_j] = \sum_{k=j+1}^{d} \alpha_{i,j}^{d} x_d \quad \text{for} \quad i < j.$$

Write down the $\alpha_{i,j}^{d}$ in a certain order and consider it as a number in base p. Convert this number to base 62 using the digits, $0, \ldots, 9$, $a, \ldots, z$, $A, \ldots, Z$.

These strings are stored in the global variables `_liealgdb_nilpotent_d*f*`.

Storing nilpotent Lie algebras

The package contains about 30000 nilpotent Lie algebras.

Every such algebra is encoded:
Let $L = \langle x_1, \ldots, x_d \rangle$ be such an algebra over $\mathbb{F}_p$. Then

$$[x_i, x_j] = \sum_{k=j+1}^{d} \alpha_{i,j}^{d} x_d \quad \text{for} \quad i < j.$$

Write down the $\alpha_{i,j}^{d}$ in a certain order and consider it as a number in base p. Convert this number to base 62 using the digits, $0, \ldots, 9$, $a, \ldots, z$, $A, \ldots, Z$.

These strings are stored in the global variables `_liealgdb_nilpotent_d*f*`.

Loading nilpotent Lie algebras

The files containing the codewords for nilpotent Lie algebras are about $1/2$ MB long.

We don't want to read these files, unless the user really needs them. So we added to read.g

```
DeclareAutoreadableVariables( "liealgdb",
        "gap/nilpotent/nilpotent_data62.gi",
        ["_liealgdb_nilpotent_d6f2"] );
```

Loading nilpotent Lie algebras

The files containing the codewords for nilpotent Lie algebras are about 1/2 MB long.

We don't want to read these files, unless the user really needs them. So we added to read.g

```
DeclareAutoreadableVariables( "liealgdb",
        "gap/nilpotent/nilpotent_data62.gi",
        ["_liealgdb_nilpotent_d6f2"] );
```

To do

Sophus

(i) more computations over extension fields;

(ii) better automorphism group and isomorphism testing.

LieAlgDB

(i) 6-dim nilpotent over characteristic 2;

(ii) check Strade's classification;

(iii) add more classes of algebras.

References

W. A. de Graaf. Classification of solvable Lie algebras. *Experiment. Math.* **14**(1):15–25, 2005. arxiv.org/abs/math.RA/0404071.

Willem A. de Graaf. Classification of 6-dimensional nilpotent Lie algebras over fields of characteristic not 2. *J. Algebra*, **309**(2):640–653, 2007. arxiv.org/abs/math.RA/0511668.

Csaba Schneider. A computer-based approach to the classification of nilpotent Lie algebras. *Experiment. Math.* **14**(2):153-160, 2005. arxiv.org/math.RA/0406365

Helmut Strade. Lie algebras of small dimension. arxiv.org/abs/math.RA/0601413.

Michael Vaughan-Lee. Simple Lie algebras of small dimension over GF(2). *LMS J. Comput. Math.* **9**:174–192, 2006.